

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、誤りが無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

μPD753208

4ビット・シングルチップ・マイクロコンピュータ

μPD753204

μPD753206

μPD753208

μPD75P3216

〔メモ〕

目 次 要 約

第1章	概 説	...	23
第2章	端子機能	...	31
第3章	アーキテクチャの特徴とメモリ・マップ	...	47
第4章	内部CPU機能	...	75
第5章	周辺ハードウェア機能	...	99
第6章	割り込み機能とテスト機能	...	295
第7章	スタンバイ機能	...	327
第8章	リセット機能	...	337
第9章	PROM (プログラム・メモリ) の書き込みとベリファイ	...	341
第10章	マスク・オプション	...	347
第11章	命令セット	...	349
付録A	μ PD753108 , 753208 , 75P3216の機能一覧表	...	429
付録B	開発ツール	...	431
付録C	マスクROM発注の手順	...	437
付録D	命令索引	...	439
付録E	ハードウェア索引	...	445
付録F	改版履歴	...	451

CMOSデバイスの一般的注意事項

静電気対策（MOS全般）

注意 MOSデバイス取り扱いの際は静電気防止を心がけてください。

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

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

未使用入力の処理（CMOS特有）

注意 CMOSデバイスの入力レベルは固定してください。

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

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

初期化以前の状態（MOS全般）

注意 電源投入時、MOSデバイスの初期状態は不定です。

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

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

QTOPIは、日本電気株式会社の商標です。

MS-DOS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。

本製品が外国為替および外国貿易管理法の規定による規制貨物等（または役務）に該当するか否かは、ユーザー（仕様を決定した者）が判定してください。

- 本資料の内容は予告なく変更することがありますので、最新のものであることをご確認の上ご使用ください。
 - 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。

M7 98.8

本版で改訂された主な箇所

箇所	内容
全般	μPD753204, 753206, 753208, 75P3216を開発中から開発済みに変更
	ポート 4, 5 のN-chオープン・ドレイン時, 入力耐圧を12 Vから13 Vに変更
	データ・バス端子 (D0-D7) を追加
p.45	表 2 - 3 未使用端子の処理一覧を変更
p.75	表 4 - 1 Mk モードとMk モードの違いの注意を変更
p.232, 233	5.6.7 (a) バス・リリース信号 (REL) と 5.6.7 (b) コマンド信号 (CMD) に注意を追加
p.334	7.4 マスク・オプションの選択を追加
p.343	9.2 プログラム・メモリの書き込みの手順を変更
p.344	9.3 プログラム・メモリの読み出しの手順を変更
p.378	11.3 各命令の命令コードの命令一覧表を修正
p.437	付録 C マスクROM発注の手順の発注用媒体を変更
p.451	付録 F 改版履歴を追加

本文欄外の★印は、本版で改訂された主な箇所を示しています。

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

はじめに

ご利用対象者 このマニュアルは、 μ PD753204、753206、753208、75P3216の機能を理解し、それを用いたアプリケーション・システムを設計するユーザのエンジニアを対象としています。

目的 このマニュアルは、次の構成に示す μ PD753204、753206、753208、75P3216の持つハードウェア機能をユーザに理解していただくことを目的としています。

構成 このマニュアルは、大きく分けて次の内容で構成しております。

- 概説
- 端子機能
- アーキテクチャの特徴とメモリ・マップ
- 内部CPU機能
- 周辺ハードウェア機能
- 割り込み機能とテスト機能
- スタンバイ機能
- リセット機能
- PROMの書き込みとベリファイ
- マスク・オプション
- 命令セット

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

μ PD753108をすでに経験しているユーザ

付録A μ PD753108、753208、75P3216の機能一覧表で違っている部分を確認し、それらの説明を読んでください。

μ PD753204、753206、75P3216のマニュアルとしてお使いになるユーザ

このマニュアルは、特に機能面で違いがないかぎり μ PD753208を代表品種として説明してあります。1.3 μ PD753208サブシリーズ製品間の違いで機能の違いを確認したうえで、 μ PD753208をそれぞれの製品に読み替えて使用してください。

ニモニックが分かっているときの命令機能を調べるとき

付録D 命令索引を利用してください。

特定の内蔵回路などの機能を調べたいとき

付録E ハードウェア索引を利用してください。

一通り μ PD753204, 753206, 753208, 75P3216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。

凡 例	データの表記の重み	: 左が上位桁, 右が下位桁
	アクティブ・ロウの表記	: $\overline{\text{xxx}}$ (端子, 信号名称に上線)
	メモリ・マップのアドレス	: 上部-下位, 下部-上位
	注	: 本文中につけた注の説明
	注意	: 気をつけて読んでいただきたい内容
	備考	: 本文の補足説明
	重要事項, 強調	: 太字で表記
	数の表記	: 2進数..... xxxxB 10進数..... xxx 16進数..... xxxH

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。

デバイス関連資料一覧

資料名	資料番号	
	和文	英文
μPD753204,753206,753208 データ・シート	U10166J	U10166E
μPD75P3216 データ・シート	U10241J	U10241E
μPD753208 ユーザーズ・マニュアル	U10158J (このマニュアル)	U10158E
75XLシリーズ セレクション・ガイド	U10453J	U10453E

開発ツール関連資料一覧

資料名	資料番号			
	和文	英文		
ハードウェア	IE-75000-R/IE-75001-R ユーザーズ・マニュアル	EEU-846	EEU-1416	
	IE-75300-R-EM ユーザーズ・マニュアル	U11354J	U11354E	
	EP-753208GT-R ユーザーズ・マニュアル	U10739J	U10739E	
	PG-1500 ユーザーズ・マニュアル	U11940J	U11940E	
ソフトウェア	RA75X アセンブラ・パッケージ ユーザーズ・マニュアル	操作編	U12622J	U12622E
		言語編	U12385J	U12385E
	PG-1500コントローラ ユーザーズ・マニュアル	PC-9800シリーズ (MS-DOS)ベース	EEU-704	EEU-1291
		IBM PCシリーズ (PC DOS)ベース	EEU-5008	U10540E

その他のドキュメント

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 社外メーカー編	U11416J	-

注意 上記関連ドキュメント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

第1章 概説 ... 23

- 1.1 機能概要 ... 24
- 1.2 オーダ情報 ... 26
- 1.3 μ PD753208サブシリーズ製品間の違い ... 26
- 1.4 ブロック図 ... 27
- 1.5 端子接続図 (Top View) ... 28

第2章 端子機能 ... 31

- 2.1 μ PD753208の端子機能一覧 ... 31
- 2.2 端子機能の説明 ... 35
 - 2.2.1 P00-P03 (PORT0), P10, P13 (PORT1) ... 35
 - 2.2.2 P20-P23 (PORT2), P30-P33 (PORT3)
P50-P53 (PORT5), P60-P63 (PORT6)
P80-P83 (PORT8), P90-P93 (PORT9) ... 36
 - 2.2.3 T10 ... 37
 - 2.2.4 PTO0-PTO2 ... 37
 - 2.2.5 PCL ... 37
 - 2.2.6 BUZ ... 37
 - 2.2.7 \overline{SCK} , SO/SB0, SI/SB1 ... 37
 - 2.2.8 INT4 ... 38
 - 2.2.9 INT0 ... 38
 - 2.2.10 KR0-KR3 ... 38
 - 2.2.11 S12-S23 ... 38
 - 2.2.12 COM0-COM3 ... 39
 - 2.2.13 VLc0-VLc2 ... 39
 - 2.2.14 BIAS ... 39
 - 2.2.15 LCDCL ... 39
 - 2.2.16 SYNC ... 39
 - 2.2.17 X1, X2 ... 39
 - 2.2.18 \overline{RESET} ... 40
 - 2.2.19 MD0-MD3 (μ PD75P3216のみ) ... 40
 - ★ 2.2.20 D0-D7 (μ PD75P3216のみ) ... 40
 - 2.2.21 IC (μ PD753204, 753206, 753208のみ) ... 40
 - 2.2.22 VPP (μ PD75P3216のみ) ... 40
 - 2.2.23 VDD ... 41
 - 2.2.24 VSS ... 41
- 2.3 端子の入出力回路 ... 42
- 2.4 未使用端子の処理について ... 45

第3章	アーキテクチャの特徴とメモリ・マップ	... 47
3.1	データ・メモリのバンク構成とアドレッシング・モード	... 47
3.1.1	データ・メモリのバンク構成	... 47
3.1.2	データ・メモリのアドレッシング・モード	... 49
3.2	汎用レジスタのバンク構成	... 63
3.3	メモリ・マップトI/O	... 68
第4章	内部CPU機能	... 75
4.1	Mk モードとMk モードの切り替え機能	... 75
4.1.1	Mk モードとMk モードの違い	... 75
4.1.2	スタック・バンク選択レジスタ (SBS) の設定方法	... 76
4.2	プログラム・カウンタ (PC)	... 77
4.3	プログラム・メモリ (ROM)	... 78
4.4	データ・メモリ (RAM)	... 83
4.4.1	データ・メモリの構成	... 83
4.4.2	データ・メモリのバンクの指定	... 84
4.5	汎用レジスタ	... 88
4.6	アキュムレータ	... 89
4.7	スタック・ポインタ (SP), スタック・バンク選択レジスタ (SBS)	... 89
4.8	プログラム・ステータス・ワード (PSW)	... 93
4.9	バンク選択レジスタ (BS)	... 97
第5章	周辺ハードウェア機能	... 99
5.1	デジタル入出力ポート	... 99
5.1.1	デジタル入出力ポートの種類, 特徴, 構成	... 100
5.1.2	入出力モードの設定	... 105
5.1.3	デジタル入出力ポート操作命令	... 107
5.1.4	デジタル入出力ポートの動作	... 110
5.1.5	プルアップ抵抗の内蔵	... 112
5.1.6	デジタル入出力ポートの入出力タイミング	... 114
5.2	クロック発生回路	... 116
5.2.1	クロック発生回路の構成	... 116
5.2.2	クロック発生回路の機能と動作	... 117
5.2.3	CPUクロックの設定	... 123
5.2.4	クロック出力回路	... 125
5.3	ベーシック・インターバル・タイマ/ウォッチドッグ・タイマ	... 128
5.3.1	ベーシック・インターバル・タイマ/ウォッチドッグ・タイマの構成	... 128
5.3.2	ベーシック・インターバル・タイマ・モード・レジスタ (BTM)	... 129
5.3.3	ウォッチドッグ・タイマ許可フラグ (WDTM)	... 131
5.3.4	ベーシック・インターバル・タイマの動作	... 132
5.3.5	ウォッチドッグ・タイマの動作	... 133
5.3.6	その他の機能	... 135

5.4	時計用タイマ	...	137
5.4.1	時計用タイマの構成	...	137
5.4.2	時計モード・レジスタ	...	138
5.5	タイマ/イベント・カウンタ	...	140
5.5.1	タイマ/イベント・カウンタの構成	...	140
5.5.2	8ビット・タイマ/イベント・カウンタ・モードの動作	...	150
5.5.3	PWMパルス・ジェネレータ・ モード (PWMモード) の動作	...	162
5.5.4	16ビット・タイマ・カウンタ・モードの動作	...	168
5.5.5	キャリア・ジェネレータ・モード (CGモード) の動作	...	178
5.5.6	タイマ/イベント・カウンタ応用上の注意	...	190
5.6	シリアル・インタフェース	...	197
5.6.1	シリアル・インタフェースの機能	...	197
5.6.2	シリアル・インタフェースの構成	...	198
5.6.3	レジスタの機能	...	202
5.6.4	動作停止モード	...	210
5.6.5	3線式シリアルI/Oモードの動作	...	212
5.6.6	2線式シリアルI/Oモードの動作	...	222
5.6.7	SBIモードの動作	...	229
5.6.8	SCK端子出力の操作	...	262
5.7	LCDコントローラ/ドライバ	...	263
5.7.1	LCDコントローラ/ドライバの構成	...	263
5.7.2	LCDコントローラ/ドライバの機能	...	265
5.7.3	表示モード・レジスタ (LCDM)	...	265
5.7.4	表示制御レジスタ (LCDC)	...	267
5.7.5	LCD/ポート選択レジスタ (LPS)	...	269
5.7.6	表示データ・メモリ	...	270
5.7.7	コモン信号とセグメント信号	...	272
5.7.8	LCD駆動用電源V _{LC0} , V _{LC1} , V _{LC2} の供給	...	276
5.7.9	表示モード	...	279
5.8	ビット・シーケンシャル・バッファ	...	292

第6章 割り込み機能とテスト機能 ... 295

6.1	割り込み制御回路の構成	...	295
6.2	割り込みソースの種類とベクタ・テーブル	...	297
6.3	割り込み機能を制御する各種ハードウェア	...	299
6.4	割り込みシーケンス	...	307
6.5	多重割り込み処理制御	...	308
6.6	ベクタ・アドレス共用割り込み処理	...	310
6.7	割り込み処理までのマシン・サイクル	...	312
6.8	割り込みの効果的な使い方	...	314
6.9	割り込みの応用	...	314
6.10	テスト機能	...	322
6.10.1	テスト・ソースの種類	...	322
6.10.2	テスト機能を制御する各種ハードウェア	...	322

第7章 スタンバイ機能 ... 327

- 7.1 スタンバイ・モードの設定および動作状態 ... 329
- 7.2 スタンバイ・モードの解除 ... 331
- 7.3 スタンバイ・モード解除後の動作 ... 333
- ★ 7.4 マスク・オプションの選択 ... 334
- 7.5 スタンバイ・モードの応用 ... 334

第8章 リセット機能 ... 337

第9章 PROM (プログラム・メモリ) の書き込みとベリファイ ... 341

- 9.1 プログラム・メモリ書き込み/ベリファイ時の動作モード ... 342
- 9.2 プログラム・メモリの書き込みの手順 ... 343
- 9.3 プログラム・メモリの読み出しの手順 ... 344
- 9.4 ワン・タイムPROMのスクリーニングについて ... 345

第10章 マスク・オプション ... 347

- 10.1 端子 ... 347
 - 10.1.1 P50-P53のマスク・オプション ... 347
 - 10.1.2 VLc0-VLc2のマスク・オプション ... 347
- 10.2 スタンバイ機能のマスク・オプション ... 348

第11章 命令セット ... 349

- 11.1 特徴的な命令 ... 349
 - 11.1.1 GETI命令 ... 349
 - 11.1.2 ビット操作命令 ... 350
 - 11.1.3 たてづみ命令 ... 350
 - 11.1.4 進数補正命令 ... 351
 - 11.1.5 スキップ命令とスキップに要するマシン・サイクル数 ... 352
- 11.2 命令セットとそのオペレーション ... 353
- 11.3 各命令の命令コード ... 373
- 11.4 命令機能と応用 ... 379
 - 11.4.1 転送命令 ... 380
 - 11.4.2 テーブル参照命令 ... 388
 - 11.4.3 ビット転送命令 ... 392
 - 11.4.4 演算命令 ... 393
 - 11.4.5 アキュムレータ操作命令 ... 401
 - 11.4.6 増減命令 ... 402
 - 11.4.7 比較命令 ... 404
 - 11.4.8 キャリー・フラグ操作命令 ... 406
 - 11.4.9 メモリ・ビット操作命令 ... 407
 - 11.4.10 分岐命令 ... 411
 - 11.4.11 サブルーチン・スタック制御命令 ... 416
 - 11.4.12 割り込み制御命令 ... 421
 - 11.4.13 入出力命令 ... 422

11.4.14 CPU制御命令 ... 424

11.4.15 特殊命令 ... 425

付録A μ PD753108 , 753208 , 75P3216の機能一覧表 ... 429

付録B 開発ツール ... 431

付録C マスクROM発注の手順 ... 437

付録D 命令索引 ... 439

D.1 命令索引(機能別) ... 439

D.2 命令索引(アルファベット順) ... 442

付録E ハードウェア索引 ... 445

E.1 ハードウェア索引(50音順) ... 445

E.2 ハードウェア索引(アルファベット順) ... 448

★ **付録F** 改版履歴 ... 451

図の目次 (1/5)

図番号	タイトル, ページ
3 - 1	MBE = 0 のモードと MBE = 1 のモードの使い分け ... 48
3 - 2	データ・メモリの構成と, 各アドレッシング・モードのアドレッシング範囲 ... 50
3 - 3	スタティックRAMのアドレス更新方法 ... 56
3 - 4	レジスタ・バンクの使い分け例 ... 64
3 - 5	汎用レジスタの構成 (4ビット処理の場合) ... 66
3 - 6	汎用レジスタの構成 (8ビット処理の場合) ... 67
3 - 7	μPD753208 I/Oマップ ... 70
4 - 1	スタック・バンク選択レジスタのフォーマット ... 76
4 - 2	プログラム・カウンタの構成 ... 77
4 - 3	プログラム・メモリ・マップ ... 79
4 - 4	データ・メモリ・マップ ... 85
4 - 5	表示データ・メモリの構成 ... 87
4 - 6	汎用レジスタの構成 ... 88
4 - 7	レジスタ・ペアの構成 ... 88
4 - 8	アキュムレータ ... 89
4 - 9	スタック・ポインタおよびスタック・バンク選択レジスタの構成 ... 90
4 - 10	スタック・メモリへ退避されるデータ (Mk モード) ... 91
4 - 11	スタック・メモリから復帰されるデータ (Mk モード) ... 91
4 - 12	スタック・メモリへ退避されるデータ (Mk モード) ... 92
4 - 13	スタック・メモリから復帰されるデータ (Mk モード) ... 92
4 - 14	プログラム・ステータス・ワードの構成 ... 93
4 - 15	バンク選択レジスタの構成 ... 97
5 - 1	デジタル・ポートのデータ・メモリ・アドレス ... 99
5 - 2	ポート 0, 1 の構成 ... 101
5 - 3	ポート 3, 6 の構成 ... 102
5 - 4	ポート 2 の構成 ... 102
5 - 5	ポート 5 の構成 ... 103
5 - 6	ポート 8, 9 の構成 ... 104
5 - 7	各ポート・モード・レジスタのフォーマット ... 106
5 - 8	プルアップ抵抗レジスタのフォーマット ... 113
5 - 9	デジタル入出力ポートの入出力タイミング ... 114
5 - 10	ソフトウェアによる内蔵プルアップ抵抗のONタイミング ... 115

図の目次 (2/5)

図番号	タイトル, ページ
5 - 11	クロック発生回路のブロック図 ... 116
5 - 12	プロセッサ・クロック・コントロール・レジスタのフォーマット ... 119
5 - 13	システム・クロック発振回路の外付け回路 ... 120
5 - 14	発振子の接続回路の悪い例 ... 121
5 - 15	CPUクロックの切り替え ... 124
5 - 16	クロック出力回路のブロック図 ... 125
5 - 17	クロック出力モード・レジスタのフォーマット ... 126
5 - 18	リモコン波形出力応用例 ... 127
5 - 19	ベーシック・インターバル・タイマ/ウォッチドッグ・タイマのブロック図 ... 128
5 - 20	ベーシック・インターバル・タイマ・モード・レジスタのフォーマット ... 130
5 - 21	ウォッチドッグ・タイマ許可フラグ (WDTM) のフォーマット ... 131
5 - 22	時計用タイマのブロック図 ... 138
5 - 23	時計モード・レジスタのフォーマット ... 139
5 - 24	タイマ/イベント・カウンタ (チャンネル0) のブロック図 ... 141
5 - 25	タイマ・カウンタ (チャンネル1) のブロック図 ... 142
5 - 26	タイマ・カウンタ (チャンネル2) のブロック図 ... 143
5 - 27	タイマ/イベント・カウンタ・モード・レジスタ (チャンネル0) のフォーマット ... 145
5 - 28	タイマ・カウンタ・モード・レジスタ (チャンネル1) のフォーマット ... 147
5 - 29	タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマット ... 147
5 - 30	タイマ/イベント・カウンタ出力許可フラグのフォーマット ... 148
5 - 31	タイマ・カウンタ・コントロール・レジスタのフォーマット ... 149
5 - 32	タイマ/イベント・カウンタ・モード・レジスタの設定 ... 151
5 - 33	タイマ・カウンタ・コントロール・レジスタの設定 ... 154
5 - 34	タイマ/イベント・カウンタ出力許可フラグの設定 ... 154
5 - 35	タイマ/イベント・カウンタ動作時の構成 ... 157
5 - 36	カウント動作時のタイミング ... 157
5 - 37	カウント動作時の構成 ... 159
5 - 38	カウント動作時のタイミング ... 160
5 - 39	タイマ・カウンタ・モード・レジスタの設定 ... 163
5 - 40	タイマ・カウンタ・コントロール・レジスタの設定 ... 164
5 - 41	PWMパルス・ジェネレータ動作時の構成 ... 166
5 - 42	PWMパルス・ジェネレータ動作時のタイミング ... 166
5 - 43	タイマ・カウンタ・モード・レジスタの設定 ... 169
5 - 44	タイマ・カウンタ・コントロール・レジスタの設定 ... 170

図の目次 (3/5)

図番号	タイトル, ページ
5 - 45	タイマ・カウンタ動作時の構成 ... 173
5 - 46	カウント動作時のタイミング ... 173
5 - 47	カウント動作時の構成 ... 175
5 - 48	カウント動作時のタイミング ... 176
5 - 49	タイマ・カウンタ・モード・レジスタの設定 (n=1, 2) ... 179
5 - 50	タイマ・カウンタ出力許可フラグの設定 ... 180
5 - 51	タイマ・カウンタ・コントロール・レジスタの設定 ... 180
5 - 52	キャリア・ジェネレータ動作時の構成 ... 183
5 - 53	キャリア・ジェネレータ動作時のタイミング ... 184
5 - 54	SBIのシステム構成例 ... 198
5 - 55	シリアル・インタフェースのブロック図 ... 199
5 - 56	シリアル動作モード・レジスタ (CSIM) のフォーマット ... 202
5 - 57	シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) のフォーマット ... 205
5 - 58	シフト・レジスタの周辺の構成 ... 208
5 - 59	3線式シリアルI/Oのシステム構成例 ... 212
5 - 60	3線式シリアルI/Oモードのタイミング ... 215
5 - 61	RELT, CMDTの動作 ... 216
5 - 62	転送ビットの切り替え回路 ... 217
5 - 63	2線式シリアルI/Oのシステム構成例 ... 222
5 - 64	2線式シリアルI/Oモードのタイミング ... 225
5 - 65	RELT, CMDTの動作 ... 226
5 - 66	SBIのシステム構成例 ... 229
5 - 67	SBI転送のタイミング ... 231
5 - 68	バス・リリース信号 ... 232
5 - 69	コマンド信号 ... 232
5 - 70	アドレス ... 233
5 - 71	アドレスによるスレーブの選択 ... 233
5 - 72	コマンド ... 234
5 - 73	データ ... 234
5 - 74	アクノリッジ信号 ... 235
5 - 75	ビジィ信号, レディ信号 ... 236
5 - 76	RELT, CMDT, RELD, CMDD (マスタ) の動作 ... 242
5 - 77	RELT, CMDT, RELD, CMDD (スレーブ) の動作 ... 242
5 - 78	ACKTの動作 ... 242

図の目次 (4/5)

図番号	タイトル, ページ
5 - 79	ACKEの動作 ... 243
5 - 80	ACKDの動作 ... 244
5 - 81	BSYEの動作 ... 244
5 - 82	端子構成図 ... 247
5 - 83	マスタ・デバイスからスレーブ・デバイス (WUP = 1) へのアドレス送信動作 ... 249
5 - 84	マスタ・デバイスからスレーブ・デバイスへのコマンド送信動作 ... 250
5 - 85	マスタ・デバイスからスレーブ・デバイスへのデータ送信動作 ... 251
5 - 86	スレーブ・デバイスからマスタ・デバイスへのデータ送信動作 ... 252
5 - 87	シリアル・バス構成例 ... 255
5 - 88	READコマンドの転送フォーマット ... 257
5 - 89	WRITE, ENDコマンドの転送フォーマット ... 258
5 - 90	STOPコマンドの転送フォーマット ... 258
5 - 91	STATUSコマンドの転送フォーマット ... 259
5 - 92	STATUSコマンドのステータス・フォーマット ... 259
5 - 93	RESETコマンドの転送フォーマット ... 260
5 - 94	CHGMSTコマンドの転送フォーマット ... 260
5 - 95	エラー発生時のマスタ, スレーブの動作 ... 261
5 - 96	$\overline{SCK}/P01$ 端子の構成 ... 262
5 - 97	LCDコントローラ/ドライバのブロック図 ... 264
5 - 98	表示モード・レジスタのフォーマット ... 266
5 - 99	表示制御レジスタのフォーマット ... 267
5 - 100	LCD/ポート選択レジスタのフォーマット ... 269
5 - 101	データ・メモリ・マップ ... 270
5 - 102	表示データ・メモリとコモン・セグメント対応 ... 271
5 - 103	コモン信号波形 (スタティック) ... 274
5 - 104	コモン信号波形 (1/2バイアス法) ... 274
5 - 105	コモン信号波形 (1/3バイアス法) ... 274
5 - 106	コモン信号とセグメント信号の電位と位相 ... 275
5 - 107	LCD駆動用電源の接続例 (分割抵抗内蔵時) ... 277
5 - 108	LCD駆動用電源の接続例 (分割抵抗外付け時) ... 278
5 - 109	スタティック方式LCDの表示パターンと電極結線 ... 279
5 - 110	スタティックLCDパネルの結線例 ... 280
5 - 111	スタティックLCD駆動波形例 ... 281
5 - 112	2時分割方式LCDの表示パターンと電極結線 ... 282

図の目次 (5/5)

図番号	タイトル, ページ
5 - 113	2時分割LCDパネルの結線例 ... 283
5 - 114	2時分割LCD駆動波形例 (1/2バイアス法) ... 284
5 - 115	3時分割方式LCDの表示パターンと電極結線 ... 285
5 - 116	3時分割LCDパネルの結線例 ... 286
5 - 117	3時分割LCD駆動波形例 (1/2バイアス法) ... 287
5 - 118	3時分割LCD駆動波形例 (1/3バイアス法) ... 288
5 - 119	4時分割方式LCDの表示パターンと電極結線 ... 289
5 - 120	4時分割LCDパネルの結線例 ... 290
5 - 121	4時分割LCD駆動用波形例 (1/3バイアス法) ... 291
5 - 122	ビット・シーケンシャル・パツファのフォーマット ... 292
6 - 1	割り込み制御回路ブロック図 ... 296
6 - 2	割り込みベクタ・テーブル ... 298
6 - 3	割り込みプライオリティ選択レジスタ ... 301
6 - 4	INT0, INT4の構成 ... 303
6 - 5	ノイズ除去回路の入出力タイミング ... 304
6 - 6	INT0エッジ検出モード・レジスタ (IM0) のフォーマット ... 305
6 - 7	割り込み処理シーケンス ... 307
6 - 8	高位の割り込みによる多重割り込み ... 308
6 - 9	割り込みステータス・フラグの変更による多重割り込み ... 309
6 - 10	KR0-KR3のブロック図 ... 324
6 - 11	INT2エッジ検出モード・レジスタ (IM2) のフォーマット ... 325
7 - 1	スタンバイ・モードの解除動作 ... 331
7 - 2	STOPモード解除時のウェイト時間 ... 333
8 - 1	リセット機能の構成 ... 337
8 - 2	$\overline{\text{RESET}}$ 信号発生によるリセット動作 ... 337

表の目次 (1/2)

表番号	タイトル, ページ
2 - 1	デジタル入出力ポートの端子機能一覧 ... 31
2 - 2	ポート端子以外の端子機能一覧 ... 33
2 - 3	未使用端子の処理一覧 ... 45
3 - 1	アドレッシング・モード ... 51
3 - 2	RBE, RBSと選択されるレジスタ・バンク ... 63
3 - 3	通常ルーチンと割り込みルーチンでのレジスタ・バンクの使い分けの例 ... 64
3 - 4	周辺ハードウェア操作時に適用可能なアドレッシング・モード ... 68
4 - 1	Mk モードとMk モードの違い ... 75
4 - 2	SBSで選択されるスタック・エリア ... 89
4 - 3	スタック動作時に退避 / 復帰されるPSW のフラグ ... 93
4 - 4	キャリー・フラグ操作命令 ... 94
4 - 5	割り込みステータス・フラグの指示内容 ... 95
4 - 6	RBE, RBSと選択されるレジスタ・バンク ... 97
5 - 1	デジタル・ポートの種類と特徴 ... 100
5 - 2	入出力端子操作命令一覧表 ... 109
5 - 3	入出力ポート操作時の動作 ... 111
5 - 4	プルアップ抵抗内蔵の指定法 ... 112
5 - 5	CPUクロックの切り替えに要する最大時間 ... 123
5 - 6	使用可能モード一覧表 ... 140
5 - 7	分解能と最長設定時間 (8 ビット・タイマ時) ... 155
5 - 8	分解能と最長設定時間 (16ビット・タイマ時) ... 171
5 - 9	シリアル・クロックの選択と応用 (3 線式シリアル/Oモード時) ... 216
5 - 10	シリアル・クロックの選択と応用 (2 線式シリアル/Oモード時) ... 226
5 - 11	シリアル・クロックの選択と応用 (SBIモード時) ... 241
5 - 12	SBIモードにおける各種の信号 ... 245
5 - 13	最大表示画素 ... 265
5 - 14	COM信号 ... 272
5 - 15	LCD駆動電圧 (スタティック) ... 273
5 - 16	LCD駆動電圧 (1/2バイアス法) ... 273
5 - 17	LCD駆動電圧 (1/3 バイアス法) ... 273
5 - 18	LCD駆動用電源の供給値 ... 276

表の目次 (2/2)

表番号	タイトル, ページ
5 - 19	S12-S19端子の選択, 非選択電圧 (スタティック表示) ... 279
5 - 20	S12-S15端子の選択, 非選択電圧 (2時分割表示例) ... 282
5 - 21	S12-S14端子の選択, 非選択電圧 (3時分割表示例) ... 285
5 - 22	S14, S15端子の選択, 非選択電圧 (4時分割表示例) ... 289
6 - 1	割り込みソースの種類 ... 297
6 - 2	割り込み要求フラグのセット信号 ... 300
6 - 3	IST1, IST0と割り込み処理状態 ... 306
6 - 4	共用割り込みの判別 ... 310
6 - 5	テスト・ソースの種類 ... 322
6 - 6	テスト要求フラグのセット信号 ... 322
7 - 1	スタンバイ・モード時の各動作状態 ... 329
7 - 2	BTMによるウェイト時間の選択 ... 333
8 - 1	各ハードウェアのリセット後の状態 ... 338
9 - 1	プログラム・メモリ書き込み/ベリファイ時の使用端子 ... 341
9 - 2	動作モード ... 342
10 - 1	端子のマスク・オプションの選択 ... 347
11 - 1	ビット操作アドレッシングの種類と指定可能な範囲 ... 350

第1章 概 説

μ PD753204, 753206, 753208, 75P3216は、豊富な製品展開を誇る75Xシリーズの後継品種75XLシリーズの4ビット・シングルチップ・マイクロコンピュータです。この μ PD753204, 753206, 753208, 75P3216を総称して μ PD753208サブシリーズといたします。

μ PD753208は、従来品である μ PD75308Bに比べて、ROM容量を拡大、CPU機能を拡張、低電圧1.8 Vでの高速動作を可能にしたLCDコントローラ/ドライバ内蔵製品です。

小型のプラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ) を用意しており、LCDパネルを使用する小型のセットに最適です。

以下に特徴を示します。

- ・低電圧動作可能: $V_{DD} = 1.8 \sim 5.5$ V
- ・高速動作とパワー・セーブに有利な命令実行時間可変機能
0.95 μ s, 1.91 μ s, 3.81 μ s, 15.3 μ s (4.19 MHz動作時)
0.67 μ s, 1.33 μ s, 2.67 μ s, 10.7 μ s (6.0 MHz動作時)
- ・タイマを5チャンネル内蔵
- ・プログラマブルなLCDコントローラ/ドライバ内蔵
- ・小型パッケージを採用 (48ピン・プラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ))

μ PD75P3216は、電氣的書き込み可能なワン・タイムPROMを備えた製品で、 μ PD753204, 753206, 753208とピン・コンパチブルです。システム開発時の試作用、または少量生産用に適しています。

応用分野

- ・リモコン
- ・CDラジカセ
- ・カメラ
- ・血圧計
- ・ガス・メータ など

備考 このマニュアルは、特に機能面での違いのないかぎり μ PD753208を代表製品として記述してあります。 μ PD753204, 753206, 75P3216のマニュアルとして使用する場合は、 μ PD753208をそれぞれの製品に読み替えて使用してください。

1.1 機能概要

(1/2)

項 目		機 能	
命令実行時間		<ul style="list-style-type: none"> ・ 0.95, 1.91, 3.81, 15.3 μs (システム・クロック : 4.19 MHz動作時) ・ 0.67, 1.33, 2.67, 10.7 μs (" : 6.0 MHz動作時) 	
内蔵メモリ	ROM	4096 \times 8 ビット (μ PD753204)	
		6144 \times 8 ビット (μ PD753206)	
		8192 \times 8 ビット (μ PD753208)	
		16384 \times 8 ビット (μ PD75P3216)	
	RAM	512 \times 4 ビット	
汎用レジスタ		<ul style="list-style-type: none"> ・ 4 ビット操作時 : 8 個 \times 4 バンク ・ 8 ビット操作時 : 4 個 \times 4 バンク 	
★ 入出力 ポート	CMOS入力	6本	ソフトウェアによる内蔵プルアップ抵抗の接続を指定可能 : 5本
	CMOS入出力	20本	ソフトウェアによる内蔵プルアップ抵抗の接続を指定可能 : 20本 セグメント端子と兼用 : 8本
	N-chオープン・ドレイン入出力	4本	13V耐圧。マスク・オプションによるプルアップ抵抗内蔵可能 ^{注1}
	合計	30本	
LCDコントローラ/ドライバ		<ul style="list-style-type: none"> ・ セグメント数選択 : 4/8/12セグメント (4本単位でCMOS入出力ポートに切り替え可能, 最大8本) ・ 表示モード選択 : スタティック, 1/2デューティ (1/2バイアス), 1/3デューティ (1/2バイアス), 1/3デューティ (1/3バイアス), 1/4デューティ (1/3バイアス) ・ LCD 駆動用分割抵抗をマスク・オプションで内蔵可能^{注2} 	
タイマ		5チャンネル <ul style="list-style-type: none"> ・ 8ビット・タイマ/イベント・カウンタ : 1チャンネル ・ 8ビット・タイマ・カウンタ : 2チャンネル (16ビット・タイマ・カウンタ, キャリア・ジェネレータ, ゲート付きタイマとして使用可能) ・ ベーシック・インターバル・タイマ/ウォッチドッグ・タイマ : 1チャンネル ・ 時計用タイマ : 1チャンネル 	
シリアル・インタフェース		<ul style="list-style-type: none"> ・ 3線式シリアルI/Oモード...MSB/LSB先頭切り替え ・ 2線式シリアルI/Oモード ・ SBIモード 	
ビット・シーケンシャル・バッファ (BSB)		16ビット	
クロック出力 (PCL)		<ul style="list-style-type: none"> ・ , 524, 262, 65.5 kHz (システム・クロック : 4.19 MHz動作時) ・ , 750, 375, 93.8 kHz (" : 6.0 MHz動作時) 	
ブザー出力 (BUZ)		<ul style="list-style-type: none"> ・ 2, 4, 32 kHz (システム・クロック : 4.19 MHz動作時) ・ 2.93, 5.86, 46.9 kHz (システム・クロック : 6.0 MHz動作時) 	

注1 . μ PD75P3216にはマスク・オプションによるプルアップ抵抗の内蔵機能はありません。

2 . μ PD75P3216は, マスク・オプションによる分割抵抗を内蔵していません。

(2/2)

項 目	機 能
	外部：2本，内部：5本
ベクタ割り込み	外部：1本，内部：1本
テスト入力	システム・クロック発振用セラミック/クリスタル発振回路
システム・クロック発振回路	STOPモード/HALTモード
スタンバイ機能	$V_{DD} = 1.8 \sim 5.5 \text{ V}$
電源電圧	48ピン・プラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ)

パッケージ

1.2 オーダ情報

オーダ情報	パッケージ	内蔵ROM
μPD753204GT-xxx	48ピン・プラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ)	マスクROM
μPD753206GT-xxx	"	"
μPD753208GT-xxx	"	"
μPD75P3216GT	"	ワン・タイムPROM

備考 xxxはROMコード番号です。

1.3 μPD753208サブシリーズ製品間の違い

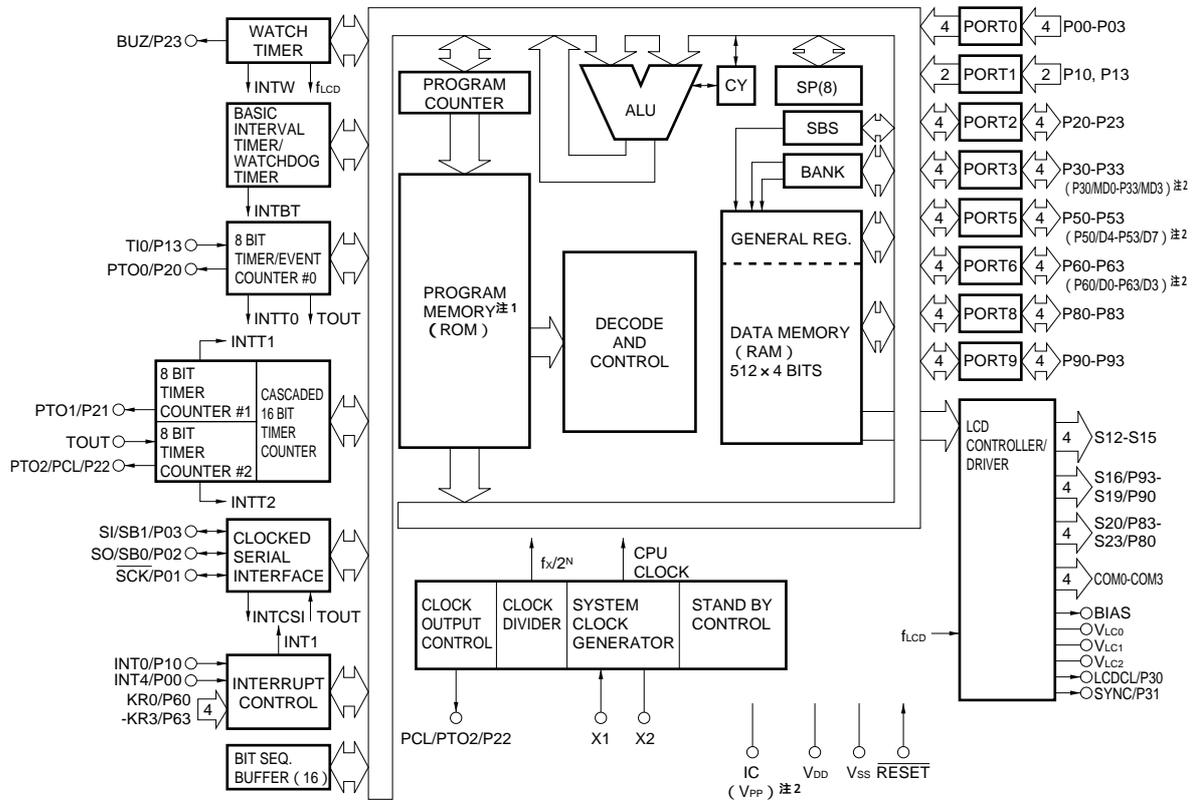
項 目		μPD753204	μPD753206	μPD753208	μPD75P3216
プログラム・カウンタ		12ビット	13ビット		14ビット
プログラム・メモリ (バイト)		マスクROM 4096	マスクROM 6144	マスクROM 8192	ワン・タイムPROM 16384
データ・メモリ (x4ビット)		512			
マスク・オプション	ポート5のプルアップ抵抗	あり (内蔵する / しないの指定可能)			なし (内蔵不可)
	RESET時のウェイト時間	あり (2 ¹⁷ /fx, 2 ¹⁵ /fxの2つから選択可能) ^注			なし (2 ¹⁵ /fx固定) ^注
	LCD駆動用分割抵抗	あり (内蔵する / しないの指定可能)			なし (内蔵不可)
端子接続	9-12番ピン	P30-P33			P30/MD0-P33/MD3
	14-17番ピン	P50-P53			P50/D4-P53/D7
	18-21番ピン	P60/KR0-P63/KR3			P60/KR0/D0-P63/KR3/D3
	25番ピン	IC			V _{PP}
その他		回路の規模やマスク・レイアウトが異なるため、ノイズ耐量、ノイズ輻射などが異なります。			

注 2¹⁷/fxは、6.0 MHz動作時：21.8 ms, 4.19 MHz動作時：31.3 msです。

2¹⁵/fxは、6.0 MHz動作時：5.46 ms, 4.19 MHz動作時：7.81 msです。

注意 PROMとマスクROMでは、ノイズ耐量やノイズ輻射が異なります。試作から量産の過程でPROM品からマスクROM品への置き換えを検討される場合は、マスクROM品のCS品 (ES品ではなく) で十分な評価を行ってください。

1.4 ブロック図



注1 . ROMの容量は、品種により異なります。

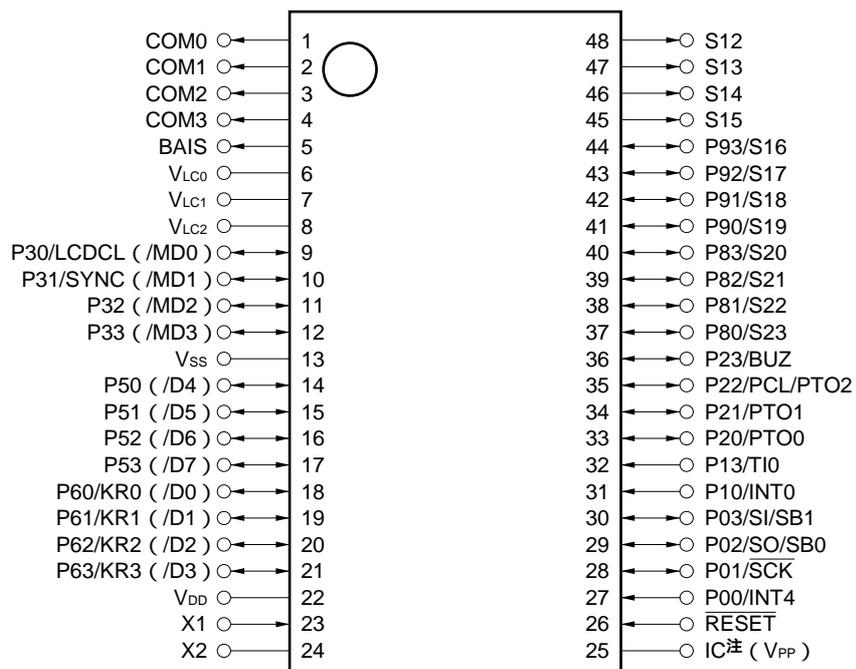
2 . () 内は、μPD75P3216の場合です。

1.5 端子接続図 (Top View)

・48ピン・プラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ)

μ PD753204GT-x x x , μ PD753206GT-x x x ,

μ PD753208GT-x x x , μ PD75P3216GT



注 IC (Internally Connected) 端子は, V_{Dd}に直接接続してください。

備考 ()内は, μ PD75P3216の場合です。

端子名称

BIAS	: LCD Power Supply Bias Control	P90-P93	: Port9
BUZ	: Buzzer Clock	PCL	: Programmable Clock
COM0-COM3	: Common Output 0-3	PTO0-PTO2	: Programmable Timer Output 0-2
D0-D7	: Data Bus 0-7	$\overline{\text{RESET}}$: Reset
IC	: Internally Connected	S12-S23	: Segment Output 12-23
INT0, INT4	: External Vectored Interrupt 0, 4	SB0, SB1	: Serial Data Bus 0, 1
KR0-KR3	: Key Return 0-3	$\overline{\text{SCK}}$: Serial Clock
LCDCL	: LCD Clock	SI	: Serial Input
MD0-MD3	: Mode selection 0-3	SO	: Serial Output
P00-P03	: Port0	SYNC	: LCD Synchronization
P10, P13	: Port1	TI0	: Timer Input 0
P20-P23	: Port2	V _{DD}	: Positive Power Supply
P30-P33	: Port3	V _{LC0-V_{LC2}}	: LCD Power Supply 0-2
P50-P53	: Port5	V _{PP}	: Programming Power Supply
P60-P63	: Port6	V _{SS}	: Ground
P80-P83	: Port8	X1, X2	: System Clock Oscillation 1, 2

〔メモ〕

第2章 端子機能

2.1 μPD753208の端子機能一覧

表2-1 デジタル入出力ポートの端子機能一覧 (1/2)

端子名称	入出力	兼用端子	機能	8ビット I/O	リセット時	入出力回路 TYPE ^{注1}
P00	入力	INT4	4ビット入力ポート (PORT0)。 P01-P03は3ビット単位で、ソフトウェアにより内蔵プルアップ抵抗の接続を指定可能。	×	入力	Ⓑ
P01	入出力	$\overline{\text{SCK}}$				Ⓕ-A
P02	入出力	SO/SB0				Ⓕ-B
P03	入出力	SI/SB1				Ⓜ-C
P10	入力	INT0	2ビット入力ポート (PORT1)。 1ビット単位でのみ操作可能。 2ビット単位で、ソフトウェアにより内蔵プルアップ抵抗の接続を指定可能。 P10/INT0はノイズ除去回路を選択可能。	×	入力	Ⓑ-C
P13		TI0				
P20	入出力	PTO0	4ビット入出力ポート (PORT2)。 4ビット単位で、ソフトウェアにより内蔵プルアップ抵抗の接続を指定可能。	×	入力	E-B
P21		PTO1				
P22		PCL/PTO2				
P23		BUZ				
P30	入出力	LCDC $\overline{\text{L}}$ (MD0) ^{注2}	プログラマブル4ビット入出力ポート (PORT3)。 ビット単位で入力/出力指定可能。 4ビット単位でソフトウェアにより内蔵プルアップ抵抗の接続を指定可能。	×	入力	E-B
P31		SYNC (MD1) ^{注2}				
P32		(MD2) ^{注2}				
P33		(MD3) ^{注2}				
★ P50 ^{注3}	入出力	(D4) ^{注2}	N-chオープン・ドレイン4ビット入出力ポート (PORT5)。オープン・ドレイン時、13V耐圧。 ビット単位でプルアップ抵抗を内蔵可能 (マスク・オプション) ^{注4} 。	×	ハイ・レベル (プルアップ抵抗内蔵時) またはハイ・インピーダンス	M-D (M-E) ^{注2}
P51 ^{注3}		(D5) ^{注2}				
P52 ^{注3}		(D6) ^{注2}				
P53 ^{注3}		(D7) ^{注2}				

注1. 印はシュミット・トリガ入力を示します。

2. () 内はμPD75P3216の場合です。

3. マスク・オプションによるプルアップ抵抗を内蔵しない場合 (N-chオープン・ドレイン入力ポートとして使用している場合), 入力命令, ビット操作命令を行ったときにロウ・レベル入力リーク電流が増加します。

4. μPD75P3216にはマスク・オプションによるプルアップ抵抗の内蔵機能はありません。

表2-1 デジタル入出力ポートの端子機能一覧(2/2)

端子名称	入出力	兼用端子	機 能	8 ビット I/O	リセット時	入出力 回 路 TYPE ^{注1}
P60	入出力	KR α /D0 ^{注2}	プログラマブル4ビット入出力ポート (PORT6)。 ビット単位で入力 / 出力指定可能。 4ビット単位でソフトウェアにより内蔵プルアップ 抵抗の接続を指定可能。	×	入 力	Ⓔ-A
P61		KR1 κ /D1 ^{注2}				
P62		KR χ /D2 ^{注2}				
P63		KR ξ /D3 ^{注2}				
P80	入出力	S23	4ビット入出力ポート (PORT8)。 4ビット単位で、ソフトウェアにより内蔵プルアッ プ抵抗の接続を指定可能 ^{注3} 。		入 力	H
P81		S22				
P82		S21				
P83		S20				
P90	入出力	S19	4ビット入出力ポート (PORT9)。 4ビット単位で、ソフトウェアにより内蔵プルアッ プ抵抗の接続を指定可能 ^{注3} 。		入 力	H
P91		S18				
P92		S17				
P93		S16				

注1 . 印はシュミット・トリガ入力を示します。

2 . () 内は μ PD75P3216の場合です。

3 . セグメント信号出力として使用する場合は、ソフトウェアにより内蔵プルアップ抵抗の接続を指定しないでください。

表2-2 ポート端子以外の端子機能一覧(1/2)

端子名称	入出力	兼用端子	機 能	リセット時	入出力回路 TYPE ^{注1}
TI0	入 力	P13	タイマ/イベント・カウンタへの外部イベント・パルス入力。	入 力	Ⓑ-C
PTO0	出 力	P20	タイマ/イベント・カウンタ出力。	入 力	E-B
PTO1		P21	タイマ・カウンタ出力。		
PTO2		P22/PCL			
PCL		P22/PTO2	クロック出力。		
BUZ		P23	任意の周波数出力(ブザー用またはシステム・クロックのトリミング用)。		
SCK	入出力	P01	シリアル・クロック入出力。	入 力	Ⓕ-A
SO/SB0		P02	シリアル・データ出力。 シリアル・データ・バス入出力。		Ⓕ-B
SI/SB1		P03	シリアル・データ入力。 シリアル・データ・バス入出力。		Ⓜ-C
INT4	入 力	P00	エッジ検出ベクタ割り込み入力(立ち上がりおよび立ち下がり の両エッジ検出)。	入 力	Ⓑ
INT0	入 力	P10	エッジ検出ベクタ割り込み入力(検出エッジ 選択可能)。ノイズ除去回路を選択可能。 ノイズ除去回路付き/非同期選択可能。	入 力	Ⓑ-C
KR0-KR3	入 力	P60-P63	立ち下がりエッジ検出テスト入力。	入 力	Ⓕ-A
S12-S15	出 力	-	セグメント信号出力。	注2	G-A
S16-S19	出 力	P93-P90	セグメント信号出力。	入 力	H
S20-S23	出 力	P83-P80	セグメント信号出力。	入 力	H
COM0-COM3	出 力	-	コモン信号出力。	注2	G-B

注1. 印はシュミット・トリガ入力を示します。

2. 各表示出力は、次に示すV_{LCx}を入力ソースとして選択します。

S12-S15 : V_{LC1}, COM0-COM2 : V_{LC2}, COM3 : V_{LC0}

表2-2 ポート端子以外の端子機能一覧(2/2)

端子名称	入出力	兼用端子	機能	リセット時	入出力回路 TYPE ^{注1}
V _{LC0} -V _{LC2}	-	-	LCD駆動用電源。 分割抵抗内蔵可能(マスク・オプション) ^{注2} 。	-	-
BIAS	出力	-	外付け分割抵抗カット用出力。	注3	-
LCDCL ^{注4}	出力	P30	外部拡張ドライバ駆動用クロック出力。	入力	E-B
SYNC ^{注4}	出力	P31	外部拡張ドライバ同期用クロック出力。	入力	E-B
X1	入力	-	システム・クロック発振用クリスタル/セラミック接続端子。外部クロックの場合、X1へ入力しX2へその逆相を入力。	-	-
X2	-				
RESET	入力	-	システム・リセット入力(ロウ・レベル・アクティブ)。	-	Ⓑ
★ MD0-MD3	入力	P30-P33	μPD75P3216のみ内蔵。 プログラム・メモリ(PROM)書き込み/ベリファイ時のモード選択。	入力	E-B
★ D0-D3	入出力	P60/KR0- P63/KR3	μPD75P3216のみ内蔵。 プログラム・メモリ(PROM)書き込み/ベリファイ時のデータ・バス端子。	入力	Ⓕ-A
★ D4-D7		P50-P53		ハイ・インピーダンス	M-E
IC	-	-	Internally Connected。V _{DD} に直接接続してください。	-	-
V _{PP}	-	-	μPD75P3216のみ内蔵。 プログラム・メモリ(PROM)書き込み/ベリファイ時のプログラム電圧印加。 通常動作時はV _{DD} と直接接続。 PROM書き込み/ベリファイ時は+12.5Vを印加。	-	-
V _{DD}	-	-	正電源。	-	-
V _{SS}	-	-	グランド電位。	-	-

注1 . 印はシュミット・トリガ入力を示します。

2 . μPD75P3216は、マスク・オプションによる分割抵抗を内蔵していません。

3 . 分割抵抗を内蔵している場合.....ロウ・レベル

分割抵抗を内蔵していない場合.....ハイ・インピーダンス

4 . 将来のシステム拡張に備えた端子です。現在はP30, P31端子としてのみ使用します。

2.2 端子機能の説明

2.2.1 P00-P03 (PORT0) ...INT4, \overline{SCK} , SO/SB0, SI/SB1兼用入力 P10, P13 (PORT1) ...INT0, TI0兼用入力

ポート0は4ビットの入力ポート、ポート1は2ビットの入力ポートです。

ポート0, 1は、入力ポートとしての機能のほかに、次のような機能を兼用しています。

ポート0：ベクタ割り込み入力 (INT4)

シリアル・インタフェース入出力 (\overline{SCK} , SO/SB0, SI/SB1)

ポート1：ベクタ割り込み入力 (INT0)

タイマ/イベント・カウンタへの外部イベント・パルス入力 (TI0)

ポート0は、シリアル・インタフェース機能使用時は、動作モードによって兼用端子が出力機能を持ちます。

ポート0はビット単位または4ビット単位で、ポート1はビット単位でのみ操作できます。

ポート0およびポート1の各端子は、ノイズによる誤動作を防ぐためにシュミット・トリガ入力になっています。さらに、P10はノイズ除去回路を選択できます (詳細は6.3(3)INT0, INT4のハードウェアを参照してください)。

ポート0は3ビット単位 (P01-P03) で、ポート1は2ビット単位 (P10, P13) で、ソフトウェアにより内蔵プルアップ抵抗の接続を指定することができます。これらは、プルアップ抵抗指定レジスタ・グループA (POGA) を操作して指定します。

\overline{RESET} 信号発生により、入力モードになります。

2.2.2 P20-P23 (PORT2) ...PTO0-PTO2, PCL, BUZ兼用入出力 P30-P33 (PORT3) ...LCDCL, SYNC, MD0-MD3^注兼用 入出力

- ★ P50-P53 (PORT5) ...N-chオープン・ドレイン中耐圧 (13V),
D4-D7^注兼用入出力
- ★ P60-P63 (PORT6) ...KR0-KR3, D0-D3^注兼用入出力
P80-P83 (PORT8),
P90-P93 (PORT9) ...S23-S20, S19-S16兼用入出力

出力ラッチ付き4ビット入出力ポートです。

入出力ポートの機能のほかに、次のような機能を兼用しています。

- ポート2 : タイマ/イベント・カウンタ出力 (PTO0-PTO2)
クロック出力 (PCL)
任意の周波数出力 (BUZ)
- ポート3 : LCD外部拡張ドライバ駆動用クロック (LCDCL)
LCD外部拡張ドライバ同期用クロック (SYNC)
- ★ プログラム・メモリ (PROM) 書き込み/ベリファイ時のモード選択 (MD0-MD3)^注
- ★ ポート5 : プログラム・メモリ (PROM) 書き込み/ベリファイ時のデータ・バス (D4-D7)^注
- ポート6 : キー割り込み入力 (KR0-KR3)
- ★ プログラム・メモリ (PROM) 書き込み/ベリファイ時のデータ・バス (D0-D3)^注
- ポート8, 9 : セグメント信号出力 (S23-S20, S19-S16)

注 μ PD75P3216のみ兼用している機能です。

ポート5はN-chオープン・ドレイン中耐圧 (13V) です。

おのこのポートの入出力のモード選択はポート・モード・レジスタにより設定します。ポート2, 5, 8, 9は4ビット単位で、ポート3, 6はビット単位で指定できます。

また、ポート2, 3, 6, 8, 9は、ソフトウェアにより内蔵プルアップ抵抗の接続を4ビット単位で指定可能で、プルアップ抵抗指定レジスタ・グループA, B (POGA, POGB) を操作して行います。ポート5はマスク・オプションにより、ビット単位でプルアップ抵抗内蔵指定ができます。ただし、 μ PD75P3216にはマスク・オプションによるプルアップ抵抗の内蔵機能はありません。

ポート8, 9はペアで8ビット単位の入出力ができます。 $\overline{\text{RESET}}$ 信号発生により、ポート2, 3, 6, 8, 9は入力モード (ハイ・インピーダンス) に、ポート5はハイ・レベル (マスク・オプションのプルアップ抵抗内蔵時) またはハイ・インピーダンスになります。

2.2.3 TIO...ポート1兼用入力

タイマ/イベント・カウンタ0の外部イベント・パルス入力端子です。

タイマ/イベント・カウンタ・モード・レジスタ(TM0)で、カウント・パルス(CP)に外部イベント・パルス入力を選択することにより用いることができます。

TIOはシュミット・トリガ入力になっています。

詳細は5.5.1(1)タイマ/イベント・カウンタ・モード・レジスタ(TM0, TM1, TM2)を参照してください。

2.2.4 PTO0-PTO2...ポート2兼用出力

タイマ/イベント・カウンタ0, タイマ・カウンタ1, 2の出力端子です。方形波パルスを出力します。タイマ/イベント・カウンタまたはタイマ・カウンタの信号を出力する場合は、出力ラッチを“0”にクリアし、ポート・モード・レジスタのポート2のビットを出力モード“1”にセットします。

タイマ・スタート命令により、出力は“0”にクリアされます。

詳細は5.5.2(3)タイマ/イベント・カウンタ動作を参照してください。

2.2.5 PCL...ポート2兼用出力

プログラマブル・クロック出力端子です。周辺LSI(スレーブのマイクロコンピュータなど)にクロックを供給するときに使用します。 $\overline{\text{RESET}}$ 信号発生により、クロック出力モード・レジスタ(CLOM)が“0”にクリアされクロックは出力禁止状態になり、通常のポートとしての動作モードになります。

詳細は5.2.4 クロック出力回路を参照してください。

2.2.6 BUZ...ポート2兼用出力

周波数出力端子です。任意の周波数(2 kHz, 4 kHz, 32 kHz:メイン・システム・クロック4.19 MHz動作時, またはサブシステム・クロック32.768 kHz動作時)を出力することにより、ブザー音発生やシステム・クロック発振周波数のトリミングに使用します。P23端子と兼用となっており、時計モード・レジスタ(WM)のビット7(WM7)が“1”にセットされているときのみ有効となります。

$\overline{\text{RESET}}$ 信号発生により、WM7は“0”にクリアされ、通常のポートとしての動作モードになります。

詳細は5.4.2 時計モード・レジスタを参照してください。

2.2.7 $\overline{\text{SCK}}$, SO/SB0, SI/SB1...ポート0兼用3ステート入出力

シリアル・インタフェースのための入出力端子です。シリアル動作モード・レジスタ(CSIM)の設定に従って動作します。3線式シリアルI/Oモードを選択した場合、 $\overline{\text{SCK}}$ はCMOS入出力, SOはCMOS出力, SIはCMOS入力として機能します。また, 2線式シリアルI/Oモード, SBIモードを選択した場合, $\overline{\text{SCK}}$ はCMOS入出力, SB1(SB0)はN-chオープン・ドレイン入出力として機能します。

$\overline{\text{RESET}}$ 信号発生によって、シリアル・インタフェース動作は停止し、入力ポートとなります。

いずれの端子とも、シュミット・トリガ入力になっています。

詳細は5.6 シリアル・インタフェースを参照してください。

2.2.8 INT4...ポート0 兼用入力

立ち上がりおよび立ち下がり両エッジ・アクティブの外部ベクタ割り込み入力端子です。この端子に入力された信号がロウ・レベルからハイ・レベルに変化したときと、ハイ・レベルからロウ・レベルに変化したときに割り込み要求フラグがセットされます。

INT4は非同期入力であり、CPUの動作クロックに関係なく一定のハイ・レベル幅またはロウ・レベル幅を持つ信号が入力されると受け付けられます。

INT4はSTOPモード、HALTモードの解除にも利用できます。また、シュミット・トリガ入力になっています。

2.2.9 INT0...ポート1 兼用入力

エッジ検出ベクタ割り込み入力端子で、ノイズ除去回路を選択できます。エッジ検出モード・レジスタ(IM0)により、検出エッジの選択ができます。

・INT0 (IM0のビット0, 1)

- (a) 立ち上がりエッジ・アクティブ
- (b) 立ち下がりエッジ・アクティブ
- (c) 立ち上がりおよび立ち下がり両エッジ・アクティブ
- (d) 外部割り込み信号入力禁止

INT0は非同期入力であり、CPUの動作クロックに関係なく一定のハイ・レベル幅を持つ入力があれば受け付けられます。また、ソフトウェアにより、ノイズ除去回路付きにすることができ、ノイズ除去を行うサンプリング・クロックを2段階に変えることができます。この場合、CPU動作クロックによって受け付けられる信号の幅は異なります。

$\overline{\text{RESET}}$ 信号発生により、IM0は“0”にクリアされ、立ち上がりエッジ・アクティブが選択されます。

STOPモード、HALTモードの解除に利用できます。ただし、ノイズ除去回路選択時には、STOPモード、HALTモードの解除はできません。

INT0はシュミット・トリガ入力となっています。

2.2.10 KR0-KR3...ポート6 兼用入力

キー割り込み入力端子です。KR0-KR3は、並列立ち下がりエッジ検出割り込み入力端子です。

エッジ検出モード・レジスタ(IM2)により、割り込みソースを“KR2とKR3”または“KR0-KR3”の2つから選択できます。

$\overline{\text{RESET}}$ 信号発生により、ポート6の入力モードになります。

2.2.11 S12-S23...出力

LCDのセグメント端子(前面電極)を直接駆動可能なセグメント信号出力端子でスタティック、1/2バイアス法の2または3時分割、1/3バイアス法の3または4時分割駆動を行います。

S16-S19, S20-S23はポート9, 8と兼用しており, 表示モード・レジスタ (LCDM) により切り替えて使用します。

2.2.12 COM0-COM3...出力

LCDのコモン端子 (背面電極) を直接駆動可能なコモン信号出力端子です。スタティック (COM0, 1, 2, 3 出力), 1/2バイアス法の2時分割駆動時 (COM0, 1 出力) または3時分割駆動時 (COM0, 1, 2 出力), 1/3バイアス法の3時分割駆動時 (COM0, 1, 2 出力) または4時分割駆動時 (COM0, 1, 2, 3 出力) にコモン信号をそれぞれ出力します。

2.2.13 VLCO-VLC2

LCD駆動用電源供給端子です。μPD753208はマスク・オプションにより, VLCO-VLC2端子の内部に, 各バイアス法に応じたLCD駆動用電源を外付け分割抵抗なしで供給できるように分割抵抗を内蔵することができます。ただし, μPD75P3216にはマスク・オプションがなく, 分割抵抗は内蔵されていません。

2.2.14 BIAS

分割抵抗カット用出力端子です。各種のLCD駆動電圧に対応するためにVLCO端子と接続し, 抵抗分割比率を変えたり, VLCO-VLC2端子やVSS端子とともに外部に抵抗を接続し, LCD駆動電源電圧値の微調整を行うために使用します。

2.2.15 LCDCL

LCD外部拡張ドライバ駆動用クロック出力端子です。

2.2.16 SYNC

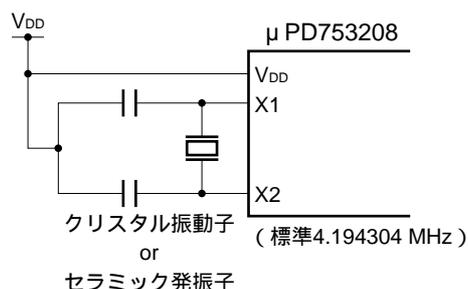
LCD外部拡張ドライバ同期用クロック出力端子です。

2.2.17 X1, X2

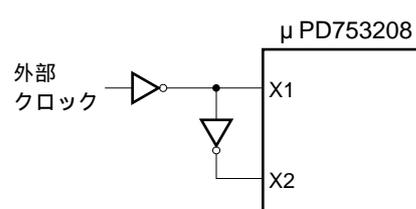
システム・クロック発振用クリスタル/セラミック接続端子です。

外部クロックを入力することも可能です。

(a) クリスタル/セラミック発振



(b) 外部クロック



2.2.18 $\overline{\text{RESET}}$

ロウ・レベル・アクティブのリセット入力端子です。

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

通常のCPUのイニシャライズ/スタートのほかにも、STOPモード、HALTモードの解除にも使用します。

$\overline{\text{RESET}}$ 入力はシュミット・トリガ入力になっています。

2.2.19 MD0-MD3 ($\mu\text{PD75P3216}$ のみ)

$\mu\text{PD75P3216}$ のみMD0-MD3を内蔵しています。

プログラム・メモリ(ワン・タイムPROM)への書き込み/ベリファイ時のモードの選択端子です。

★ 2.2.20 D0-D7 ($\mu\text{PD75P3216}$ のみ)

$\mu\text{PD75P3216}$ のみD0-D7を内蔵しています。

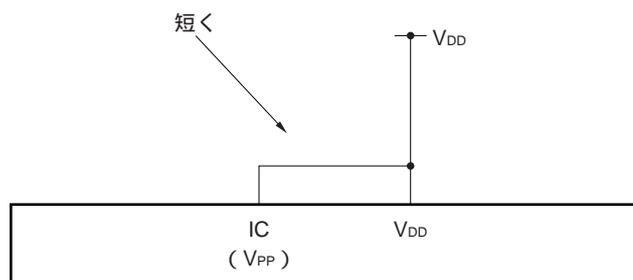
プログラム・メモリ(ワン・タイムPROM)への書き込み/ベリファイ時のデータ・バス端子です。

2.2.21 IC ($\mu\text{PD753204}$, 753206 , 753208 のみ)

IC (Internally Connected) 端子は、当社出荷時に $\mu\text{PD753208}$ 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。通常動作時には、IC端子は V_{DD} 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_{DD} 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_{DD} 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_{DD} 端子に直接接続してください。



2.2.22 V_{PP} ($\mu\text{PD75P3216}$ のみ)

プログラム・メモリ(ワン・タイムPROM)書き込み/ベリファイ時のプログラム電圧入力端子です。

通常動作時は V_{DD} に直接接続してください(上図参照)。PROMへの書き込み/ベリファイ時は12.5 Vを印加します。

2.2.23 VDD

正電源供給端子です。

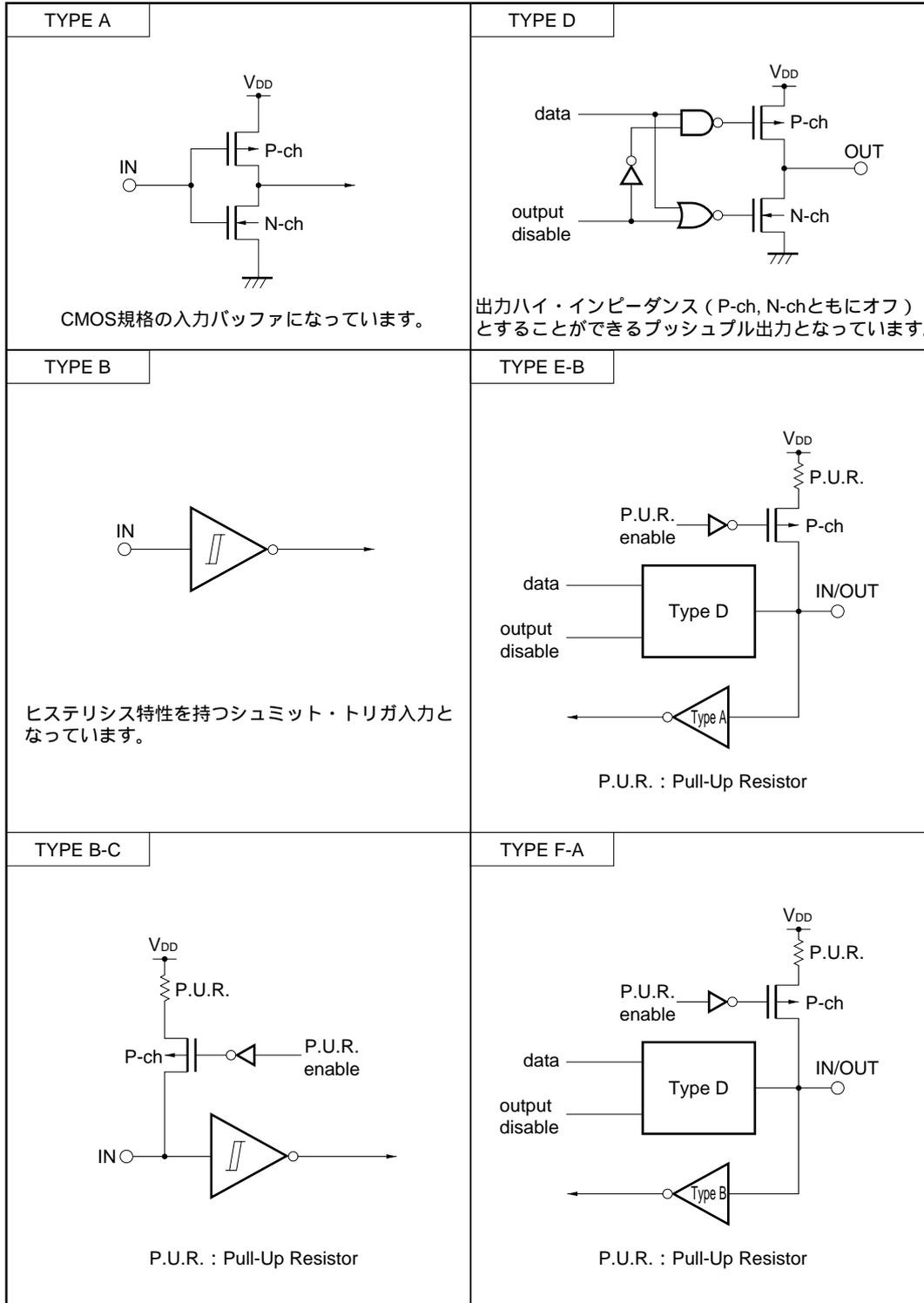
2.2.24 VSS

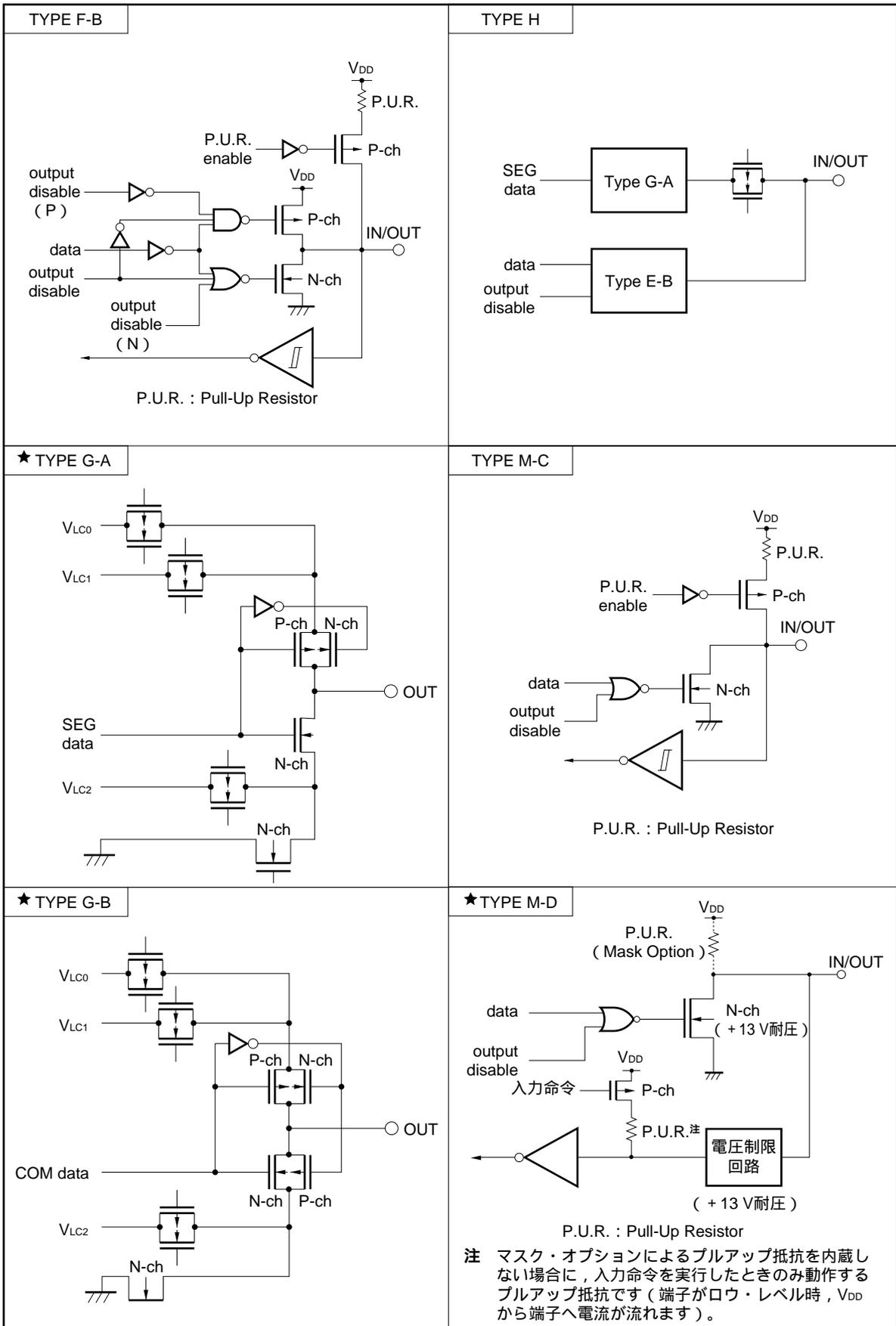
GND電位です。

2.3 端子の入出力回路

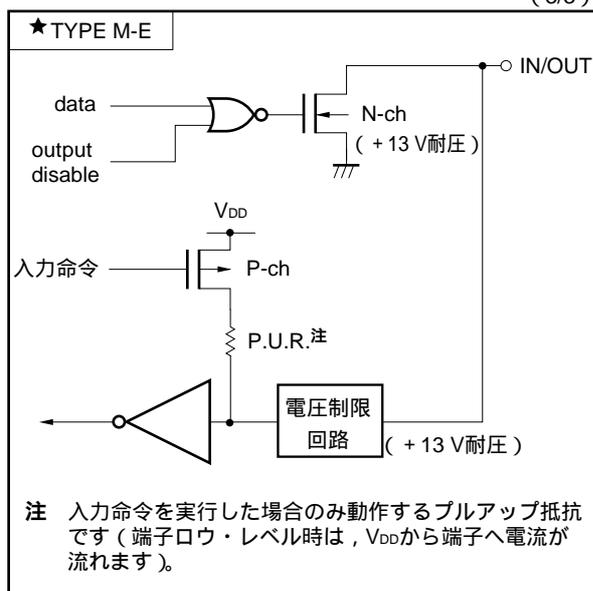
μPD753208の各端子の入出力回路を一部簡略した形式を用いて示します。

(1/3)





(3/3)



2.4 未使用端子の処理について

★

表2-3 未使用端子の処理一覧

端 子	推奨接続方法
P00/INT4	V _{SS} またはV _{DD} に接続してください。
P01/ $\overline{\text{SCK}}$	個別に抵抗を介して、V _{SS} またはV _{DD} に接続してくださ
P02/SO/SB0	い。
P03/SI/SB1	V _{SS} に接続してください。
P10/INT0	V _{SS} またはV _{DD} に接続してください。
P13/TI0	
P20/PTO0	入力状態：個別に抵抗を介して、V _{SS} またはV _{DD} に接続
P21/PTO1	してください。
P22/PCL/PTO2	出力状態：オープンにしてください。
P23/BUZ	
P30/LCDCL (/MD0) 注	
P31/SYNC (/MD1) 注	
P32 (/MD2) 注	
P33 (/MD3) 注	
P50 (/D4) 注	V _{SS} に接続してください (マスク・オプションのプル
P51 (/D5) 注	アップ抵抗を接続しないでください)。
P52 (/D6) 注	
P53 (/D7) 注	
P60/KR0 (/D0) 注	入力状態：個別に抵抗を介して、V _{SS} またはV _{DD} に接続
P61/KR1 (/D1) 注	してください。
P62/KR2 (/D2) 注	出力状態：オープンにしてください。
P63/KR3 (/D3) 注	
S12-S15	オープンにしてください。
COM0-COM3	
S16/P93-S19/P90	入力状態：個別に抵抗を介して、V _{SS} またはV _{DD} に接続
S20/P83-S23/P80	してください。 出力状態：オープンにしてください。
V _{LC0} -V _{LC2}	V _{SS} に接続してください。
BIAS	V _{LC0} -V _{LC2} のすべてが未使用のときのみV _{SS} に接続、そ 他の場合はオープンにしてください。
IC (V _{PP}) 注	V _{DD} に直接接続してください。

注 () 内はμPD75P3216のみの端子です。

(メモ)

第3章 アーキテクチャの特徴とメモリ・マップ

μPD753208に採用されている75XLのアーキテクチャは以下の特徴を持っています。

内蔵RAM最大4Kワード×4ビット(12ビット・アドレス)

周辺ハードウェアの拡張性

この優れた特徴を実現するため、以下の方式を採用しています。

(1) データ・メモリのバンク構成

(2) 汎用レジスタのバンク構成

(3) メモリ・マップI/O

本章では、これらのおのおのについて解説します。

3.1 データ・メモリのバンク構成とアドレッシング・モード

3.1.1 データ・メモリのバンク構成

μPD753208ではデータ・メモリ空間の000H-1FFH番地にスタティックRAMが内蔵されており、そのうち1ECH-1F7H番地の12×4ビットは表示データ・メモリとしても使用されます。また、周辺ハードウェア(入出力ポートやタイマなど)がF80H-FFFH番地に割り付けられています。

μPD753208は、12ビット・アドレス(4Kワード×4ビット)のデータ・メモリ空間をアドレッシングするために、下位8ビット・アドレスは命令によって直接あるいは間接に指定し、上位4ビット・アドレスをメモリ・バンクによって指定するメモリ・バンク構成となっています。

このメモリ・バンク(MB)を指定するために、以下のハードウェアが内蔵されています。

メモリ・バンク許可フラグ(MBE)

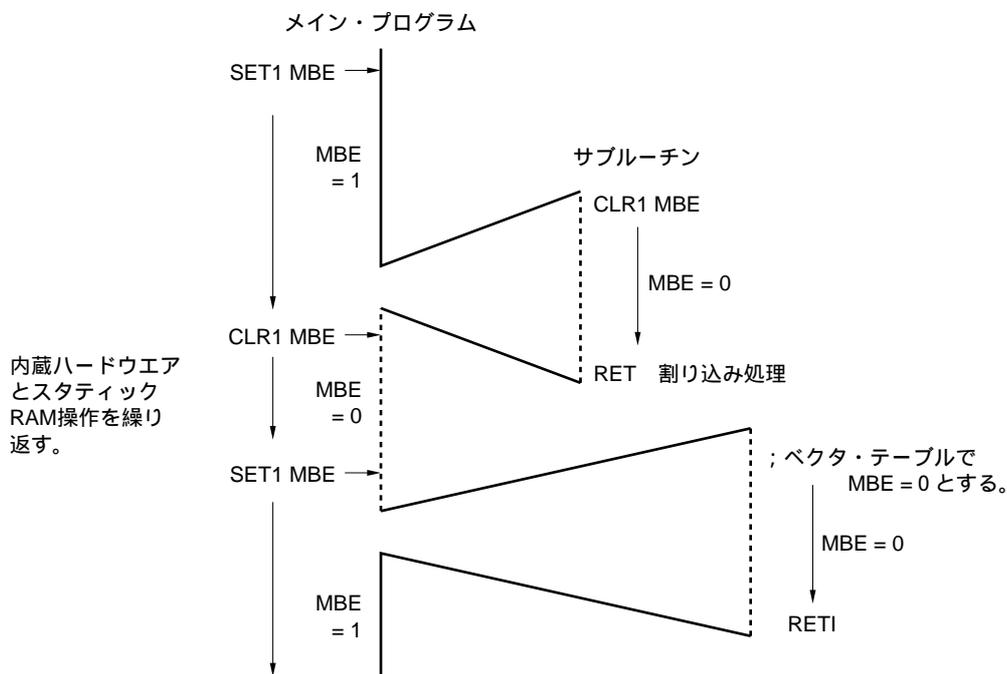
メモリ・バンク選択レジスタ(MBS)

MBSはメモリ・バンクを選択するレジスタで、メモリ・バンク0, 1, 15が設定できます。MBEはMBSで選択されたメモリ・バンクを有効とするか否かを決定するフラグです。図3-1に示すようにMBEが0の場合には指定されるメモリ・バンク(MB)はMBSにかかわらず固定されますが、MBEが1の場合にはMBSの設定によってメモリ・バンクを切り替え、データ・メモリ空間を拡張することができます。

データ・メモリ空間のアドレッシングにあたっては、通常はMBE=1とし、MBSで指定されたメモリ・バンクのデータ・メモリを操作しますが、プログラムの各処理でMBE=0のモードとMBE=1のモードを使い分けることにより、効率よくプログラムすることができます。

	適合するプログラムの処理	効果
MBE=0のモード	割り込み処理	MBSの退避/復帰が不要になる
	内蔵ハードウェア操作とスタティックRAM操作を繰り返す処理	MBSの変更が不要になる
	サブルーチン処理	MBSの退避/復帰が不要になる
MBE=1のモード	通常のプログラム処理	

図3 - 1 MBE = 0 のモードとMBE = 1 のモードの使い分け



備考 ————— MBE = 1 のとき, MBE = 0 のとき。

MBEは、サブルーチン処理時に自動的に退避 / 復帰されますから、サブルーチン処理中でも自由に変更できます。また、割り込み処理時には、自動的に退避 / 復帰されるとともに、割り込みベクタ・テーブルの設定によって、割り込み処理開始と同時に割り込み処理中のMBEを指定することができます。したがって、高速の割り込み処理に有利です。

MBSをサブルーチン処理あるいは割り込み処理で変更する場合は、PUSH/POP命令によって退避 / 復帰します。

MBEの設定は、SET1命令/CLR1命令で行います。MBSの設定は、SEL命令により行います。

例1 . MBEをクリアし、メモリ・バンクを固定する。

```
CLR1 MBE ; MBE 0
```

2 . メモリ・バンク 1 を選択する。

```
SET1 MBE ; MBE 1
```

```
SEL MB1 ; MBS 1
```

3.1.2 データ・メモリのアドレッシング・モード

μPD753208に採用されている75XLシリーズのアーキテクチャでは、データ・メモリ空間を、処理するデータのビット長ごとに効率よくアドレッシングするために、表3 - 1にまとめたように7種類のアドレッシング・モードが用意されており、効率のよいプログラミングが可能となります。

(1) 1ビット直接アドレッシング (mem. bit)

全データ・メモリ空間の各ビットを、命令のオペランドで直接に指定するアドレッシング・モードです。

指定されるメモリ・バンク (MB) は、MBE = 0 のモードでは、オペランドで指定されたアドレスが00H-7FHの場合MB = 0 に、アドレスが80H-FFHの場合MB = 15 に固定されます。したがって、MBE = 0 のモードでは000H-07FHのデータ・エリアと、F80H-FFFFHの周辺ハードウェア・エリアの両方をアドレッシングすることができます。

MBE = 1 のモードでは、MB = MBSとなり、指定できるデータ・メモリ空間を拡張することができます。

このアドレッシング・モードは、ビットのセット、リセット命令 (SET1/CLR1)、ビットのテスト命令 (SKT/SKF) の4つの命令に適用できます。

例 FLAG1をセット、FLAG2をリセット、FLAG3が0であるかテストする。

FLAG1	EQU	03FH. 1	; 3FH番地ビット 1
FLAG2	EQU	087H. 2	; 87H番地ビット 2
FLAG3	EQU	0A7H. 0	; A7H番地ビット 0
SET1	MBE		; MBE 1
SEL	MB0		; MBS 0
SET1	FLAG1		; FLAG1 1
CLR1	FLAG2		; FLAG2 0
SKF	FLAG3		; FLAG3 = 0?

図3 - 2 データ・メモリの構成と、各アドレッシング・モードのアドレッシング範囲

アドレッシング・モード	mem mem.bit		@HL @H + mem. bit		@DE @DL	スタック・アドレッシング	fmem. bit	pmem. @L
	MBE = 0	MBE = 1	MBE = 0	MBE = 1	-	-	-	-
メモリ・バンク許可フラグ								
0 0 0 H	汎用レジスタ・エリア							
0 1 F H								
0 2 0 H								
0 7 F H	データ・エリア スタティックRAM (メモリ・バンク0)	MBS = 0	MBS = 0	MBS = 0	SBS = 0			
0 8 0 H								
0 F F H	データ・エリア スタティックRAM (メモリ・バンク1)							
1 0 0 H								
1 E B H	表示データ・メモリ							
1 E C H								
1 F 7 H								
1 F 8 H								
1 F F H								
内蔵されていない								
F 8 0 H	周辺ハードウェア・エリア (メモリ・バンク15)							
F B 0 H								
F B F H								
F C 0 H								
F F 0 H								
F F F H								

備考 - : don't care

表3 - 1 アドレッシング・モード

アドレッシング・モード	表現形式	指定されるアドレス
1ビット 直接アドレッシング	mem.bit	MBとmemで示されるアドレスの, bitで示されるビット。 ・ MBE = 0 のとき mem = 00H-7FHのとき : MB = 0 mem = 80H-FFHのとき : MB = 15 ・ MBE = 1 のとき : MB = MBS
4ビット 直接アドレッシング	mem	MBとmemで示されるアドレス。 ・ MBE = 0 のとき mem = 00H-7FHのとき : MB = 0 mem = 80H-FFHのとき : MB = 15 ・ MBE = 1 のとき : MB = MBS
8ビット 直接アドレッシング		MBとmemで示されるアドレス (memは偶数アドレス)。 ・ MBE = 0 のとき mem = 00H-7FHのとき : MB = 0 mem = 80H-FFHのとき : MB = 15 ・ MBE = 1 のとき : MB = MBS
4ビット・レジスタ 間接アドレッシング	@HL	MBとHLで示されるアドレス。 ただし, MB = MBE · MBS
	@HL + @HL -	MBとHLで示されるアドレス。ただし, MB = MBE · MBS HL + は, アドレッシングのあとLレジスタを自動的にインクリメント。 HL - は, アドレッシングのあとLレジスタを自動的にデクリメント。
	@DE	メモリ・バンク0の, DEで示されるアドレス。
	@DL	メモリ・バンク0の, DLで示されるアドレス。
8ビット・レジスタ 間接アドレッシング	@HL	MBとHLで示されるアドレス (Lレジスタの内容は偶数)。 ただし, MB = MBE · MBS
ビット操作 アドレッシング	fmem.bit	fmemで示されるアドレスの, bitで示されるビット。 fmem = FB0H-FBFH (割り込み関係のハードウェア) FF0H-FFFH (I/Oポート)
	pmem. @L	pmemの上位10ビットとLレジスタの上位2ビットで示されるアドレスの, Lレジスタの下位2ビットで示されるビット。 ただし, pmem = FC0H-FFFH
	@H + mem.bit	MBとHそしてmemの下位4ビットで示されるアドレスの, bitで示される ビット。 ただし, MB = MBE · MBS
スタック・アドレッシング	-	SBSで選択されるメモリ・バンク0, 1の, SPで示されるアドレス。

(2) 4ビット直接アドレッシング(mem)

全データ・メモリ空間を4ビット単位で、命令のオペランドによって直接に指定するアドレッシング・モードです。

1ビット直接アドレッシングと同様に、MBE = 0のモードでは、指定できる領域が、000H-07FHのデータ・エリアとF80H-FFFHの周辺ハードウェア・エリアに固定されます。MBE = 1のモードでは、MB = MBSとなって、指定できるデータ・メモリ空間は全空間に拡張されます。

このアドレッシング・モードは、MOV/XCH/INCS/IN/OUT命令に適用されます。

注意 例1のように、入出力ポートに関連するデータをバンク1のスタティックRAMに格納するとプログラムの効率が悪くなってしまいます。入出力ポートと関連したデータはバンク0の00H-7FH番地に格納するようにすれば、例2のようにMBSを変更せずにプログラムすることができます。

例1 . 「BUFF」のデータをポート5に出力する。

```

BUFF EQU 11AH ; 「BUFF」は11AH番地
SET1 MBE ; MBE 1
SEL MB1 ; MBS 1
MOV A, BUFF ; A (BUFF)
SEL MB15 ; MBS 15
OUT PORT5, A ; PORT5 A

```

2 . ポート5を入力し、「DATA1」に格納する。

```

DATA1 EQU 5FH ; 「DATA1」は5FH番地
CLR1 MBE ; MBE 0
IN A, PORT5 ; A PORT5
MOV DATA1, A ; (DATA1) A

```

(3) 8ビット直接アドレッシング (mem)

全データ・メモリ空間を8ビット単位で、命令のオペランドによって直接に指定するアドレッシング・モードです。

オペランドで指定できるアドレスは偶数アドレスであり、オペランドで指定されたアドレスの4ビット・データと、プラス1されたアドレスの4ビット・データがペアとなって、8ビット・アキュムレータ (XAレジスタ・ペア) との間で8ビット処理されます。

指定されるメモリ・バンクは、4ビット直接アドレッシングと同じです。

このアドレッシング・モードは、MOV/XCH/IN/OUT命令に適用されます。

例1 . ポート8, 9の8ビット・データを20H, 21H番地に転送する。

```
DATA EQU 020H
      CLR1 MBE ; MBE 0
      IN XA, PORT8 ; X ポート9, A ポート8
      MOV DATA, XA ; (21H) X, (20H) A
```

2 . シリアル・インタフェースのシフト・レジスタ (SIO) に入力された8ビット・データを取り込むと同時に、転送データをセットし、転送スタート指示する。

```
SEL MB15 ; MBS 15
XCH XA, SIO ; XA (SIO)
```

(4) 4ビット・レジスタ間接アドレッシング (@rpa)

データ・メモリ空間を命令のオペランドで指定されたデータ・ポインタ（汎用レジスタのレジスタ・ペア）によって4ビット単位で間接に指定するアドレッシング・モードです。

データ・ポインタとしては、MB = MBE・MBSの指定によって全データ・メモリ空間を指定できるHLレジスタ・ペアとMBE、MBSの指定にかかわらず常にメモリ・バンク0に固定されているDEレジスタ・ペア/DLレジスタ・ペアの3種類があります。使用するデータ・メモリのバンクによってデータ・ポインタを使い分けることによって効率のよいプログラムが可能となります。

例 50H-57Hのデータを110H-117Hに転送する。

```

DATA1    EQU    57H
DATA2    EQU    117H
          SET1   MBE
          SEL   MB1
          MOV   D, #DATA1 SHR 4
          MOV   HL, #DATA2 AND 0FFH    ; HL 17H
LOOP :    MOV   A, @DL                  ; A (DL)
          XCH  A, @HL                  ; A (HL)
          DECS L                        ; L L - 1
          BR   LOOP

```

データ・ポインタとしてHLレジスタ・ペアを使用するアドレッシング・モードは、データの転送、演算、比較、入出力などに幅広く適用されます。DE、DLレジスタ・ペアを使用するアドレッシング・モードは、MOV/XCH命令に適用されます。

また、汎用レジスタあるいはレジスタ・ペアの増減命令と組み合わせることによって、図3-3に示されるようにデータ・メモリ空間のアドレスを自由自在に更新することができます。

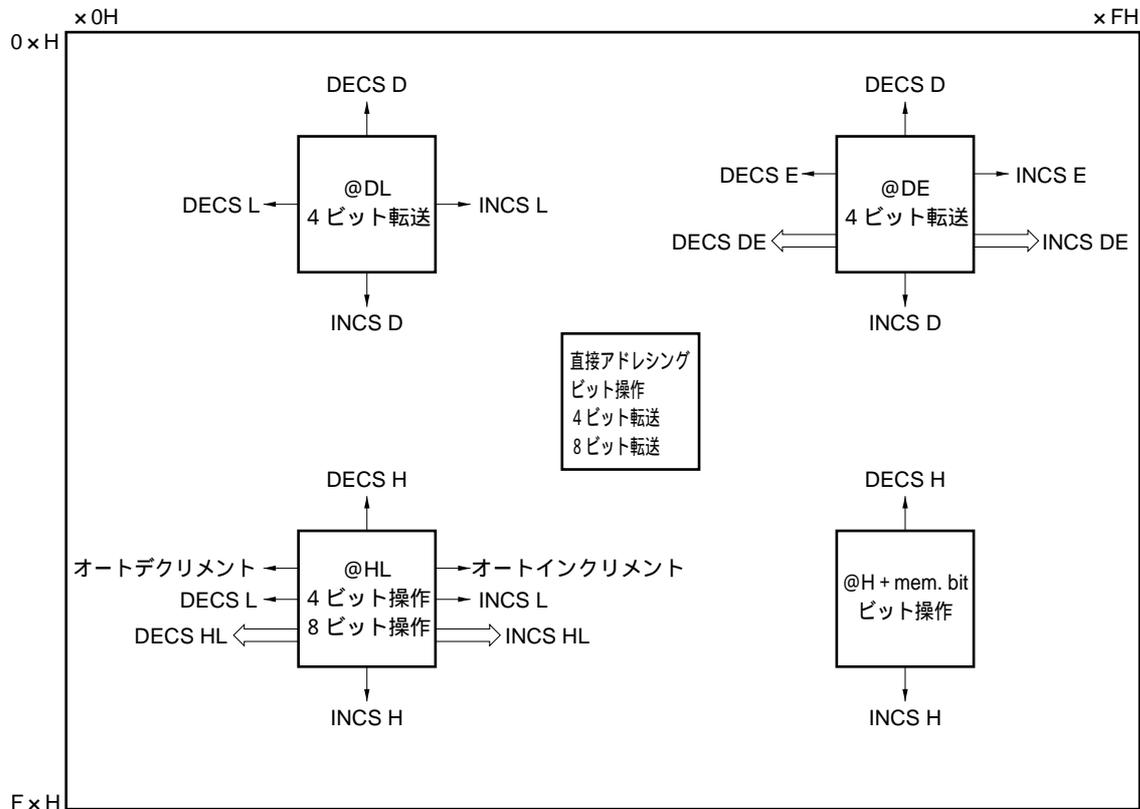
例1 . 50H-57Hのデータと110H-117Hのデータを比較する。

```
DATA1 EQU 57H
DATA2 EQU 117H
SET1 MBE
SEL MB1
MOV D, #DATA1 SHR 4
MOV HL, #DATA2 AND 0FFH
LOOP: MOV A, @DL
SKE A, @HL ; A = (HL) ?
BR NO ; NO
DECS L ; YES, L L - 1
BR LOOP
```

2 . 00H-FFHのデータ・メモリを0クリアする。

```
CLR1 RBE
CLR1 MBE
MOV XA, #00H
MOV HL, #04H
LOOP: MOV @HL, A ; (HL) A
INCS L ; L L + 1
BR LOOP
INCS H ; H H + 1
BR LOOP
```

図3 - 3 スタティックRAMのアドレス更新方法



(5) 8ビット・レジスタ間接アドレッシング (@HL)

全データ・メモリ空間を8ビット単位で、データ・ポインタ(HLレジスタ・ペア)により間接に指定するアドレッシング・モードです。

データ・ポインタのビット0(Lレジスタのビット0)を0にしたアドレスの4ビット・データとプラス1されたアドレスの4ビット・データがペアとなって、8ビット・アキュムレータ(XAレジスタ)との間で、8ビット処理されます。

指定されるメモリ・バンクは4ビット・レジスタ間接アドレッシングでHLレジスタを指定した場合と同じで、MB = MBE・MBSとなります。このアドレッシング・モードは、MOV/XCH/SKE命令に適用されます。

例1. タイマ/イベント・カウンタ0のカウント・レジスタ(T0)値が30H, 31H番地のデータと等しいかどうか比較する。

```

DATA EQU 30H
CLR1 MBE
MOV HL, #DATA
MOV XA, T0 ; XA カウント・レジスタ0
SKE A, @HL ; A = (HL) ?
BR NO
INCS L
MOV A, X ; A X
SKE A, @HL ; A = (HL) ?
    
```

2. 00H-FFHのデータ・メモリを0クリアする。

```

CLR1 RBE
CLR1 MBE
MOV XA, #00H
MOV HL, #04H
LOOP: MOV @HL, XA ; (HL) XA
INCS L
BR LOOP
INCS H
BR LOOP
    
```

(6) ビット操作アドレッシング

全データ・メモリ空間の各ビットに対し、ビット操作（ブーリアン処理、ビット転送等）を行うためのアドレッシング・モードです。

1ビット直接アドレッシングがビットのセット/リセット/テスト命令にしか適用できないのに対し、このアドレッシングではAND1/OR1/XOR1命令によるブーリアン処理、SKTCLR命令によるテスト&リセットなど、多彩なビット操作が可能になります。

ビット操作アドレッシングには、次に示す3つの方法があり、使用するデータ・メモリ・アドレスによって使い分けることができます。

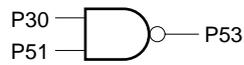
(a) 特定アドレス・ビット直接アドレッシング (fmem. bit)

このアドレッシング・モードは、周辺ハードウェアのうち入出力ポートや割り込み関係のフラグなど、特にビット操作をよく使うハードウェアをメモリ・バンクの設定に無関係にいつでも操作できるようにするためのアドレッシングです。したがって、このアドレッシングを適用できるデータ・メモリ・アドレスは、入出力ポートをマッピングしてあるFF0H-FF9Hと、割り込み関係のハードウェアをマッピングしてあるFB0H-FBFHとなっています。この2つのデータ・メモリ領域にあるハードウェアは、MBSおよびMBEの設定にかかわらず、いつでも自由に直接アドレッシングでビット操作することができます。

例1．タイマ0 割り込み要求フラグ (IRQT0) をテストし、セットされていたら要求フラグをクリアし、P63をリセットする。

```
SKTCLR  IRQT0          ; IRQT0 = 1 ?
BR       NO            ; NO
CLR1    PORT6. 3      ; YES
```

例2 . P30とP51の端子がともに1であれば、P53をリセットする。



```
( i )      SET1   CY           ; CY  1
           AND1   CY, PORT3. 0 ; CY  P30
           AND1   CY, PORT5. 1 ; CY  P51
           SKT    CY           ; CY = 1 ?
           BR     SETP
           CLR1   PORT5. 3     ; P53  0
           ⋮
SETP :     SET1   PORT5. 3     ; P53  1
           ⋮

( ii )     SKT    PORT3. 0     ; P30 = 1 ?
           BR     SETP
           SKT    PORT5. 1     ; P51 = 1 ?
           BR     SETP
           CLR1   PORT5. 3     ; P53  0
           ⋮
SETP :     SET1   PORT5. 3     ; P53  1
```

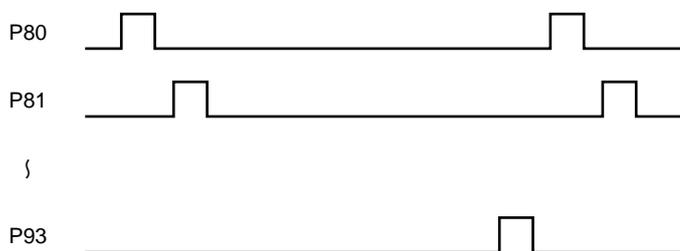
(b) 特定アドレス・ビット・レジスタ間接アドレッシング (pmem. @L)

このアドレッシング・モードは周辺ハードウェアのうち入出力ポートなどの各ビットをレジスタ間接指定し、連続的に操作するためのアドレッシングです。このアドレッシングを適用できるデータ・メモリ・アドレスはFC0H-FFFHとなっています。

このアドレッシング・モードは、データ・メモリ・アドレス12ビットの上位10ビット・アドレスをオペランドで直接指定し、下位2ビット・アドレスとビット・アドレスはLレジスタによって間接指定します。したがって、Lレジスタの指定によって16ビット（4ポート）を連続的に操作できます。

このアドレッシングも、MBE、MBSの設定に無関係にいつでもビット操作することができます。

例 ポート8からポート9の各ビットに順にパルス出力する。



```

LOOP2:  MOV     L, #0
LOOP1:  SET1    PORT8.@L    ; ポート8 , 9のビット (L1-0)  1
        CLR1    PORT8.@L    ; ポート8 , 9のビット (L1-0)  0
        INCS   L
        NOP
        SKE    L, #08H
        BR     LOOP1
        BR     LOOP2
    
```

(c) 特殊1ビット直接アドレッシング (@H+mem. bit)

このアドレッシング・モードは、全データ・メモリ空間の各ビットに対し、ビット操作を可能にするためのアドレッシングです。

このアドレッシングでは、MB = MBE・MBSで指定されるメモリ・バンクのデータ・メモリ・アドレスの上位4ビット・アドレスをHレジスタで間接指定し、下位4ビット・アドレスとビット・アドレスはオペランドで直接指定します。このアドレッシングによってすべてのデータ・メモリの各ビットに対し多彩なビット操作が可能となります。

例 30H番地のビット3 (FLAG1) と31H番地のビット0 (FLAG2) がともに0かともに1なら32H番地のビット2 (FLAG3) をリセットする。



```

FLAG1 EQU 30H. 3
FLAG2 EQU 31H. 0
FLAG3 EQU 32H. 2

SEL MB0
MOV H, #FLAG1 SHR 6
CLR1 CY ; CY 0
OR1 CY, @H+FLAG1 ; CY CY FLAG1
XOR1 CY, @H+FLAG2 ; CY CY FLAG2
SET1 @H+FLAG3 ; FLAG3 1
SKT CY ; CY = 1 ?
CLR1 @H+FLAG3 ; FLAG3 0
    
```

(7) スタック・アドレッシング

このアドレッシング・モードは、割り込み処理時、サブルーチン処理時の退避 / 復帰動作のためのアドレッシングです。

このアドレッシングでは、データ・メモリ・バンク 0 の、スタック・ポインタ (8 ビット) で示されるアドレスが指定されます。

このアドレッシングは、割り込み処理時、サブルーチン処理時のほか PUSH/POP 命令によるレジスタの退避 / 復帰にも適用されます。

例 1 . サブルーチン処理時でレジスタを退避 / 復帰する。

```
SUB :  PUSH   XA
      PUSH   HL
      PUSH   BS   ; MBSとRBSの退避
      :
      POP    BS
      POP    HL
      POP    XA
      RET
```

2 . HLレジスタ・ペアの内容をDEレジスタ・ペアに転送する。

```
PUSH   HL
POP    DE   ; DE HL
```

3 . [XABC] のレジスタで示されるアドレスへ分岐する。

```
PUSH   BC
PUSH   XA
RET           ; XABC番地へ分岐
```

3.2 汎用レジスタのバンク構成

μPD753208は、X, A, B, C, D, E, H, Lの8つの汎用レジスタを1バンクとして、4つのレジスタ・バンクを内蔵しています。この汎用レジスタ・エリアはデータ・メモリのメモリ・バンク0の00H-1FH番地にマッピングされています(図3-5参照)。この汎用レジスタのバンクを指定するためにレジスタ・バンク許可フラグ(RBE)とレジスタ・バンク選択レジスタ(RBS)が内蔵されています。RBSはレジスタ・バンクを選択するためのレジスタで、RBEはRBSで選択されたレジスタ・バンクを有効とするか否かを決定するフラグです。命令実行の際に有効となるレジスタ・バンク(RB)は、次のようになります。

$$RB = RBE \cdot RBS$$

表3-2 RBE, RBSと選択されるレジスタ・バンク

RBE	RBS				レジスタ・バンク
	3	2	1	0	
0	0	0	x	x	バンク0に固定
1	0	0	0	0	バンク0を選択
			0	1	バンク1 "
			1	0	バンク2 "
			1	1	バンク3 "

↑ ↑
0に固定

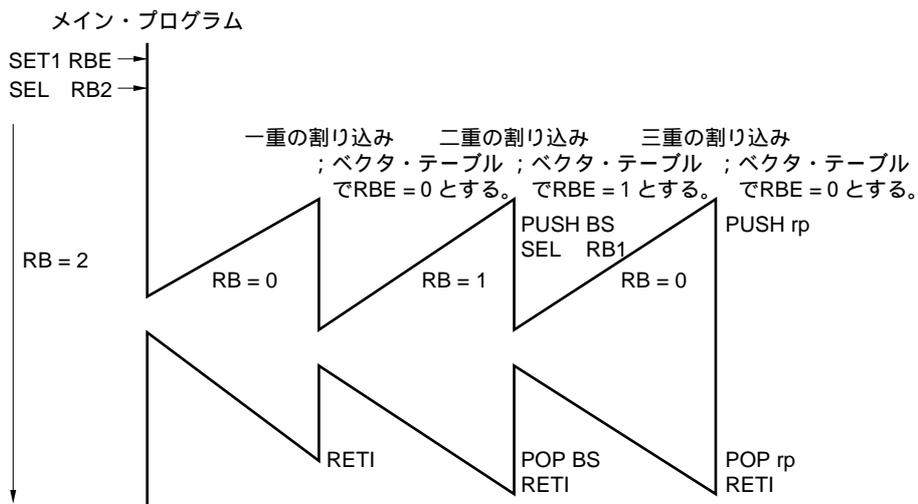
備考 x = don't care

RBEは、サブルーチン処理時に自動的に退避/復帰されますので、サブルーチン処理中に自由に設定することができます。また、割り込み処理時は、自動的に退避/復帰されるとともに、割り込みベクタ・テーブルの設定によって、割り込み処理開始と同時に、割り込み処理中のRBEを設定することができます。したがって、表3-3に示すように、通常処理と割り込み処理で、レジスタ・バンクを使い分ければ、一重割り込みでは、汎用レジスタの退避/復帰は不要、二重割り込みでは、RBSの退避/復帰のみとなり、割り込み処理の高速化が図れます。

表3 - 3 通常ルーチンと割り込みルーチンでのレジスタ・バンクの使い分けの例

通常の処理	RBE = 1 とし、レジスタ・バンク 2, 3 を使用する。
一重割り込み処理	RBE = 0 とし、レジスタ・バンク 0 を使用する。
二重割り込み処理	RBE = 1 とし、レジスタ・バンク 1 を使用する。 (このとき、RBSの退避 / 復帰が必要)
三重以上の割り込み処理	PUSH, POPでレジスタ退避 / 復帰をする。

図3 - 4 レジスタ・バンクの使い分け例



RBSを、サブルーチン処理あるいは割り込み処理で変更する場合は、PUSH/POP命令によって退避 / 復帰します。

RBEの設定は、SET1/CLR1命令によって行います。RBSの設定は、SEL命令により行います。

例 SET1 RBE ; RBE 1
CLR1 RBE ; RBE 0
SEL RB0 ; RBS 0
SEL RB3 ; RBS 3

μ PD753208に内蔵されている汎用レジスタ・エリアは、4ビット・レジスタとしての使用のほかに、レジスタ・ペアによる8ビット・レジスタとして使用ができ、8ビット・マイコンに匹敵する転送、演算、比較、増減命令によって、汎用レジスタ中心のプログラミングが可能となります。

(1) 4ビット・レジスタとして使用する場合

汎用レジスタ・エリアを4ビット・レジスタとして使用する場合には、図3-5に示すように、 $RB = RBE \cdot RBS$ で指定されたレジスタ・バンクのX, A, B, C, D, E, H, L, 計8個の汎用レジスタを使うことができます。このうちAレジスタは4ビット・アキュムレータとして、4ビット・データの転送、演算、比較などに中心的な働きをします。ほかの汎用レジスタは、アキュムレータとの転送、比較、増減ができます。

(2) 8ビット・レジスタとして使用する場合

汎用レジスタ・エリアを8ビット・レジスタとして使用する場合には、図3-6に示すように、 $RB = RBE \cdot RBS$ で指定されたレジスタ・バンクのレジスタ・ペアをXA, BC, DE, HLとし、レジスタ・バンク (RB) のビット0を反転したレジスタ・バンクのレジスタ・ペアをXA', BC', DE', HL'として計8個の8ビット・レジスタを使うことができます。このうちXAレジスタ・ペアは、8ビット・アキュムレータとして、8ビット・データの転送、演算、比較などに中心的な働きをします。ほかのレジスタ・ペアは、アキュムレータとの転送、演算、比較、増減ができます。また、HLレジスタ・ペアは、主にデータ・ポインタとして機能します。DE, DLレジスタ・ペアも補助的なデータ・ポインタとして機能します。

例1 .	INCS	HL	; HL HL + 1, HL = 00Hでスキップ
	ADDS	XA, BC	; XA XA + BC, キャリーでスキップ
	SUBC	DE', XA	; DE' DE' - XA - CY
	MOV	XA, XA'	; XA XA'
	MOVT	XA, @PCDE	; XA (PC ₁₂₋₈ + DE) _{ROM} , テーブル参照
	SKE	XA, BC	; XA = BCならスキップ

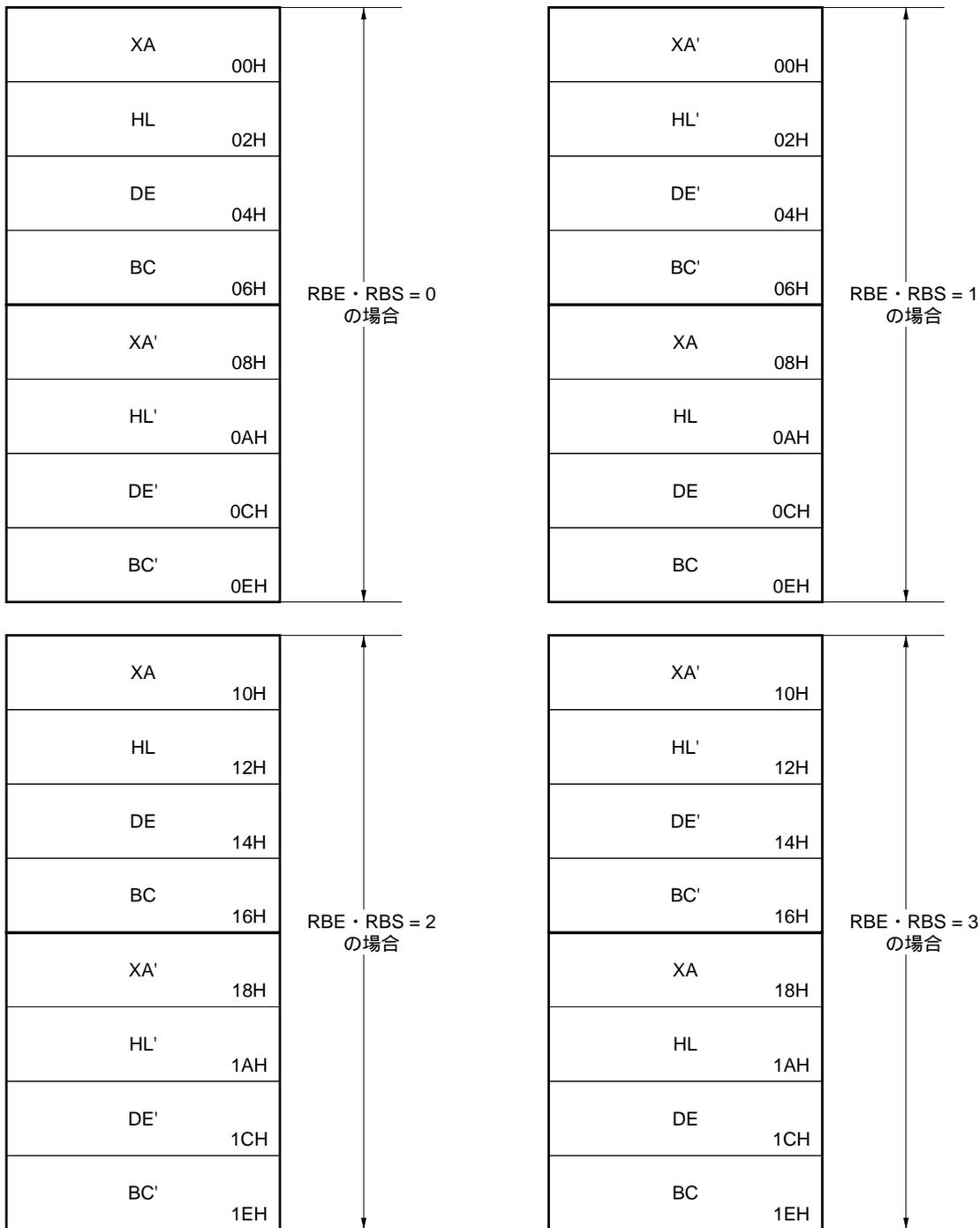
2. タイマ/イベント・カウンタ0のカウント・レジスタ (T0) の値がBC'レジスタ・ペアの値より大きいかどうかをテストし、大きくなるまで待つ。

	CLR1	MBE	
NO :	MOV	XA, T0	; カウント・レジスタ読み取り
	SUBS	XA, BC'	; XA BC' ?
	BR	YES	; YES
	BR	NO	; NO

図3 - 5 汎用レジスタの構成 (4ビット処理の場合)

X	01H	A	00H	レジスタ・バンク 0 (RBE・RBS = 0)
H	03H	L	02H	
D	05H	E	04H	
B	07H	C	06H	
X	09H	A	08H	レジスタ・バンク 1 (RBE・RBS = 1)
H	0BH	L	0AH	
D	0DH	E	0CH	
B	0FH	C	0EH	
X	11H	A	10H	レジスタ・バンク 2 (RBE・RBS = 2)
H	13H	L	12H	
D	15H	E	14H	
B	17H	C	16H	
X	19H	A	18H	レジスタ・バンク 3 (RBE・RBS = 3)
H	1BH	L	1AH	
D	1DH	E	1CH	
B	1FH	C	1EH	

図3 - 6 汎用レジスタの構成 (8ビット処理の場合)



3.3 メモリ・マップトI/O

μPD753208は、図3 - 2に示したようにデータ・メモリ空間上のF80H-FFFH番地に入出力ポートやタイマなどの周辺ハードウェアをマッピングしたメモリ・マップトI/Oを採用しています。このため、周辺ハードウェアを制御する特別な命令はなく、すべてメモリ操作命令により制御します（一部、プログラムをわかりやすくするためのハードウェア制御の二モニックが用意してあります）。

周辺ハードウェアを操作する場合は表3 - 4のようなアドレッシング・モードが使用できます。

なお、1ECH-1F7H番地にマッピングしてある表示データ・メモリはメモリ・バンク1を指定して操作します。

表3 - 4 周辺ハードウェア操作時に適用可能なアドレッシング・モード

	適用可能なアドレッシング・モード	適用可能なハードウェア
ビット操作	MBE = 0 , または (MBE = 1 , MBS = 15) として 直接アドレッシングmem.bitで指定	ビット操作可能なすべてのハードウェア
	MBE, MBSにかかわらず, 直接アドレッシングfmem.bitで指定	IST1, IST0, MBE, RBE IE x x x , IRQ x x x , PORTn. x
	MBE, MBSにかかわらず, 間接アドレッシングpmem.@Lで指定	BSBn. x PORTn. x
4ビット操作	MBE = 0 , または (MBE = 1 , MBS = 15) として 直接アドレッシングmemで指定	4ビット操作可能なすべてのハードウェア
	(MBE = 1 , MBS = 15) として レジスタ間接アドレッシング@HLで指定	
8ビット操作	MBE = 0 , または (MBE = 1 , MBS = 15) として直接アドレッシングmemで指定。ただし、memは偶数アドレス	8ビット操作可能なすべてのハードウェア
	MBE = 1 , MBS = 15としてレジスタ間接アドレッシング@HLで指定。ただし、Lレジスタの内容は偶数	

例 CLR1 MBE ; MBE = 0
 SET1 TM0. 3 ; タイマ0のスタート
 EI IE0 ; INT0の許可
 DI IET1 ; INTT1の禁止
 SKTCLR IRQ2 ; INT2要求フラグのテスト・アンド・クリア
 SET1 PORT5. @L ; ポート5のセット

図3 - 7 に μ PD753208のI/Oマップをまとめます。

同図における項目の意味は次のようになっています。

- ・略号.....内蔵されたハードウェアのアドレスを示すネームです。
命令のオペランド欄に記述できます。
- ・R/W.....該当するハードウェアが、読み出し / 書き込み可能かどうかを示します。
 - R/W : 読み出し (Read) / 書き込み (Write) 可能
 - R : 読み出し (Read) のみ
 - W : 書き込み (Write) のみ
- ・操作可能なビット数.....該当するハードウェアを操作する場合に、適用可能な処理ビット数を示します。
 - : その欄で使用している単位 (1/4/8 ビット) でのビットの操作が可能。
 - ◐ : 一部のビットのみ操作が可能。操作の可能なビットについては備考欄を参照してください。
 - : その欄で使用している単位 (1/4/8 ビット) でのビットの操作が不可能。
- ・ビット操作アドレッシング.....該当するハードウェアをビット操作する場合に、適用可能なビット操作アドレッシングを示します。

図3-7 μPD753208 I/Oマップ (1/5)

アドレス	ハードウェア名称 (略号)				R/W	操作可能なビット数			ビット操作 アドレッシング	備 考
	b3	b2	b1	b0		1ビット	4ビット	8ビット		
F80H	スタック・ポインタ (SP)				R/W	-	-		-	ビット0は0に固定
F82H	レジスタ・バンク選択レジスタ (RBS)				R	-			-	注1
F83H	バンク選択レジスタ (BS)					-				
F84H	メモリ・バンク選択レジスタ (MBS)				R/W	-		-	-	
F85H	スタック・バンク選択レジスタ (SBS)				R/W	-		-	-	
F86H	ベーシック・インターバル・タイム・モード・レジスタ (BTM)				W			-	mem. bit	ビット3のみ ビット操作可能
F86H	ベーシック・インターバル・タイム (BT)				R	-	-		-	
F88H	タイム・カウンタ・ハイ・レベル期間設定用				R/W	-	-		-	
	モジュロ・レジスタ (TMOD2H)									
F8BH	WDTM注2				W		-	-	mem. bit	ビット3のみ ビット操作可能
F8CH	表示モード・レジスタ (LCDM)				R/W	(W)	-		mem. bit	ビット3のみ ビット操作可能
						-	-			
F8EH	表示制御レジスタ (LCDC)				R/W	-		-	-	
F8FH	LCD/ポート選択レジスタ (LPS)				R/W	-		-	-	

注1 . 4ビット操作では、RBSおよびMBSとして別々に操作可能。

8ビット操作では、BSとして操作可能。

MBS, RBSへの書き込みは、それぞれSEL MBn, SEL RBn命令で行ってください。

2 . WDTM : ウォッチドッグ・タイム許可フラグ (W) ; 1度セットすると命令ではクリアできません。

図3-7 μPD753208 I/Oマップ (2/5)

アドレス	ハードウェア名称 (略号)				R/W	操作可能なビット数			ビット操作 アドレッシング	備 考
	b3	b2	b1	b0		1ビット	4ビット	8ビット		
F90H	タイマ・カウンタ2モード・レジスタ				R/W	(W)	-		-	ビット3のみ ビット操作可能
	(TM2)					-	-		-	
F92H	TOE2	REMC	NRZB	NRZ	R/W				-	ビット3はW のみ可能
	タイマ・カウンタ2コントロール・レジスタ (TC2)								-	ビット3のみ ビット操作可能
	TGCE	-	-	-			-			
F94H	タイマ・カウンタ2カウント・レジスタ				R	-	-		-	
	(T2)									
F96H	タイマ・カウンタ2モジュロ・レジスタ				R/W	-	-		-	
	(TMOD2)									
F98H	時計モード・レジスタ (WM)				R/W	-	-		-	
						-	-			ビット6は0に 固定

FA0H	タイマ/イベント・カウンタ0				R/W	(W)	-		mem. bit	ビット3のみ ビット操作可能
	モード・レジスタ (TM0)					-	-		-	
FA2H	TOE0 ^{注1}				W	-	-		mem. bit	
FA4H	タイマ/イベント・カウンタ0				R	-	-		-	
	カウント・レジスタ (T0)									
FA6H	タイマ/イベント・カウンタ0				R/W	-	-		-	
	モジュロ・レジスタ (TMOD0)									
FA8H	タイマ・カウンタ1モード・レジスタ				R/W	(W)	-		mem. bit	ビット3のみ ビット操作可能
	(TM1)					-	-		-	
FAAH	TOE1 ^{注2}				W	-	-		mem. bit	
FACH	タイマ・カウンタ1カウント・レジスタ				R	-	-		-	
	(T1)									
FAEH	タイマ・カウンタ1モジュロ・レジスタ				R/W	-	-		-	
	(TMOD1)									

注1 . TOE0 : タイマ/イベント・カウンタ出力許可フラグ (チャンネル0) (W)

2 . TOE1 : タイマ・カウンタ出力許可フラグ (チャンネル1) (W)

図3-7 μPD753208 I/Oマップ (3/5)

アドレス	ハードウェア名称 (略号)				R/W	操作可能なビット数			ビット操作 アドレッシング	備 考
	b3	b2	b1	b0		1ビット	4ビット	8ビット		
FB0H	IST1	IST0	MBE	RBE	R/W	(R/W)	(R/W)	(R)	fmem. bit	8ビット操作 はRのみ可能
	プログラム・ステータス・ワード (PSW)					注2	-			
	CY ^{注1}	SK2 ^{注1}	SK1 ^{注1}	SK0 ^{注1}						
FB2H	割り込みプライオリティ選択レジスタ (IPS)				R/W	-		-		注3
FB3H	プロセッサ・クロック・コントロール・レジスタ (PCC)				R/W	-		-		注4
FB4H	INT0エッジ検出モード・レジスタ (IM0)				R/W	-		-	-	
FB6H	INT2エッジ検出モード・レジスタ (IM2)				R/W	-		-		ビット0, 1のみ ビット操作可能
FB8H	INTAレジスタ (INTA)				R/W			-	fmem. bit	
	IE4	IRQ4	IEBT	IRQBT						
FBAH	INTCレジスタ (INTC)				R/W					
			IEW	IRQW						
FBCH	INTEレジスタ (INTE)				R/W			-		
	IET1	IRQT1	IET0	IRQT0						
FBDH	INTFレジスタ (INTF)				R/W					
	IET2	IRQT2	IECSI	IRQCSI						
FBEH	INTGレジスタ (INTG)				R/W			-		
			IE0	IRQ0						
FBFH	INTHレジスタ (INTH)				R/W					
			IE2	IRQ2						
FC0H	ビット・シーケンシャル・バッファ0 (BSB0)				R/W				mem. bit pmem. @L	
FC1H	" 1 (BSB1)				R/W					
FC2H	" 2 (BSB2)				R/W					
FC3H	" 3 (BSB3)				R/W					

備考1 . IE x x x は割り込み許可フラグ

2 . IRQ x x x は割り込み要求フラグ

注1 . 予約語としては登録されていません。

2 . CYへの書き込みは、CY操作命令で行ってください。

3 . ビット3のみEI/DI命令により操作可能。

4 . ビット3, 2はSTOP/HALT命令実行時ビット操作可能。

図3-7 μPD753208 I/Oマップ(4/5)

アドレス	ハードウェア名称(略号)				R/W	操作可能なビット数			ビット操作 アドレッシング	備考
	b3	b2	b1	b0		1ビット	4ビット	8ビット		
★ FD0H	クロック出力モード・レジスタ(CLOM)				R/W	-		-	-	
FDCH	プルアップ抵抗指定レジスタ・ グループA(POGA)				R/W	-	-		-	
FDEH	プルアップ抵抗指定レジスタ・ グループB(POGB)				R/W	-	-		-	
FE0H	シリアル動作モード・レジスタ(CSIM)				R/W	-	-		-	注1
	CSIE	COI	WUP		(R)	(W)	-	mem. bit		
FE2H	CMD	RELD	CMDT	RELT	R/W		-	-	mem. bit	R/Wは、ビットにより異なります。
	SBIコントロール・レジスタ(SBIC)									
	BSYE	ACKD	ACKE	ACKT						
FE4H	シリアルI/Oシフト・レジスタ(SIO)				R/W	-	-		-	
FE6H	スレーブ・アドレス・レジスタ(SVA)				R/W	-	-		-	
FE8H	PM33注2	PM32注2	PM31注2	PM30注2	R/W	-	-		-	
	ポート・モード・レジスタ・グループA(PMGA)									
	PM63注2	PM62注2	PM61注2	PM60注2						
FECH	-	PM2注2	-	-	R/W	-	-		-	
	ポート・モード・レジスタ・グループB(PMGB)									
	-	-	PM5注2	-						
FEEH	-	-	PM9注2	PM8注2	R/W	-	-		-	
	ポート・モード・レジスタ・グループC(PMGC)									
	-	-	-	-						

注1. 1ビット操作は、ビットによりR/Wが異なります。

2. 予約語としては登録されていません。

図3 - 7 μ PD753208 I/Oマップ (5/5)

アドレス	ハードウェア名称 (略号)				R/W	操作可能なビット数			ビット操作 アドレッシング	備 考
	b3	b2	b1	b0		1ビット	4ビット	8ビット		
FF0H	ポート0 (PORT0)				R			-		
FF1H	ポート1 (PORT1)				R		-			
FF2H	ポート2 (PORT2)				R/W			-		
FF3H	ポート3 (PORT3)				R/W					
FF5H	ポート5 (PORT5)				R/W			-		
FF6H	KR3	KR2	KR1	KR0	R/W			-		
	ポート6 (PORT6)									
FF8H	ポート8 (PORT8)				R/W				fmem. bit pmem. @L	
FF9H	ポート9 (PORT9)				R/W					

第4章 内部CPU機能

4.1 Mk モードとMk モードの切り替え機能

4.1.1 Mk モードとMk モードの違い

μPD753208のCPUはMk モードとMk モードの2つのモードを持ち、どちらを使用するかを選択ができます。モードの切り替え操作は、スタック・バンク選択レジスタ(SBS)のビット3で行います。

- ・Mk モード：μPD75308Bと上位互換性があります。
ROM容量が16 Kバイトまでの75XL CPUで使用できます。
- ・Mk モード：μPD75308Bとの互換性はありません。
ROM容量が16 Kバイト以上の製品も含め、75XL CPU全部で使用できます。

表4-1 Mk モードとMk モードの違い

	Mk モード	Mk モード
サブルーチン命令の スタック・バイト数	2 バイト	3 バイト
BRA !addr1命令 CALLA !addr1命令	なし	あり
CALL !addr命令	3 マシン・サイクル	4 マシン・サイクル
CALLF !faddr命令	2 マシン・サイクル	3 マシン・サイクル

★ **注意** Mk モードは、75Xおよび75XLシリーズにおいて、16 Kバイトを越えるプログラム領域をサポートするモードです。このモードにより、16 Kバイトを越える製品とのソフトウェア上の互換性を高めることができます。

なお、Mk モードを選択すると、サブルーチン・コール命令実行時のスタック・バイト数(使用エリア)がMk モードに比べ1スタックごとに1バイト分増えます。また、CALL !addr, CALLF !faddr命令使用時は、マシン・サイクルがそれぞれ1マシン・サイクル分長くかかります。したがって、ソフトウェアの互換性よりRAMの使用効率や処理能力を重視する場合は、Mk モードを使用してください。

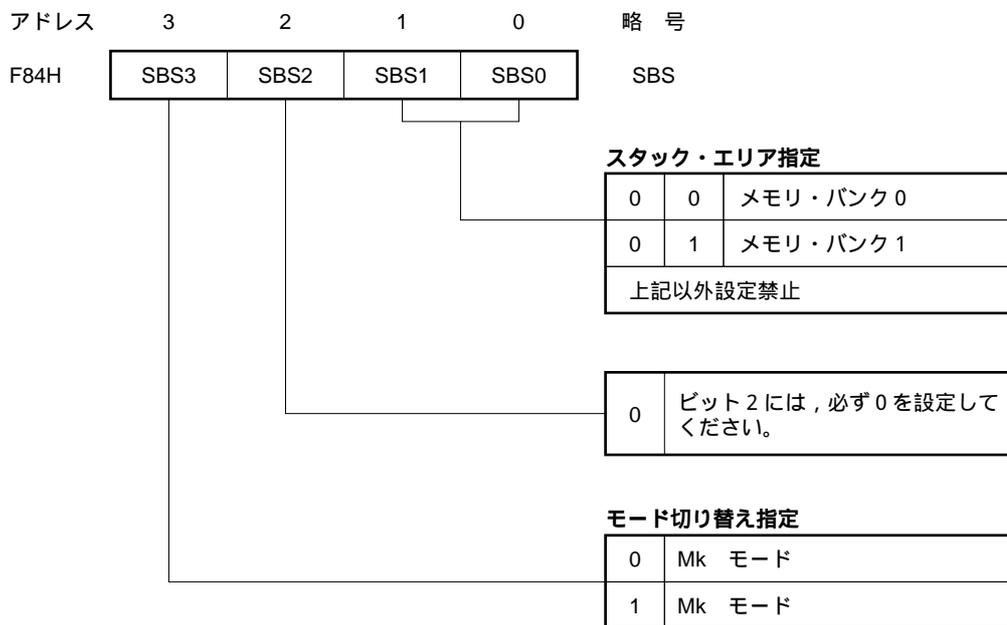
4.1.2 スタック・バンク選択レジスタ (SBS) の設定方法

Mk モードとMk モードの切り替えは、スタック・バンク選択レジスタによって行います。図4 - 1にそのフォーマットを示します。

スタック・バンク選択レジスタは、4ビット・メモリ操作命令により設定します。Mk モードを使用する場合は、プログラムの初期で必ずスタック・バンク選択レジスタを100×B^注にイニシャライズしてください。またMk モードを使用する場合は、必ず000×B^注にイニシャライズしてください。

注 ×には希望の値を設定してください。

図4 - 1 スタック・バンク選択レジスタのフォーマット



注意 SBS. 3はRESET信号発生後“1”になるので、CPUはMk モードで動作します。Mk モードの命令を使用する場合は、SBS. 3を“0”にし、Mk モードに設定してから使用してください。

4.2 プログラム・カウンタ (PC)

.....12ビット (μ PD753204)

13ビット (μ PD753206,753208)

14ビット (μ PD75P3216)

プログラム・メモリのアドレスを保持するバイナリ・カウンタです。

図4 - 2 プログラム・カウンタの構成

(a) μ PD753204

PC11	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
------	------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

(b) μ PD753206, 753208の場合

PC12	PC11	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
------	------	------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

(c) μ PD75P3216の場合

PC13	PC12	PC11	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
------	------	------	------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

プログラム・カウンタは、通常、命令を1つ実行することによりその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。

分岐命令 (BR, BRA, BRCB) 実行時には、分岐先アドレスを示すイミディエイト・データやレジスタ・ペアの内容がPCの全部または一部のビットにロードされます。

サブルーチン・コール命令 (CALL, CALLA, CALLF) 実行時およびベクタ割り込み発生時には、そのときのPC内容 (すでに次の命令をフェッチするためにインクリメントされているリターン・アドレス) がスタック・メモリ (スタック・ポインタにより指定されたデータ・メモリ) に退避されたあと、それぞれのジャンプ先のアドレスがロードされます。

リターン命令 (RET, RETS, RETI) 実行時には、スタック・メモリの内容がPCにセットされます。

μ PD753204は、 $\overline{\text{RESET}}$ 信号発生により、プログラム・メモリの0000H番地の下位4ビットの内容がPC11-PC8に、0001H番地の内容がPC7-PC0にイニシャライズされます。したがって任意のアドレスからプログラムをスタートできます。

μ PD753206, 753208は、 $\overline{\text{RESET}}$ 信号発生により、プログラム・メモリの0000H番地の下位5ビットの内容がPC12-PC8に、0001H番地の内容がPC7-PC0にイニシャライズされます。

μ PD75P3216は、 $\overline{\text{RESET}}$ 信号発生により、プログラム・メモリの0000H番地の下位6ビットの内容がPC13-PC8に、0001H番地の内容がPC7-PC0にイニシャライズされます。

4.3 プログラム・メモリ (ROM)

-4096 × 8 ビット (μPD753204)
-6144 × 8 ビット (μPD753206)
-8192 × 8 ビット (μPD753208)
-16384 × 8 ビット (μPD75P3216)

プログラム・メモリは、プログラム、割り込みベクタ・テーブル、GETI命令の参照テーブルおよびテーブル・データなどを格納するメモリです。

プログラム・メモリは、プログラム・カウンタによってアドレスされます。また、テーブル参照命令 (MOVT) によってテーブル・データを参照することができます。

分岐命令、サブルーチン・コール命令による分岐可能アドレス範囲は図4-3に示すとおりです。相対分岐命令 (BR \$addr命令) ではブロック境界に関係なく、[PCの内容 - 15 ~ - 1, + 2 ~ + 16] アドレスへ分岐できます。

プログラム・メモリのアドレスは次の範囲です。

- ・ 0000H-FFFFH : μPD753204
- ・ 0000H-17FFFH : μPD753206
- ・ 0000H-1FFFFH : μPD753208
- ・ 0000H-3FFFFH : μPD75P3216

次に示すアドレスには特別な機能を割り付けています。また、0000H、0001Hを除くすべての領域は、通常のプログラム・メモリとして使用できます。

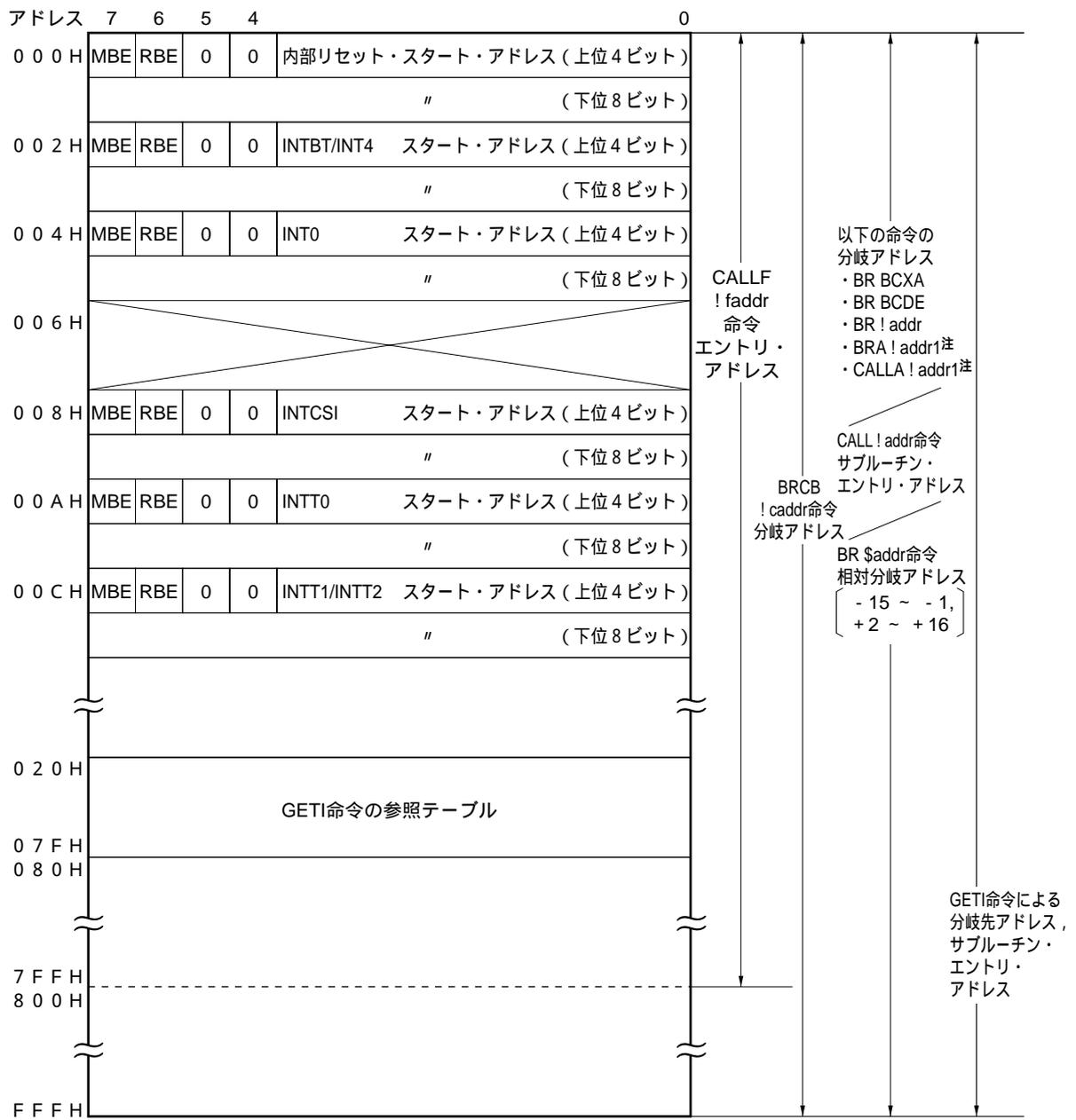
- ・ 0000H, 0001H番地
RESET信号発生でのプログラム・スタート・アドレスとRBE、MBEの設定値を書くベクタ・テーブルです。任意の番地からリセット・スタートできます。
- ・ 0002H-000DH番地
各ベクタ割り込みによるプログラム・スタート・アドレスとRBE、MBEの設定値を書くベクタ・テーブルです。任意の番地から割り込み処理をスタートできます。
- ・ 0020H-007FH番地
GETI命令^注で参照するテーブル領域です。

注 GETI命令は、任意の2バイト/3バイト命令、または1バイト命令2つを1バイトで実現するための命令で、プログラム・ステップ数を縮小するために使用します (11.1.1 GETI命令参照)。

図4-3 プログラム・メモリ・マップ(1/4)

★

(a) μPD753204

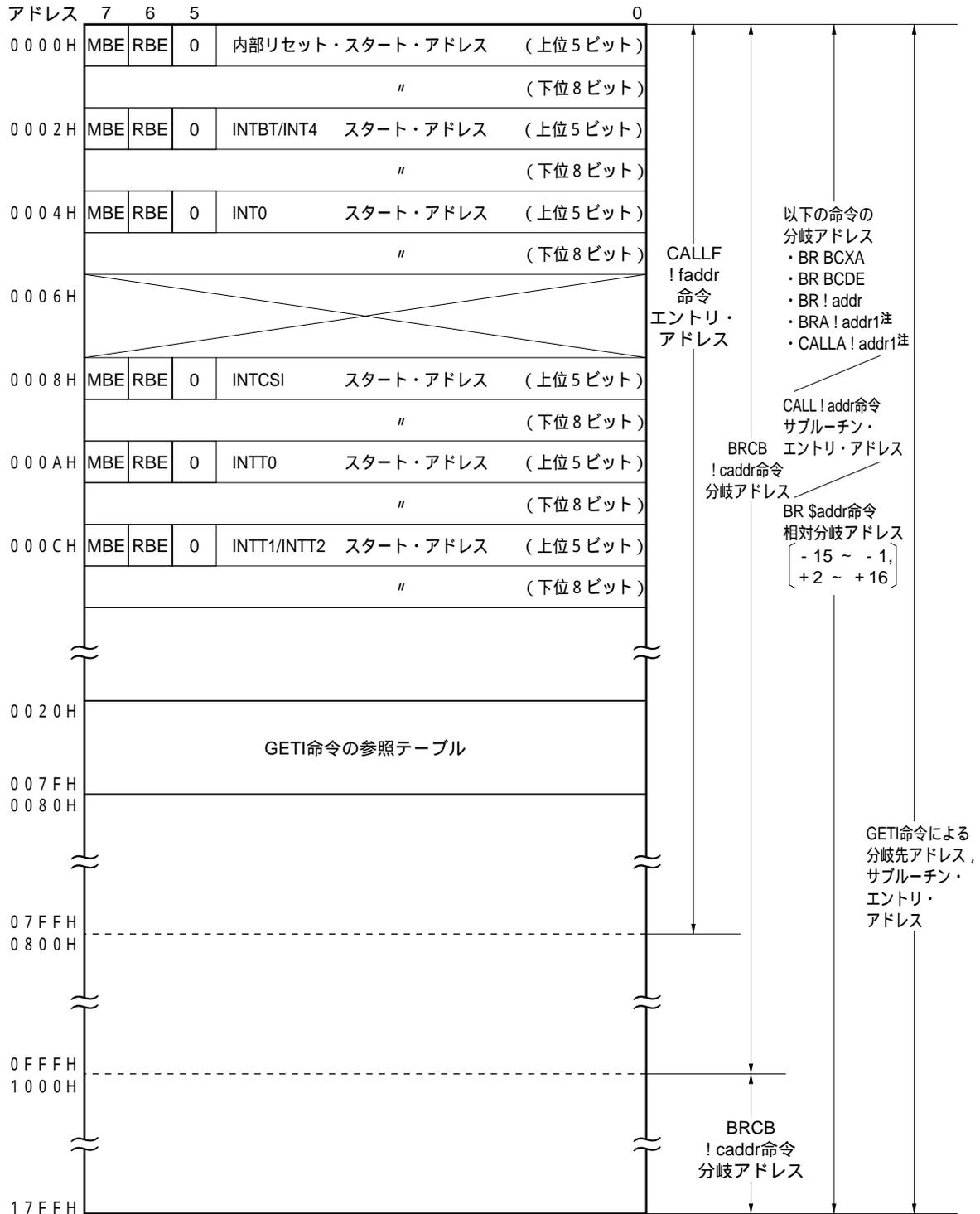


注 Mk モードでのみ使用できます。

備考 上記の命令以外では、BR PCDE, BR PCXA命令により、PCの下位8ビットのみ変更したアドレスへ分岐することができます。

図4 - 3 プログラム・メモリ・マップ(2/4)

(b) μ PD753206

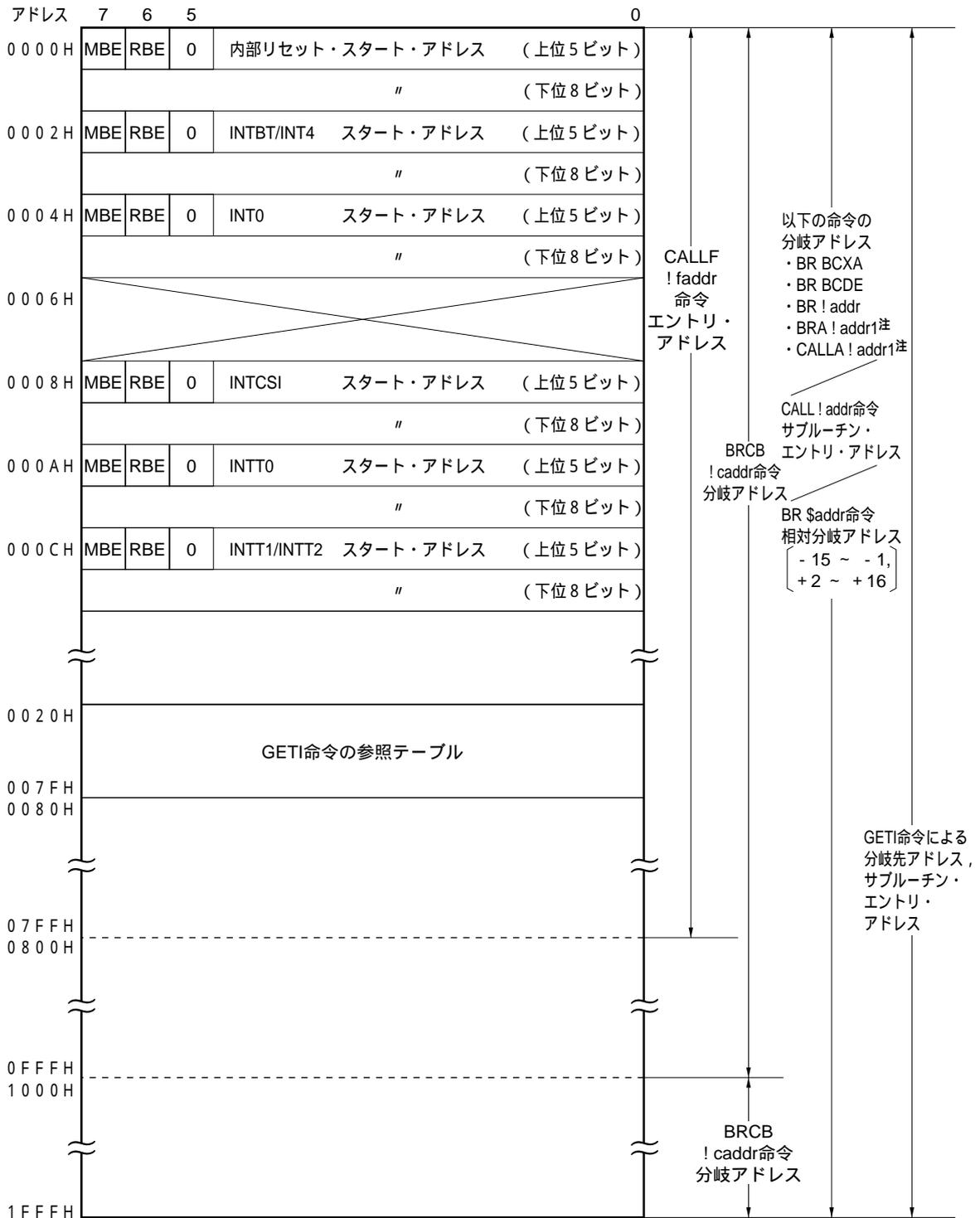


注 Mk モードでのみ使用できます。

備考 上記の命令以外では、BR PCDE、BR PCXA命令により、PCの下位8ビットのみ変更したアドレスへ分岐することができます。

図4-3 プログラム・メモリ・マップ(3/4)

(c) μ PD753208

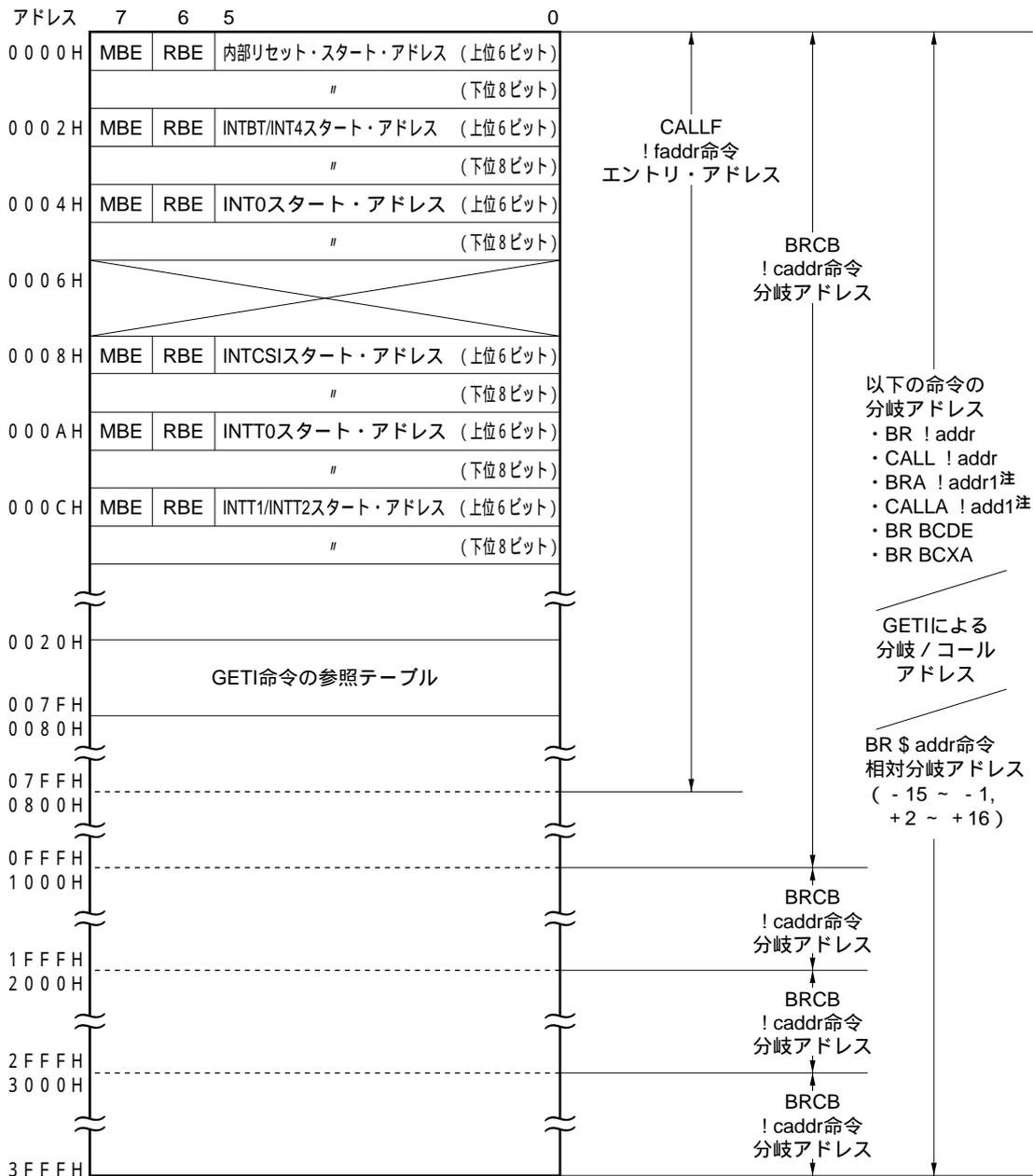


注 Mk モードでのみ使用できます。

備考 上記の命令以外では、BR PCDE, BR PCXA命令により、PCの下位8ビットのみ変更したアドレスへ分岐することができます。

図4 - 3 プログラム・メモリ・マップ (4/4)

(d) μ PD75P3216



注 Mk モードでのみ使用できます。

備考 上記の命令以外では、BR PCDE、BR PCXA命令により、PCの下位8ビットのみ変更したアドレスへ分岐することができます。

4.4 データ・メモリ (RAM)512ワード×4ビット

データ・メモリは、図4-4のようにデータ・エリアと周辺ハードウェア・エリアによって構成されています。

データ・メモリは、256ワード×4ビットを1バンクとするバンク構成になっており、次のようなメモリ・バンクがあります。

- ・メモリ・バンク0, 1 (データ・エリア)
- ・メモリ・バンク15 (周辺ハードウェア・エリア)

4.4.1 データ・メモリの構成

(1) データ・エリア

データ・エリアは、スタティックRAMで構成され、データのストアやサブルーチン、割り込み実行時のスタック・メモリとして使用します。また、スタンバイ・モードによりCPUの動作を停止させたときでも、バッテリー・バックアップなどでメモリ内容の長時間保持が可能です。操作はメモリ操作命令で行います。

スタティックRAMは、メモリ・バンク0, 1にそれぞれ256×4ビットずつマッピングされています。バンク0は、データ・エリアとしてマッピングされていますが、それ以外にも汎用レジスタ・エリア(000H-01FH)とスタック・エリア^{注1}(000H-1FFH)として使用できます。また、バンク1は表示データ・メモリ(1ECH-1F7H)として使用できます。

スタティックRAMは、1アドレスが4ビット構成です。しかし、8ビット・メモリ操作命令により8ビット単位で操作することも、ビット操作命令によりビット単位に操作することもできます^{注2}。8ビット操作命令では、偶数アドレスを指定してください。

注1. スタック・エリアは、メモリ・バンク0, 1から1つを選択できます。

2. 表示データ・メモリは、8ビット操作はできません。

・汎用レジスタ・エリア

操作は汎用レジスタ操作命令、メモリ操作命令のいずれによっても可能です。最大8本の4ビット・レジスタを使用できます。汎用レジスタ8本のうちプログラムで使用しない部分については、データ・エリアまたはスタック・エリアとして使用できます。

・スタック・エリア

スタック・エリアは命令によって設定し、サブルーチン実行時や、割り込み処理実行時の退避エリアとして使用できます。

・表示データ・メモリ

LCDの表示データを書き込むエリアです。LCD駆動時には、この表示データ・メモリに書き込まれたデータが、ハードウェアにより自動的に読み出され、表示されます。表示に使用していない部分については、データ・エリアとして使用できます。

(2) 周辺ハードウェア・エリア

周辺ハードウェア・エリアは、メモリ・バンク15のF80H-FFFHにマッピングされています。

操作はスタティックRAMと同様にメモリ操作命令で行います。ただし、周辺ハードウェアは、操作可能なビット単位が各アドレスごとに異なります。周辺ハードウェアの割り付けられていないアドレスにはデータ・メモリは内蔵されていないためアクセスできません。

4.4.2 データ・メモリのバンクの指定

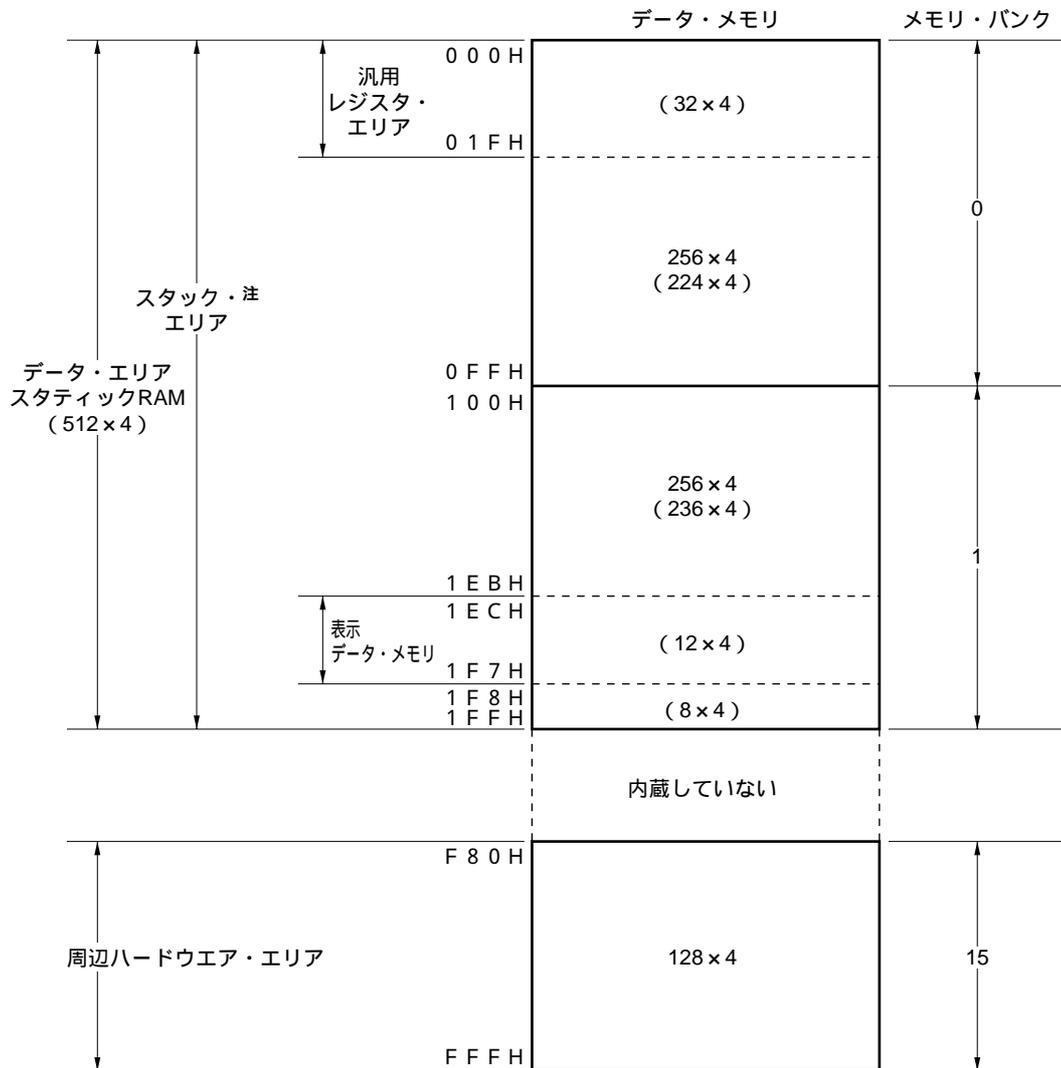
メモリ・バンクは、メモリ・バンク許可フラグ (MBE) によってバンク指定を許可したとき (MBE = 1)、4ビット・メモリ・バンク選択レジスタ (MBS) によって指定されます (MBS = 0, 1, 15)。バンク指定を禁止したとき (MBE = 0)、メモリ・バンクはそのときのアドレッシング・モードによってバンク0またはバンク15が自動的に指定されます。また、バンク内のアドレスは、8ビット・イミューディエト・データやレジスタ・ペアなどによってアドレスされます。

メモリ・バンクの選択とアドレッシングの詳細については、3.1 データ・メモリのバンク構成とアドレッシング・モードを参照してください。

データ・メモリの特定エリアの使用方法については、それぞれ下記を参照してください。

- ・汎用レジスタ・エリア 4.5 汎用レジスタ
- ・スタック・エリア 4.7 スタック・ポインタ (SP)、スタック・バンク選択レジスタ (SBS)
- ・表示データ・メモリ 5.7.6 表示データ・メモリ
- ・周辺ハードウェア・エリア 第5章 周辺ハードウェア機能

図4 - 4 データ・メモリ・マップ



注 スタック・エリアは、メモリ・バンク0, 1から1つを選択できます。

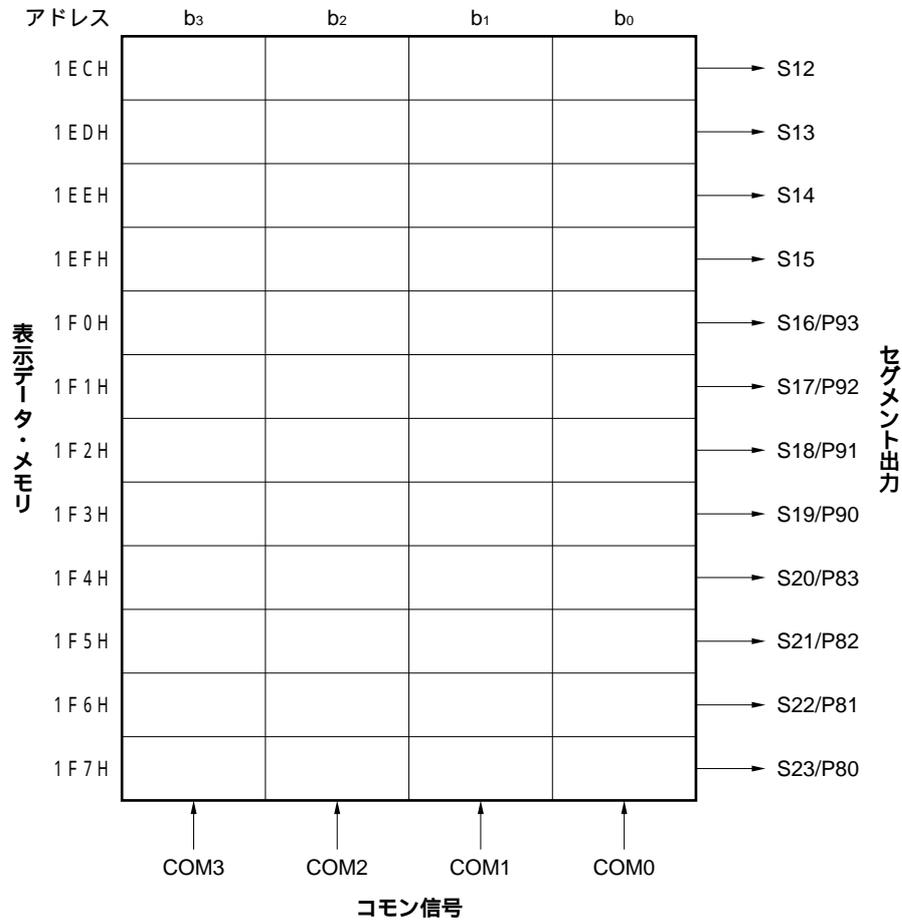
データ・メモリは、リセット時には不定となっています。このため、通常プログラムの初期でゼロにイニシャライズします（RAMクリア）。これを行わないと思わぬバグ発生の原因となりますので必ず行ってください。

例 000H-1FFH番地のRAMをクリアする。

```
SET1  MBE
SEL   MB0
MOV   XA, #00H
MOV   HL, #04H
RAMC0: MOV  @HL, A      ; 04H-FFHをクリア注
      INCS  L          ; L L+1
      BR   RAMC0
      INCS  H          ; H H+1
      BR   RAMC0
      SEL  MB1
RAMC1: MOV  @HL, A      ; 100H-1FFHをクリア
      INCS  L          ; L L+1
      BR   RAMC1
      INCS  H          ; H H+1
      BR   RAMC1
```

注 000H-003Hのデータ・メモリは、汎用レジスタXA，HLとして使用しているため、クリアしません。

図4 - 5 表示データ・メモリの構成



表示データ・メモリは1ビットまたは4ビット単位で操作します。

注意 表示データ・メモリは、8ビット操作はできません。

★ **例** 1ECH-1F7Hの表示データ・メモリをクリアする。

```

SET1  MBE
SEL   MB1
MOV   HL, #0ECH
MOV   A, #00H
LOOP: MOV  @HL, A      ; 4ビット単位で表示データ・メモリを一度に0クリアする。
INCS  HL
SKE   H, #0EH
SKE   L, #08H
BR    LOOP
    
```

4.5 汎用レジスタ..... 8 × 4 ビット × 4 バンク

汎用レジスタはデータ・メモリの特定番地にマッピングされており、4ビット・レジスタ8個（B, C, D, E, H, L, X, A）を1バンクとして4バンクのレジスタがあります。

命令実行時に有効となるレジスタ・バンク（RB）は次の式により決まります。

$$RB = RBE \cdot RBS \quad (RBS = 0-3)$$

汎用レジスタは、それぞれ4ビット単位で操作されるほか、BC, DE, HL, XAがレジスタ・ペアを形成し、8ビット操作に使用されます。また、DE, HLのほかDLもレジスタ・ペアを形成し、この3組はデータ・ポインタとして使用することができます。

8ビット操作時は、BC, DE, HL, XAのほか、レジスタ・バンク（RB）のビット0を反転したレジスタ・バンク（0 1, 2 3）のレジスタ・ペアをBC', DE', HL', XA'として使用できます（3.2 汎用レジスタのバンク構成参照）。

汎用レジスタ・エリアは、レジスタとして使用する、しないにかかわらず、通常のRAMとしてアドレス指定し、アクセスすることができます。

図4 - 6 汎用レジスタの構成

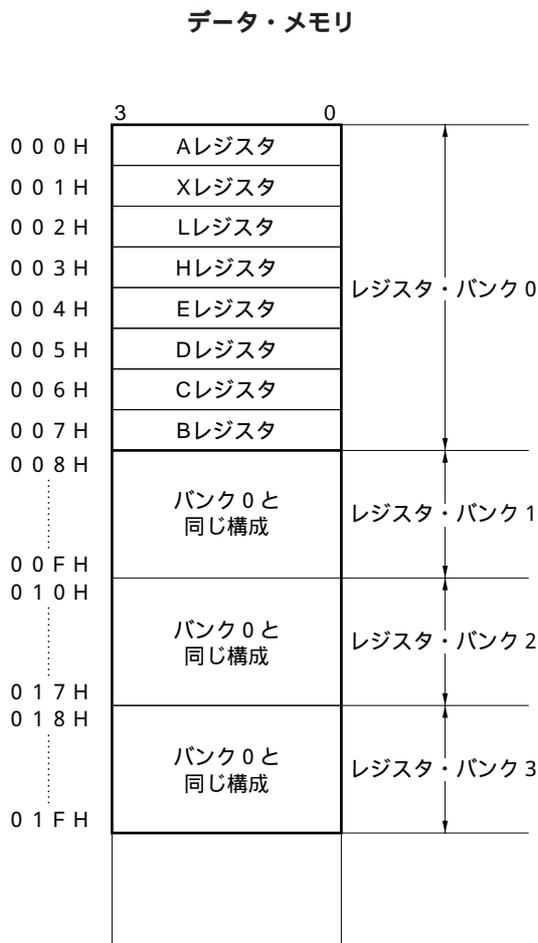
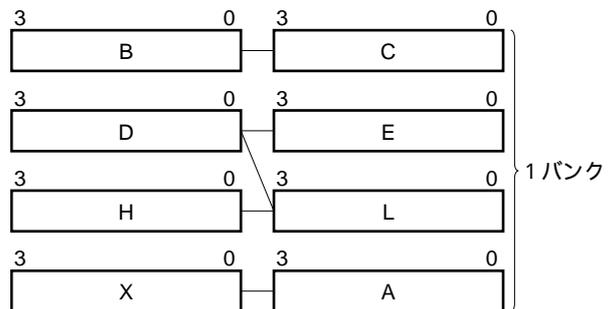


図4 - 7 レジスタ・ペアの構成

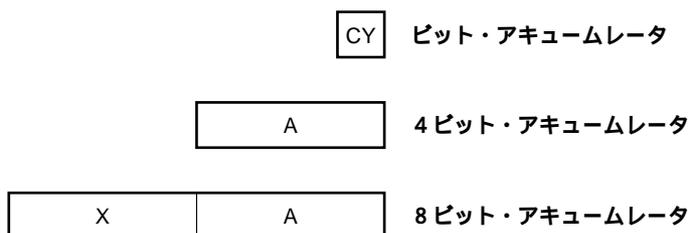


4.6 アキュムレータ

μPD753208では、Aレジスタ、XAレジスタ・ペアがアキュムレータとして機能し、4ビット・データ処理命令ではAレジスタが中心となり、8ビット・データ処理命令ではXAレジスタ・ペアが中心となって実行されます。

ビット操作命令では、キャリー・フラグ (CY) がビット・アキュムレータとして機能します。

図4 - 8 アキュムレータ



4.7 スタック・ポインタ (SP) , スタック・バンク選択レジスタ (SBS)

μPD753208では、スタティックRAMをスタック・メモリ (LIFO形式) として用いており、このスタック・エリアの先頭アドレス情報を保持している8ビット・レジスタがスタック・ポインタ (SP) です。

スタック・エリアは、メモリ・バンク0, 1の000H-1FFH番地です。2ビットのSBSによって1つのメモリ・バンクを指定します (表4 - 2参照)。

表4 - 2 SBSで選択されるスタック・エリア

SBS		スタック・エリア
SBS1	SBS0	
0	0	メモリ・バンク0
0	1	メモリ・バンク1
上記以外		設定禁止

SPは、スタック・メモリへの書き込み（退避）動作に先立ってデクリメントされ、スタック・メモリからの読み取り（復帰）動作後にインクリメントされます。

各スタック動作によって退避／復帰されるデータは図4 - 10～4 - 13のようになります。

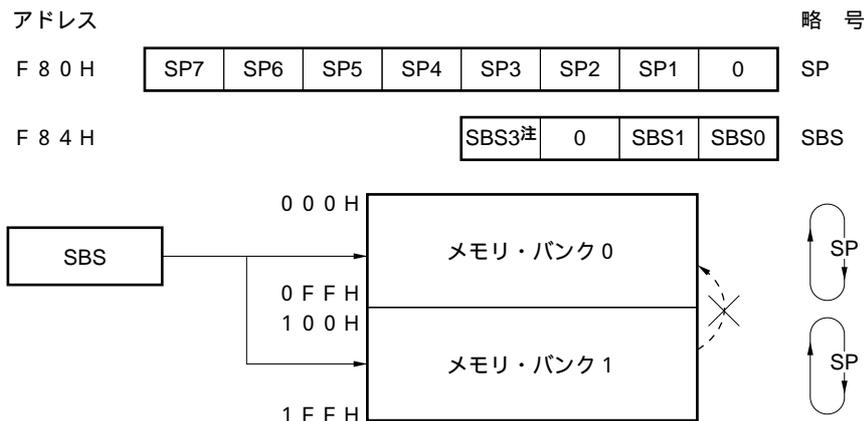
SPは、8ビット・メモリ操作命令、SBSは4ビット・メモリ操作命令によって初期値を設定し、スタック・エリアを決定します。また内容の読み出しもできます。

SPの初期値を00Hに設定すると、SBSで設定したメモリ・バンク（n）の最上位アドレス（nFFH）からスタックされます。

スタック・エリアはSBSで指定したメモリ・バンク内に限られ、n00H番地からさらにスタック動作を行うと、同一バンク内のnFFHへ戻ります。SBSを書き換えずにメモリ・バンクの境界を越えてスタックすることはできません。

$\overline{\text{RESET}}$ 信号発生により、SPの内容は不定に、SBSの内容は1000Bになりますので、プログラムの初期で必ず希望の値にイニシャライズしてください。

図4 - 9 スタック・ポインタおよびスタック・バンク選択レジスタの構成



注 SBS3でMk モードとMk モードの切り替えができます。スタック・バンク選択機能は、Mk モード時とMk モード時の両方で使用することができます（詳しくは4.1 Mk モードとMk モードの切り替え機能を参照してください）。

例 SPのイニシャライズ

スタック・エリアをメモリ・バンク1とし、1FFH番地からスタック動作させる場合

```
SEL    MB15      ;またはCLR1 MBE
MOV    A, #1
MOV    SBS, A    ;メモリ・バンク1をスタック・エリア指定
MOV    XA, #00H
MOV    SP, XA    ;SP 00H
```

図4 - 10 スタック・メモリへ退避されるデータ (Mk モード)

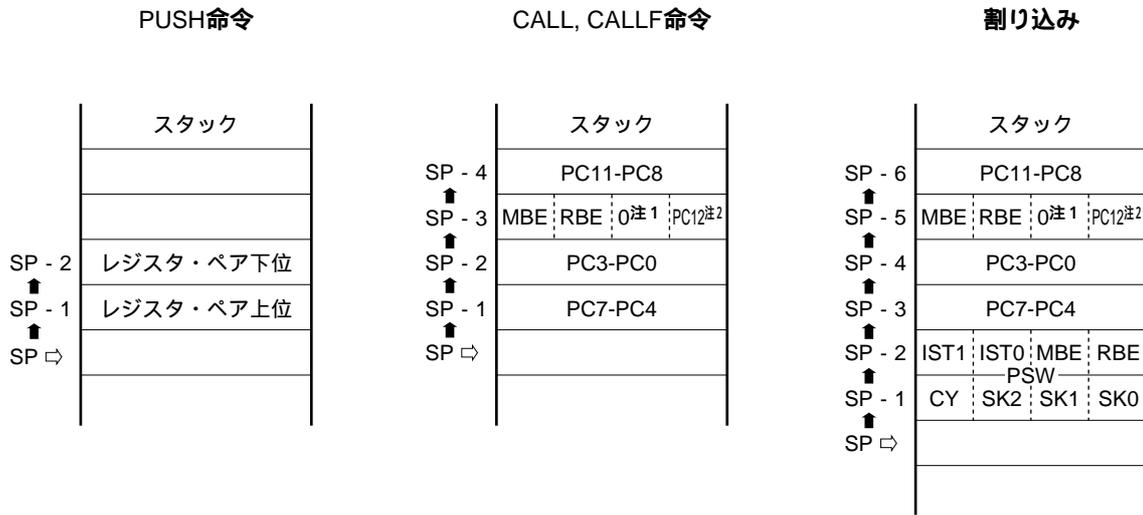
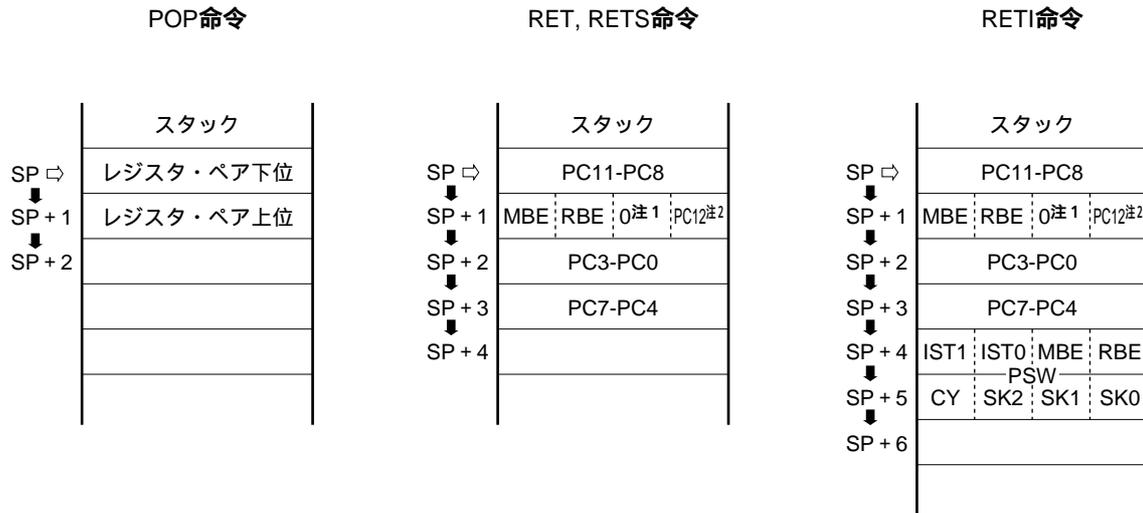


図4 - 11 スタック・メモリから復帰されるデータ (Mk モード)



注1 . μPD75P3216の場合は, PC13が入ります。

2 . μPD753204の場合は, 0が入ります。

図4 - 12 スタック・メモリへ退避されるデータ (Mk モード)

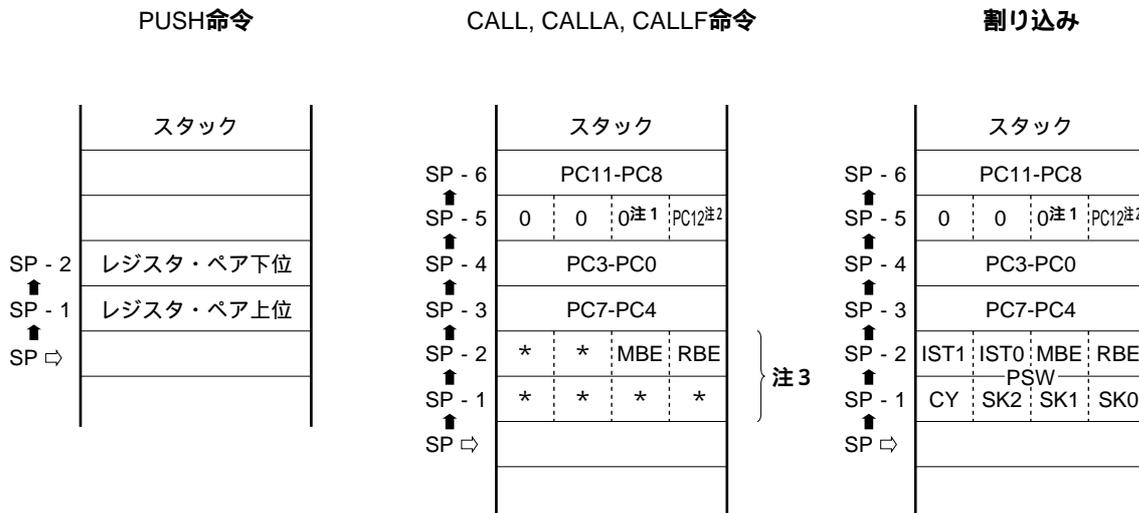
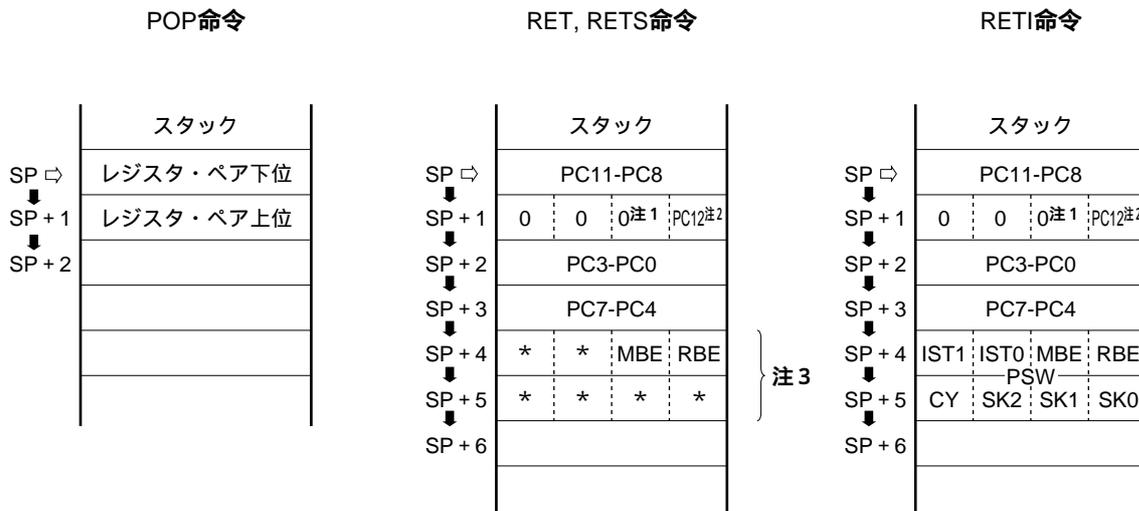


図4 - 13 スタック・メモリから復帰されるデータ (Mk モード)



- 注1 . μPD75P3216の場合は , PC13が入ります。
- 2 . μPD753204の場合は , 0 が入ります。
- 3 . MBE, RBE以外のPSWは退避 / 復帰しません。

備考 * は不定。

4.8 プログラム・ステータス・ワード (PSW) 8 ビット

プログラム・ステータス・ワード (PSW) は、プロセッサ動作と密接に関係する各種のフラグで構成されています。

PSWはデータ・メモリ空間のFB0H, FB1H番地にマッピングされており、FB0H番地の4ビットは、メモリ操作命令で操作できます。

図4 - 14 プログラム・ステータス・ワードの構成



表4 - 3 スタック動作時に退避 / 復帰されるPSWのフラグ

		退避 / 復帰されるフラグ
退 避	CALL, CALLA, CALLF命令実行時	MBE, RBEを退避
	ハードウェア割り込み時	PSW全ビットを退避
復 帰	RET, RETS命令実行時	MBE, RBEを復帰
	RETI命令実行時	PSW全ビットを復帰

(1) キャリー・フラグ (CY)

キャリー・フラグは、キャリー付き演算命令 (ADDC, SUBC) 実行時のオーバフロー、アンダフロー発生情報を記憶する1ビット・フラグです。

また、キャリー・フラグは、ビット・アキュムレータの機能を持っており、ビット・アドレス指定のデータ・メモリとの間でブール代数演算を行い結果をストアすることができます。

キャリー・フラグの操作は、他のPSWのビットとは無関係に専用の命令で行います。

RESET信号発生により、キャリー・フラグは不定となります。

表4 - 4 キャリー・フラグ操作命令

	命令 (ニモニック)	キャリー・フラグの動作, 処理
キャリー・フラグ操作 専用命令	SET1 CY	CYのセット (1)
	CLR1 CY	CYのクリア (0)
	NOT1 CY	CYの内容の反転
	SKT CY	CYの内容が 1 ならばスキップ
ビット転送命令	MOV1 mem*. bit, CY	指定ビットへCYの内容を転送
	MOV1 CY, mem*. bit	指定ビットの内容をCYへ転送
ビット・ブーリアン命令	AND1 CY, mem*. bit	指定ビットの内容とCYの内容とでAND/OR/XORを とり, 結果をCYにセット
	OR1 CY, mem*. bit	
	XOR1 CY, mem*. bit	
割り込み処理	割り込み実行時	他のPSW ビットと 8 ビット並列にスタック・メモ リへ退避
	RETI	スタック・メモリから, 他のPSWと並列に復帰

備考 mem*. bitは次の3つのビット操作アドレッシングを示します。

- ・ fmem. bit
- ・ pmem. @L
- ・ @H + mem. bit

例 3FH番地のビット 3 とP33のANDをとり, P50に出力する。

```
MOV    H, #3H          ;   Hレジスタに上位 4 ビット・アドレス・セット
MOV1   CY, @H+0FH.3   ;   CY 3FHのビット 3
AND1   CY, PORT3.3    ;   CY CY P33
MOV1   PORT5.0, CY    ;   P50 CY
```

(2) スキップ・フラグ (SK2, SK1, SK0)

スキップ・フラグは, スキップ状態を記憶するフラグで, CPUが命令を実行することにより自動的にセット/リセットされます。

ユーザがオペランドとして直接操作することはできません。

(3) 割り込みステータス・フラグ (IST1, IST0)

割り込みステータス・フラグは、現在実行中の処理のステータスを記憶する2ビットのフラグです (詳細は表6-3 IST1, IST0と割り込み処理状態参照)。

表4-5 割り込みステータス・フラグの指示内容

IST1	IST0	実行中の処理のステータス	処理内容と割り込み制御
0	0	ステータス0	通常のプログラム処理中。 すべての割り込みを受け付け可能。
0	1	ステータス1	低位の、または高位の割り込み処理中。 高位の割り込みのみ受け付け可能。
1	0	ステータス2	高位の割り込み処理中。 すべての割り込みの受け付けを禁止。
1	1	-	設定禁止

割り込みプライオリティ・コントロール回路 (図6-1 割り込み制御回路ブロック図参照) は、このフラグ内容を判別して多重割り込みの制御をします。

IST1, 0の内容は、割り込みが受け付けられればPSWの一部としてスタック・メモリに退避されたあと自動的に一段階上のステータスに変更され、RETI命令により割り込みに入る前の値が復帰します。

割り込みステータス・フラグはメモリ操作命令によって操作可能で、実行中の処理ステータスをプログラム制御で変更することもできます。

注意 このフラグを操作する場合は、操作前に必ずDI命令を実行して割り込みを禁止し、操作後にEI命令を実行して割り込みを許可するようにしてください。

(4) メモリ・バンク許可フラグ (MBE)

データ・メモリ・アドレス12ビットのうち、上位4ビットのアドレス情報発生モードを指定する1ビットのフラグです。

MBEは、メモリ・バンクの設定にかかわらず、いつでもビット操作命令によりセット/リセットすることができます。

“1”に設定すると、データ・メモリ・アドレス空間は拡張され、すべてのデータ・メモリ空間がアドレス可能となります。

“0”にリセットすると、データ・メモリ・アドレス空間は、MBSにかかわらず固定されます(図3-2 データ・メモリの構成と、各アドレッシング・モードのアドレッシング範囲参照)。

$\overline{\text{RESET}}$ 信号発生により、プログラム・メモリ0番地のビット7の内容がセットされて、自動的に初期設定されます。

ベクタ割り込み処理時は、該当するベクタ・アドレス・テーブルのビット7の内容がセットされて、割り込みサービス時のMBEの状態が自動的に設定されます。

通常、割り込み処理ではMBE = 0とし、メモリ・バンク0のスタティックRAMを使用します。

(5) レジスタ・バンク許可フラグ (RBE)

汎用レジスタのレジスタ・バンク構成を、拡張するか否かを制御する1ビットのフラグです。

RBEは、メモリ・バンクの設定にかかわらず、いつでもビット操作命令によりセット/リセットすることができます。

“1”にセットすると、レジスタ・バンク選択レジスタ(RBS)の内容により汎用レジスタを、レジスタ・バンク0-3の4組のうちから1組選択できます。

“0”にリセットすると、レジスタ・バンク選択レジスタ(RBS)の内容にかかわらず、常にレジスタ・バンク0が汎用レジスタとして選択されます。

$\overline{\text{RESET}}$ 信号発生により、プログラム・メモリ0番地のビット6の内容がセットされて自動的に初期設定されます。

ベクタ割り込み発生時は、該当するベクタ・アドレス・テーブルのビット6の内容がセットされて、割り込みサービス時のRBEの状態が自動的に設定されます。通常、割り込み処理ではRBE = 0とし、4ビット処理ではレジスタ・バンク0を、8ビット処理ではレジスタ・バンク0, 1を使用します。

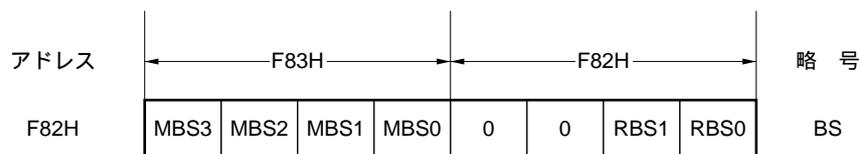
4.9 バンク選択レジスタ (BS)

バンク選択レジスタ (BS) は、レジスタ・バンク選択レジスタ (RBS) とメモリ・バンク選択レジスタ (MBS) とで構成され、それぞれ使用するレジスタ・バンクとメモリ・バンクを指定します。

RBS, MBSは、それぞれSEL RBn, SEL MBn命令で設定されます。

BSは、PUSH BS/POP BS命令により、8ビット単位でスタック・エリアへ退避/復帰することができます。

図4 - 15 バンク選択レジスタの構成



(1) メモリ・バンク選択レジスタ (MBS)

メモリ・バンク選択レジスタは、データ・メモリ・アドレス (12ビット) 上位4ビット・アドレス情報を記憶する4ビット・レジスタで、このレジスタの内容によりアクセスするメモリ・バンクを指定します。ただし、μPD753208では、バンク0, 1, 15のみ指定できます。

MBSは、SEL MBn命令により設定します (n=0, 1, 15)。

MBEとMBSの設定に対するアドレス範囲は図3 - 2に示すようになります。

RESET信号発生により、MBSは“0”にイニシャライズされます。

(2) レジスタ・バンク選択レジスタ (RBS)

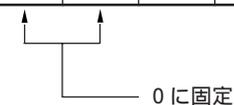
レジスタ・バンク選択レジスタは、汎用レジスタとして使用するレジスタ・バンクを指定するレジスタで、バンク0からバンク3まで設定できます。

RBSは、SEL RBn命令により設定します (n=0-3)。

RESET信号発生により、RBSは“0”にイニシャライズされます。

表4 - 6 RBE, RBSと選択されるレジスタ・バンク

RBE	RBS				レジスタ・バンク
	3	2	1	0	
0	0	0	x	x	バンク0に固定
1	0	0	0	0	バンク0を選択
			0	1	バンク1 "
			1	0	バンク2 "
			1	1	バンク3 "



x : don't care

〔メモ〕

第5章 周辺ハードウェア機能

5.1 デジタル入出力ポート

μPD753208ではメモリ・マップトI/Oが採用されており，すべての入出力ポートはデータ・メモリ空間上にマッピングされています。

図5 - 1 デジタル・ポートのデータ・メモリ・アドレス

アドレス	3	2	1	0	
FF 0 H	P03	P02	P01	P00	PORT0
FF 1 H	P13	-	-	P10	PORT1
FF 2 H	P23	P22	P21	P20	PORT2
FF 3 H	P33	P32	P31	P30	PORT3
FF 5 H	P53	P52	P51	P50	PORT5
FF 6 H	P63	P62	P61	P60	PORT6
FF 8 H	P83	P82	P81	P80	PORT8
FF 9 H	P93	P92	P91	P90	PORT9

入出力ポートの操作命令は表5 - 2のようになっており，PORT8, PORT9については4ビット入出力のほか8ビット入出力，ビット操作が可能で，きわめて多様な制御ができます。

例1．P13の状態をテストして，その結果によって異なった値をポート8，9に出力。

```
SKT  PORT1.3    ; ポート1のビット3が1ならスキップ。
MOV  XA, #18H   ; XA 18H }
MOV  XA, #14H   ; XA 14H } たてづみ
SEL  MB15       ; または，CLR1 MBE
OUT  PORT8, XA  ; ポート9，8 XA
```

2．SET1 PORT8.@L；ポート8，9のうち，Lレジスタで指定されるビットを“1”にセット。

5.1.1 デジタル入出力ポートの種類，特徴，構成

デジタル入出力ポートには表5 - 1のような種類があります。

各ポートの構成は図5 - 2 ~ 5 - 6のようになっています。

表5 - 1 デジタル・ポートの種類と特徴

ポート(端子名)	機能	動作，特徴	備考
PORT0 (P00-P03)	4ビット入力	シリアル・インタフェース機能使用時は，動作モードによって兼用端子が出力機能を持ちます。	INT4, \overline{SCK} , SO/SB0, SI/SB1と端子を兼用。
PORT1 (P10, P13)	2ビット入力	2ビット入力専用ポート。	INT0, T10と端子を兼用。
PORT2 (P20-P23)	4ビット入出力	4ビット単位で入力または出力モードに設定可能。	PTO0-PTO2, PCL, BUZと端子を兼用。
PORT3 (P30-P33)		1ビット単位で入力または出力モードに設定可能。	LCDCL, SYNC, MD0-MD3 ^{注1} と端子を兼用。
★ PORT5 (P50-P53)	4ビット入出力 (N-chオープン・ドレイン13V耐圧)	4ビット単位で入力または出力モードに設定可能。マスク・オプション ^{注2} で，1ビット単位でプルアップ抵抗内蔵の指定可能。	D4-D7 ^{注1} と端子を兼用。
★ PORT6 (P60-P63)	4ビット入出力	1ビット単位で入力または出力モードに設定可能。	KR0-KR3, D0-D3 ^{注1} と端子を兼用。
PORT8 (P80-P83)		4ビット単位で入力または出力モードに設定可能。ペアとなって8ビット単位でデータの入出力が可能。	S20-S23と端子を兼用。
PORT9 (P90-P93)			S16-S19と端子を兼用。

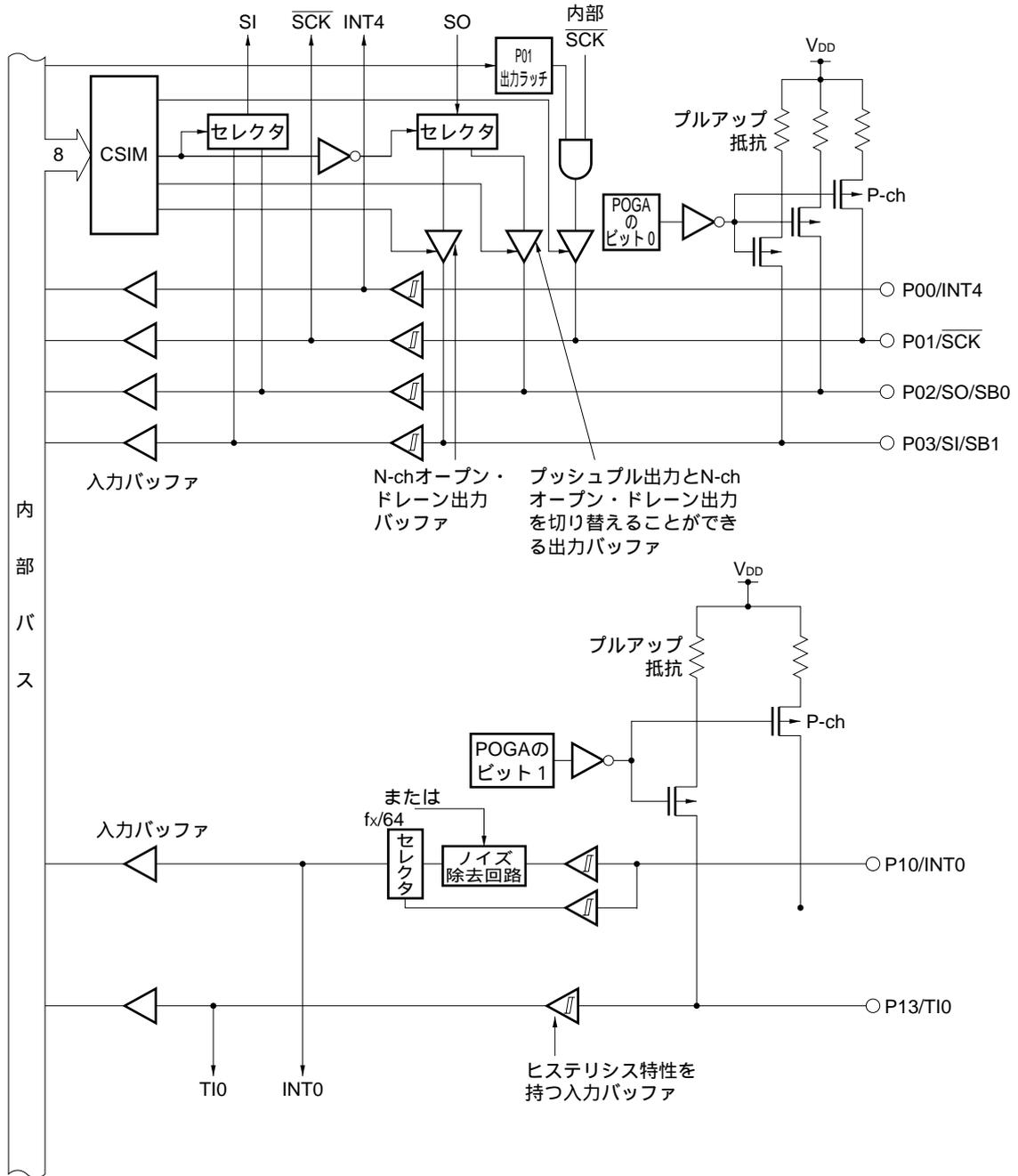
注1． μ PD75P3216のときのみ兼用している端子です。

2． μ PD75P3216には，マスク・オプションによるプルアップ抵抗の内蔵機能はありません。

P10は，外部ベクタ割り込み入力端子と兼用になっており，ノイズ除去回路付き入力となっています（詳細は6.3 割り込み機能を制御する各種ハードウェアを参照）。

★

図5-2 ポート0, 1の構成



★

図5-3 ポート3, 6の構成

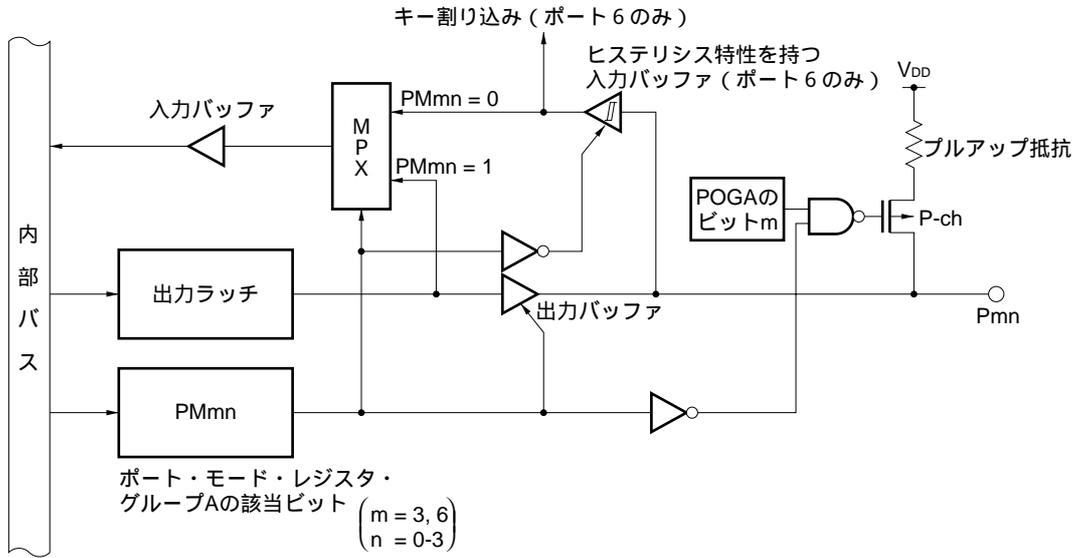


図5-4 ポート2の構成

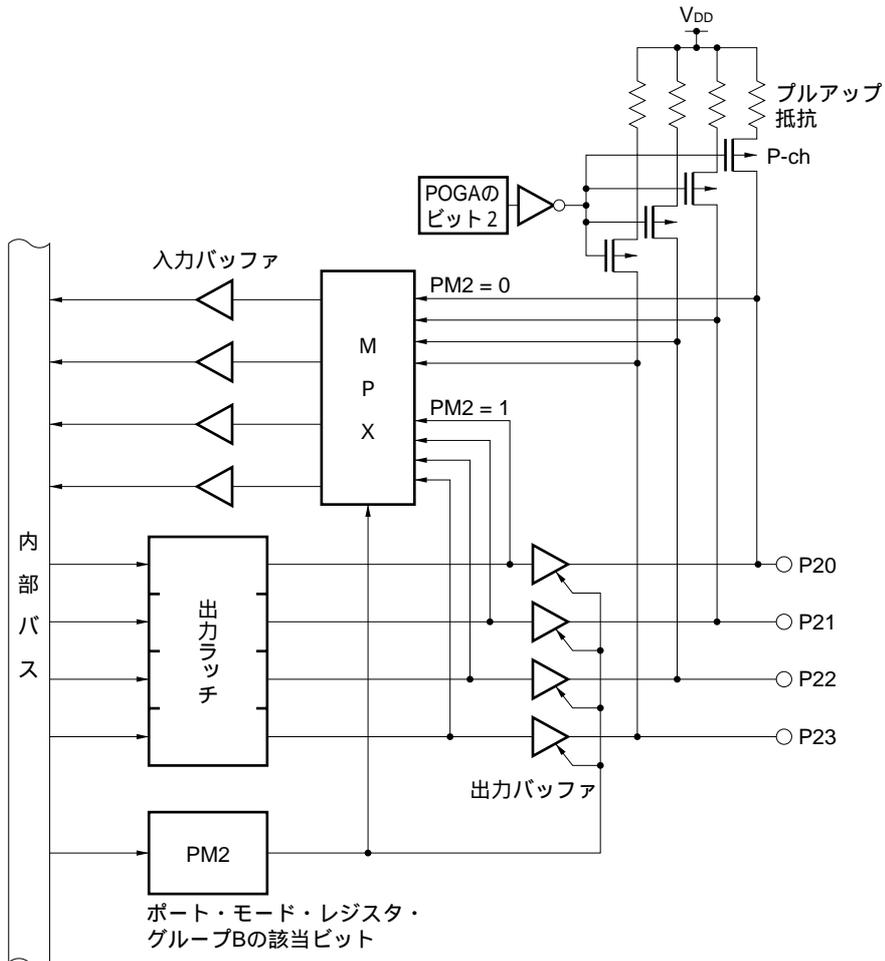


図5 - 5 ポート5の構成

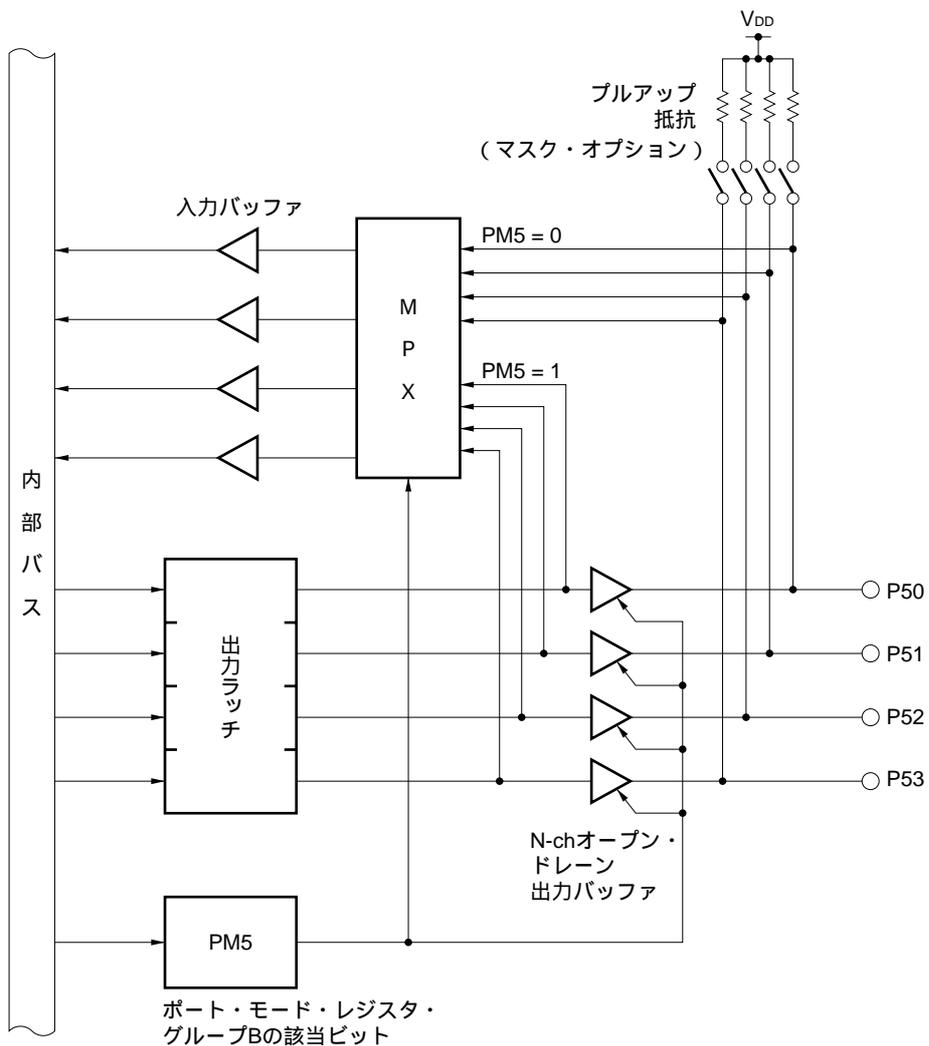
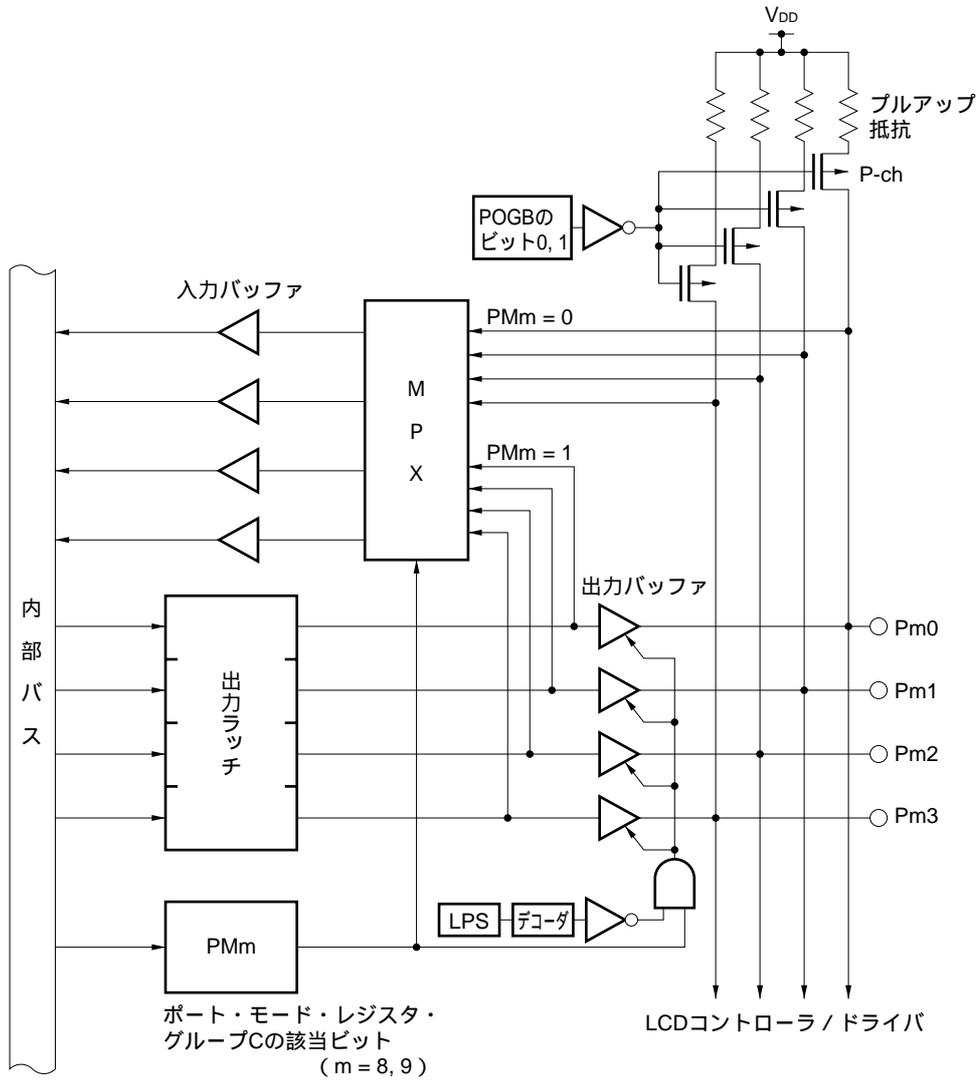


図5-6 ポート8, 9の構成



5.1.2 入出力モードの設定

各入出力ポートの入出力モードは、図5-7のようにポート・モード・レジスタにより設定されます。ポート3, 6はポート・モード・レジスタ・グループA (PMGA) によって、ビットごとに入出力を指定できます。ポート2, 5はポート・モード・レジスタ・グループB (PMGB) によって、ポート8, 9はポート・モード・レジスタ・グループC (PMGC) によって、4ビット単位で入出力を指定します。

各ポートは、対応するポート・モード・レジスタのビットが“0”のとき入力ポートとして働き、“1”のとき出力ポートとして働きます。

ポート・モード・レジスタの設定により出力モードが選択されると同時に出力ラッチの内容が、出力端子に出力されますから、出力モードを設定する前にあらかじめ、出力ラッチの内容を必要な値に書き換えておく必要があります。

ポート・モード・レジスタ・グループA, B, Cは、それぞれ8ビット・メモリ操作命令で設定します。

$\overline{\text{RESET}}$ 信号発生により、各ポート・モード・レジスタは全ビット0にクリアされるため出力バッファはオフし、全ポートは入力モードになります。

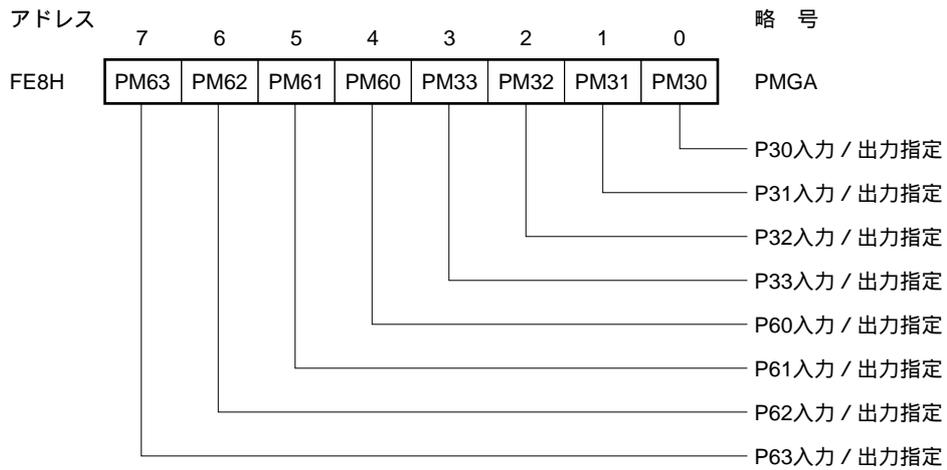
例 P30, 31, 62, 63を入力, P32, 33, 60, 61を出力端子として使う。

```
CLR1  MBE          ;またはSEL MB15
MOV   XA, #3CH
MOV   PMGA, XA
```

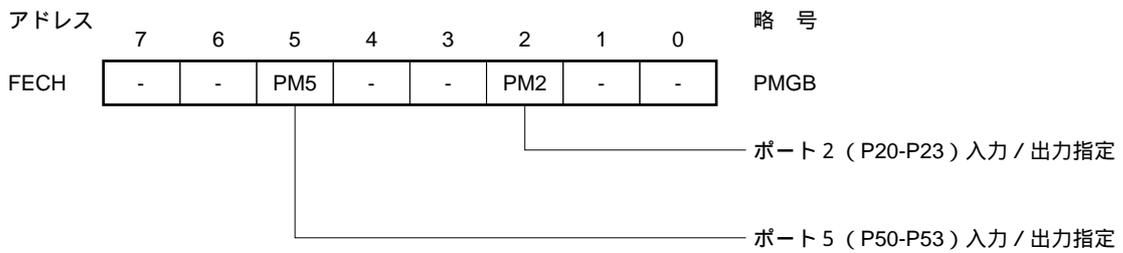
図5 - 7 各ポート・モード・レジスタのフォーマット

	指 定 内 容
0	入力モード (出力バッファ・オフ)
1	出力モード (出力バッファ・オン)

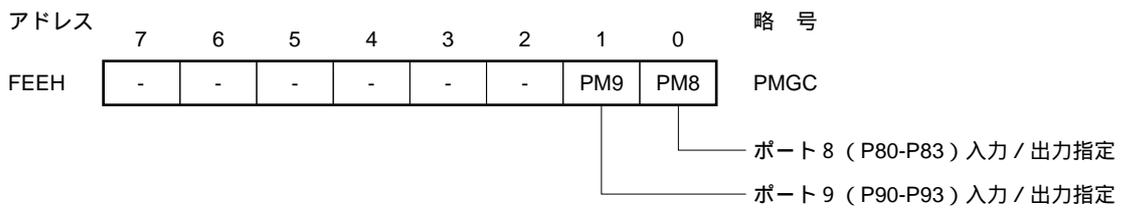
ポート・モード・レジスタ・グループA



ポート・モード・レジスタ・グループB



ポート・モード・レジスタ・グループC



5.1.3 デジタル入出力ポート操作命令

μPD753208に内蔵されている入出力ポートはすべてデータ・メモリ空間にマッピングされていますから、データ・メモリ操作命令はすべて適用することができます。データ・メモリ操作命令のうち、特に入出力端子の操作に有効と思われる命令とその適用範囲を表5 - 2にまとめます。

(1) ビット操作命令

デジタル入出力ポートPORT0-3, 5, 6, 8, 9は、特定アドレス・ビット直接アドレッシング (fmem. bit) と特定アドレス・ビット・レジスタ間接アドレッシング (pmem. @L) が適用できますから、MBE, MBSの指定にかかわらずいつでも、自由に、ポートのビット操作が可能です。

例 P50とP51のORをとってP61に出力する。

```
MOV1    CY, PORT5.0        ; CY P50
OR1     CY, PORT5.1        ; CY CY P51
MOV1    PORT6.1, CY        ; P61 CY
```

(2) 4ビット操作命令

IN/OUT命令のほか、MOV, XCH, ADDS, INCSなど4ビット・メモリ操作命令がすべて使用できますが、命令の実行に先立ってメモリ・バンク15を選択しておく必要があります。

例1 . ポート3に、アキュムレータの内容を出力する。

```
SET1    MBE
SEL     MB15                ;またはCLR1 MBE
OUT     PORT3, A
```

2 . ポート5に出力しているデータにアキュムレータの値を加えて出力する。

```
SET1    MBE
SEL     MB15
MOV     HL, #PORT5
ADDS   A, @HL                ; A A + PORT5
NOP
MOV     @HL, A                ; PORT5 A
```

3 . ポート5のデータがアキュムレータの値より大きいかどうかテストする。

```
SET1    MBE
SEL     MB15
MOV     HL, #PORT5
SUBS   A, @HL                ; A < PORT5
BR     NO                    ; NO
                        ; YES
```

(3) 8ビット操作命令

8ビット操作が可能なポート8, 9に対しIN/OUT命令のほか, MOV/XCH/SKE命令が使用できます。4ビット操作時と同様に, あらかじめ, メモリ・バンク15を選択しておく必要があります。

例 ポート8, 9から入力された8ビット・データで指定される出力ポートにBCレジスタ・ペアのデータを出力する。

```
SET1    MBE
SEL      MB15
IN       XA, PORT8          ; XA ポート9, 8
MOV      HL, XA             ; HL XA
MOV      XA, BC             ; XA BC
MOV      @HL, XA           ; ポート(L) XA
```

★

表5 - 2 入出力端子操作命令一覧表

命令 \ PORT	PORT 0	PORT 1	PORT 2	PORT 3	PORT 5	PORT 6	PORT 8	PORT 9
IN A, PORTn 注1								
IN XA, PORTn 注1	-	-	-	-	-	-	-	-
OUT PORTn, A 注1	-	-	-	-	-	-	-	-
OUT PORTn, XA 注1	-	-	-	-	-	-	-	-
MOV A, PORTn 注1								
MOV XA, PORTn 注1	-	-	-	-	-	-	-	-
MOV PORTn, A 注1								
MOV PORTn, XA 注1	-	-	-	-	-	-	-	-
XCH A, PORTn 注1								
XCH XA, PORTn 注1	-	-	-	-	-	-	-	-
MOV1 CY, PORTn. bit								
MOV1 CY, PORTn. @L 注2								
MOV1 PORTn. bit, CY	-	-	-	-	-	-	-	-
MOV1 PORTn. @L, CY 注2	-	-	-	-	-	-	-	-
INCS PORTn 注1								
SET1 PORTn. bit	-	-	-	-	-	-	-	-
SET1 PORTn. @L 注2	-	-	-	-	-	-	-	-
CLR1 PORTn. bit	-	-	-	-	-	-	-	-
CLR1 PORTn. @L 注2	-	-	-	-	-	-	-	-
SKT PORTn. bit								
SKT PORTn. @L 注2								
SKF PORTn. bit								
SKF PORTn. @L 注2								
SKTCLR PORTn. bit								
SKTCLR PORTn. @L 注2								
AND1 CY, PORTn. bit								
AND1 CY, PORTn. @L 注2								
OR1 CY, PORTn. bit								
OR1 CY, PORTn. @L 注2								
XOR1 CY, PORTn. bit								
XOR1 CY, PORTn. @L 注2								

注1 . 実行前に , あらかじめMBE = 0, または (MBE = 1, MBS = 15) としておく必要があります。

2 . アドレスの下位2ビットとビット・アドレスをLレジスタで間接指定します。

5.1.4 デジタル入出力ポートの動作

デジタル入出力ポートに対し、データ・メモリ操作命令を実行したときの各ポートおよび、各端子の動作は、入出力モードの設定によって異なります(表5-3参照)。これは入出力ポートの構成からもわかるように、内部バスに取り込まれるデータが、入力モードでは各端子のデータ、出力モードでは出力ラッチのデータになっているためです。

(1) 入力モードに設定されているときの動作

SKT命令のようなテスト命令、MOV1命令によるビット入力命令、あるいは4、8ビットで、ポート・データを内部バスに取り込む命令(IN, MOV, 演算命令, 比較命令)が実行されたときは、各端子のデータが操作されます。

4、8ビットでアキュムレータの内容をポートに転送する命令(OUT, MOV命令)が実行されたときは、アキュムレータのデータが出力ラッチにラッチされます。出力バッファはオフしたままです。

XCH命令が実行されたときは、各端子のデータがアキュムレータに入力され、アキュムレータのデータは出力ラッチにラッチされます。出力バッファはオフしたままです。

INCS命令を実行すると、各端子のデータ(4ビット)に+1したデータが、出力ラッチにラッチされます。出力バッファはオフしたままです。

SET1/CLR1/MOV1/SKTCLR命令のように、ビット単位でデータ・メモリを書き換える命令を実行したときは、指定したビットの出力ラッチは、命令の指示どおりに書き換えることができますが、その他のビットの出力ラッチの内容が不定となります。

(2) 出力モードに設定されているときの動作

テスト命令、ビット入力命令、4、8ビットでポート・データを内部バスに取り込む命令を実行すると、出力ラッチの内容が操作されます。

4、8ビットでアキュムレータの内容を転送する命令が実行されたときは、出力ラッチのデータが書き換えられると同時に端子から出力されます。

XCH命令が実行されたときは、出力ラッチの内容がアキュムレータに転送され、アキュムレータの内容は、出力ラッチにラッチされ、端子から出力されます。

INCS命令を実行すると、出力ラッチの内容を+1したデータが出力ラッチにラッチされ、端子から出力されます。

ビット出力命令を実行すると、指定された出力ラッチのビットが書き換えられ、端子から出力されません。

表5 - 3 入出力ポート操作時の動作

実行する命令	ポートおよび各端子の動作	
	入力モード	出力モード
SKT <input type="checkbox"/> SKF <input type="checkbox"/>	端子のデータをテスト	出力ラッチのデータをテスト
MOV1 CY, <input type="checkbox"/>	端子のデータをCYへ転送	出力ラッチのデータをCYへ転送
AND1 CY, <input type="checkbox"/> OR1 CY, <input type="checkbox"/> XOR1 CY, <input type="checkbox"/>	端子のデータとCYの間で演算	出力ラッチのデータとCYの間で演算
IN A, PORTn IN XA, PORTn ★ MOV A, PORTn ★ MOV XA, PORTn MOV A, @HL MOV XA, @HL	端子のデータをアキュムレータへ転送	出力ラッチのデータをアキュムレータへ転送
ADDS A, @HL ADDC A, @HL SUBS A, @HL SUBC A, @HL AND A, @HL OR A, @HL XOR A, @HL	端子のデータとアキュムレータの間で演算	出力ラッチのデータとアキュムレータの間で演算
SKE A, @HL SKE XA, @HL	端子のデータとアキュムレータを比較	出力ラッチのデータとアキュムレータを比較
★ OUT PORTn, A ★ OUT PORTn, XA ★ MOV PORTn, A ★ MOV PORTn, XA MOV @HL, A MOV @HL, XA	出力ラッチにアキュムレータのデータを転送（出力バッファはオフのまま）	出力ラッチにアキュムレータのデータを転送し、端子から出力
XCH A, PORTn XCH XA, PORTn XCH A, @HL XCH XA, @HL	端子のデータをアキュムレータに転送し、アキュムレータのデータは出力ラッチに転送（出力バッファはオフのまま）	出力ラッチとアキュムレータの間でデータを交換
INCS PORTn INCS @HL	端子のデータを + 1 したデータを出力ラッチにラッチ	出力ラッチの内容を + 1
SET1 <input type="checkbox"/> CLR1 <input type="checkbox"/> MOV1 <input type="checkbox"/> , CY SKTCLR <input type="checkbox"/>	指定したビットの出力ラッチは命令の指示通り書き替わるが、その他のビットの出力ラッチは不定	出力端子の状態を命令に従って変更

備考 : PORTn. bitとPORTn. @Lの2つのアドレッシング・モードを示します。

5.1.5 プルアップ抵抗の内蔵

μPD753208の各ポート端子にはプルアップ抵抗を内蔵することができます（ただしP00端子を除く）。プルアップ抵抗の内蔵の指定はソフトウェアによる端子と、マスク・オプションによる端子があります。

各ポート端子の指定法を表5 - 4に示します。また、ソフトウェアによる内蔵プルアップ抵抗の接続指定は、図5 - 8に示したフォーマットに基づいて行います。

ポート3, 6に対する内蔵プルアップ抵抗の接続指定は、入力モードに指定されている端子にのみ有効です。出力モードに指定された端子は、POGA, POGBの設定にかかわらず、内蔵プルアップ抵抗の接続を指定できません。

表5 - 4 プルアップ抵抗内蔵の指定法

ポート（端子名）	プルアップ抵抗内蔵の指定方法	指定ビット
ポート0（P01-P03） ^{注1}	3ビット単位でソフトウェアにより接続を指定	POGA.0
ポート1（P10, P13）	2ビット単位でソフトウェアにより接続を指定	POGA.1
ポート2（P20-P23）	4ビット単位でソフトウェアにより接続を指定	POGA.2
ポート3（P30-P33）		POGA.3
ポート5（P50-P53）	1ビット単位でマスク・オプションにより内蔵指定	-
ポート6（P60-P63）	4ビット単位でソフトウェアにより接続を指定	POGA.6
ポート8（P80-P83） ^{注2}		POGB.0
ポート9（P90-P93） ^{注2}		POGB.1

注1．P00端子には内蔵プルアップ抵抗の接続を指定できません。

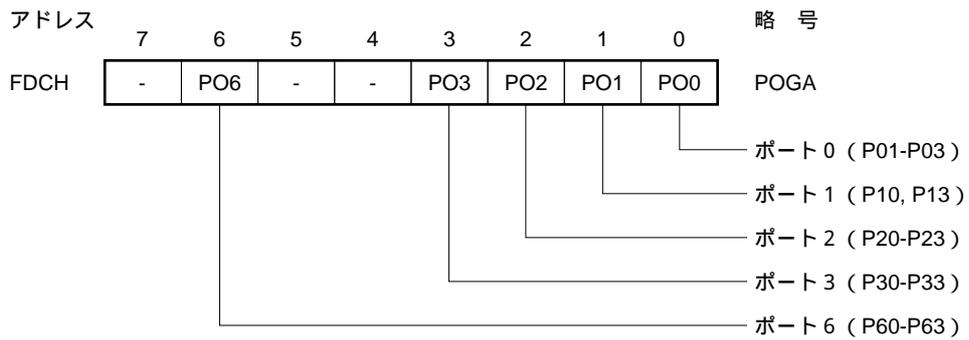
- 2．セグメント信号出力として使用する場合は、ソフトウェアによる内蔵プルアップ抵抗の接続を指定しないでください。

備考 μPD75P3216は、マスク・オプションによるプルアップ抵抗の内蔵機能はありません。

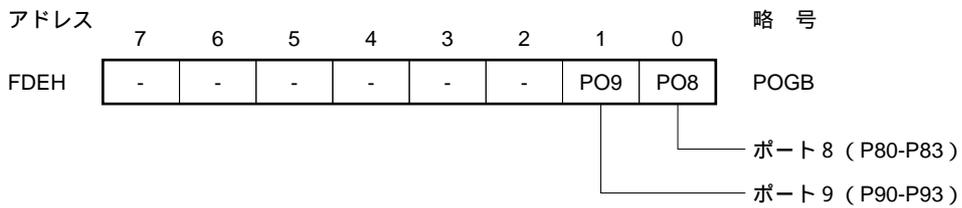
図5 - 8 プルアップ抵抗レジスタのフォーマット

	指 定 内 容
0	内蔵プルアップ抵抗の接続を指定しない
1	内蔵プルアップ抵抗の接続を指定する

プルアップ抵抗レジスタ・グループA



プルアップ抵抗レジスタ・グループB



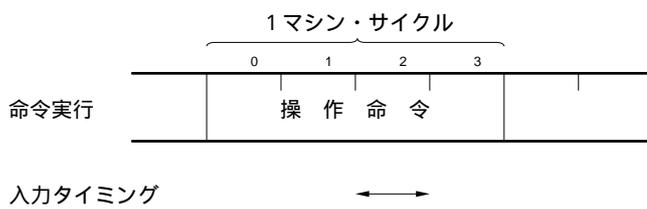
5.1.6 デジタル入出力ポートの入出力タイミング

出力ラッチにデータが出力されるタイミングおよび端子のデータまたは、出力ラッチのデータが内部バスに取り込まれるタイミングは図5 - 9に示すようになっています。

また、ソフトウェアにより内蔵プルアップ抵抗の接続を指定した場合のONタイミングを図5 - 10に示します。

図5 - 9 デジタル入出力ポートの入出力タイミング

★ (a) 1マシン・サイクル命令によってデータが取り込まれる場合



★ (b) 2マシン・サイクル命令によってデータが取り込まれる場合



(c) 1マシン・サイクル命令によってデータがラッチされる場合

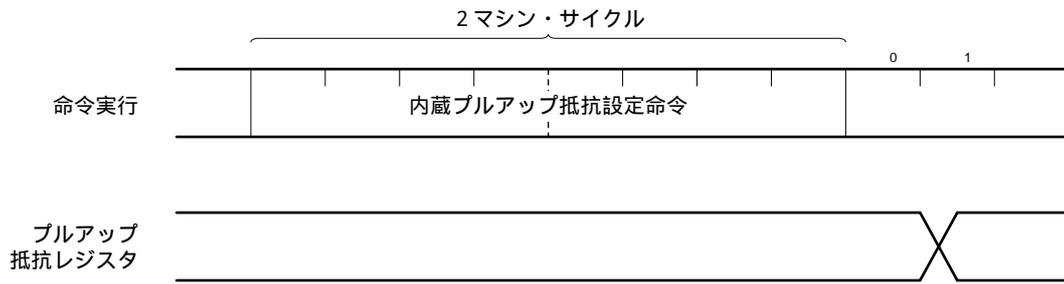


(d) 2マシン・サイクル命令によってデータがラッチされる場合



★

図5 - 10 ソフトウェアによる内蔵プルアップ抵抗のONタイミング



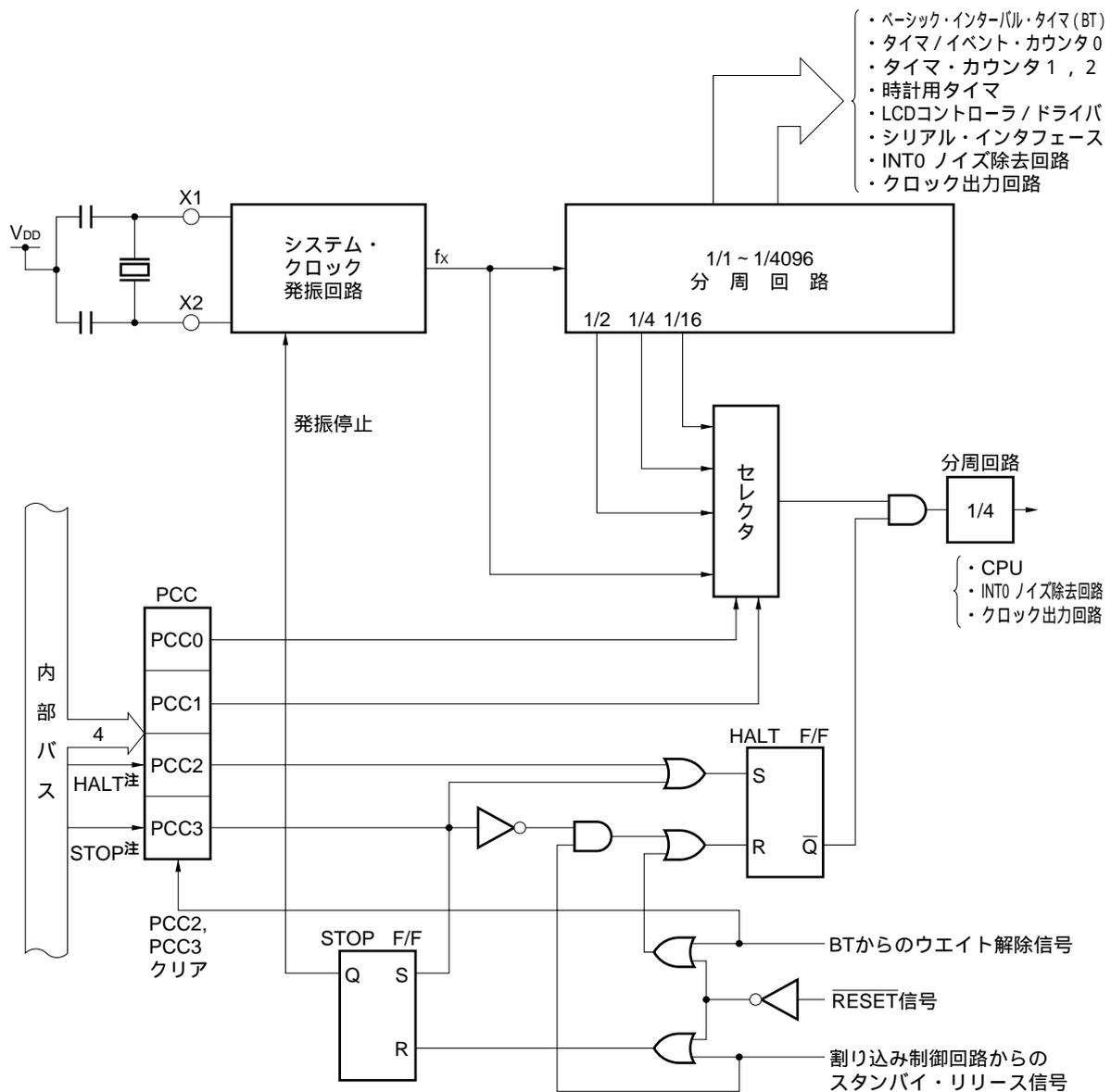
5.2 クロック発生回路

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

5.2.1 クロック発生回路の構成

クロック発生回路は図5-11のように構成されています。

図5-11 クロック発生回路のブロック図



注 命令の実行

備考1 . fx = システム・クロック周波数

2 . = CPUクロック

3 . PCC : プロセッサ・クロック・コントロール・レジスタ

4 . の1クロック・サイクル (tc_v) は命令の1マシン・サイクルです。

5.2.2 クロック発生回路の機能と動作

クロック発生回路は次に示す各種のクロックを発生し、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。

- ・システム・クロック f_x
- ・CPUクロック
- ・周辺ハードウェアへのクロック

クロック発生回路の動作はプロセッサ・クロック・コントロール・レジスタ (PCC) により決定され、以下のような機能、動作となります。

- (a) $\overline{\text{RESET}}$ 信号発生によりシステム・クロックの最低速モード (10.7 μs : 6.00 MHz動作時) が選択されます (PCC = 0)。
- (b) PCCの設定により 4 段階のCPUクロックを選択することができます (0.67 , 1.33 , 2.67 , 10.7 μs : 6.00 MHz動作時)。
- (c) STOPモード , HALTモードの 2 つのスタンバイ・モードが使用できます。
- (d) 周辺ハードウェアへのクロックはシステム・クロックを分周して供給されます。
- (e) シリアル・インタフェースやタイマ/イベント・カウンタおよびタイマ・カウンタは、クロックとして外部クロックを選択した場合、動作を継続することができます。しかし、ほかのハードウェアはシステム・クロックによって動作しますので、システム・クロックを停止させたときは使用することはできません。

(1) プロセッサ・クロック・コントロール・レジスタ (PCC)

PCCは、下位2ビットでCPUクロック の選択、上位2ビットでCPU動作モードの制御を行う4ビット・レジスタです(図5-12参照)。

ビット3またはビット2の片方のみが“1”にセットされると、スタンバイ・モードに設定されます。スタンバイ・リリース信号によって解除されると、両ビットは自動的にクリアされ通常の動作モードとなります(詳細は第7章 **スタンバイ機能**を参照)。

PCCの下位2ビットは、4ビット・メモリ操作命令により設定します(上位2ビットは“0”としてください)。

また、ビット3とビット2は、それぞれSTOP命令、HALT命令により“1”にセットします。STOP命令、HALT命令は、MBEの内容に関係なく常に実行できます。

例1 . マシン・サイクルを最高速モード(0.67 μ s : $f_x = 6.00$ MHz動作時)にする。

```
SEL  MB15
MOV  A, #0011B
MOV  PCC, A
```

2 . マシン・サイクルを1.91 μ s ($f_x = 4.19$ MHz動作時)にする。

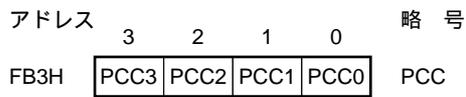
```
SEL  MB15
MOV  A, #0010B
MOV  PCC, A
```

3 . STOPモードに設定する(STOP命令およびHALT命令のあとには必ずNOP命令を書いてください)。

```
STOP
NOP
```

$\overline{\text{RESET}}$ 信号発生によりPCCは“0”にクリアされます。

図5 - 12 プロセッサ・クロック・コントロール・レジスタのフォーマット



CPU動作モード制御ビット

PCC3	PCC2	動作モード
0	0	通常動作モード
0	1	HALTモード
1	0	STOPモード
1	1	設定禁止

CPUクロック選択ビット

($f_x = 6.0 \text{ MHz}$ の場合)

PCC1	PCC0	CPUクロック周波数	1マシン・サイクル
0	0	$= f_x/64$ (93.8 kHz)	10.7 μs
0	1	$= f_x/16$ (375 kHz)	2.67 μs
1	0	$= f_x/8$ (750 kHz)	1.33 μs
1	1	$= f_x/4$ (1.5 MHz)	0.67 μs

($f_x = 4.19 \text{ MHz}$ の場合)

PCC1	PCC0	CPUクロック周波数	1マシン・サイクル
0	0	$= f_x/64$ (65.5 kHz)	15.3 μs
0	1	$= f_x/16$ (262 kHz)	3.81 μs
1	0	$= f_x/8$ (524 kHz)	1.91 μs
1	1	$= f_x/4$ (1.05 MHz)	0.95 μs

備考 f_x : システム・クロック発振回路出力周波数

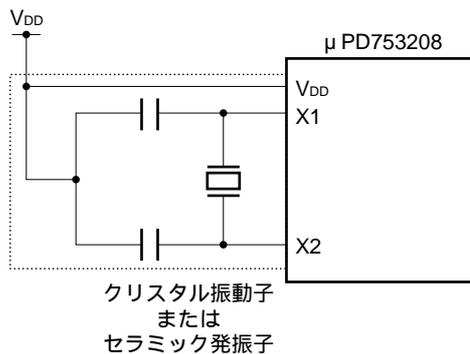
(2) システム・クロック発振回路

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によって発振します(標準: 4.194304 MHz)。

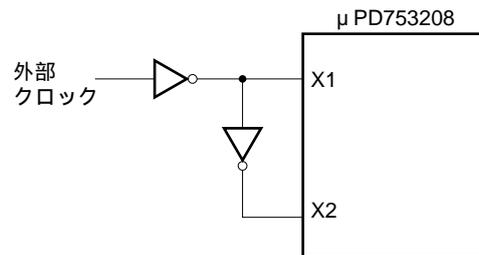
また, 外部クロックを入力することもできます。

図5 - 13 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 . STOPモードに設定すると, X2端子が内部でV_{DD}に50 k (TYP.)の抵抗でプルアップされます。

2 . 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 - 13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。

ほかの信号線と交差させない。

変化する大電流が流れる線と接近させない。

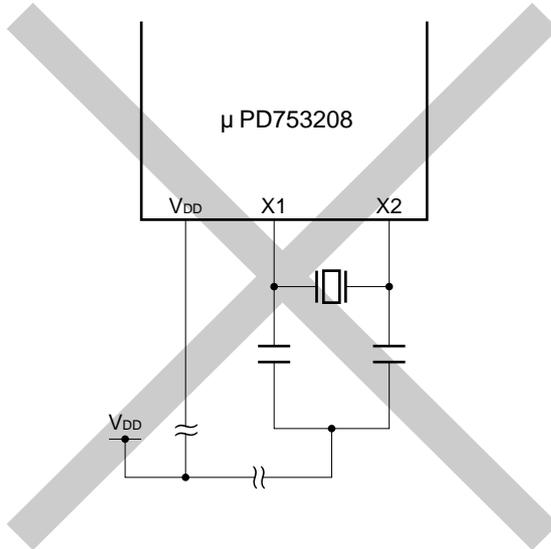
発振回路のコンデンサの接続点は, 常にV_{DD}と同電位となるようにする。大電流が流れる電源パターンに接続しない。

発振回路から信号を取り出さない。

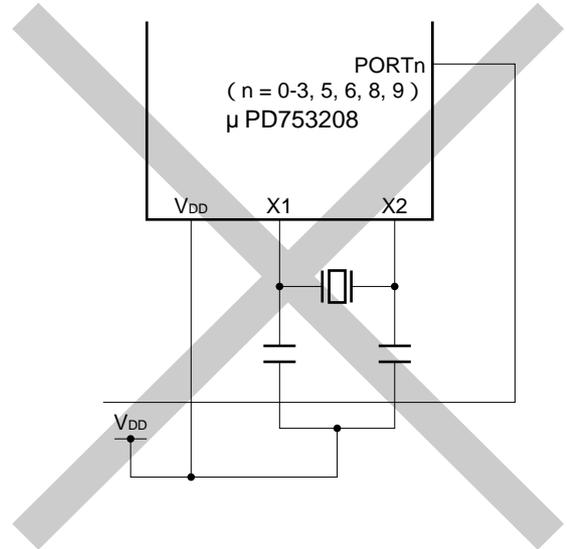
図5 - 14に発振子の接続回路の悪い例を示します。

図5-14 発振子の接続回路の悪い例(1/2)

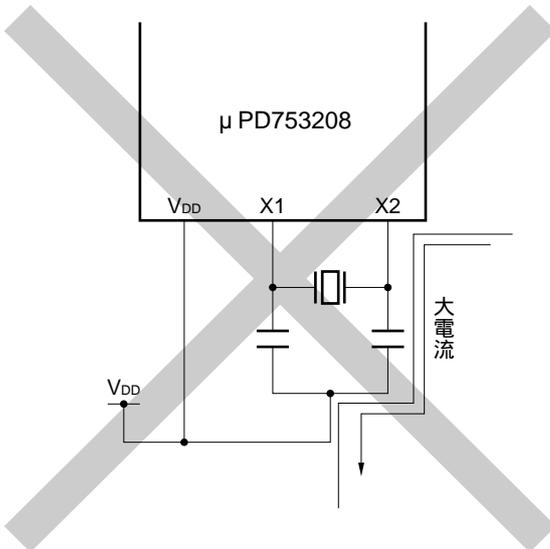
(a) 接続回路の配線が長い



(b) 信号線が交差している



(c) 変化する大電流が信号線に
近接している



(d) 発振回路部の電源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)

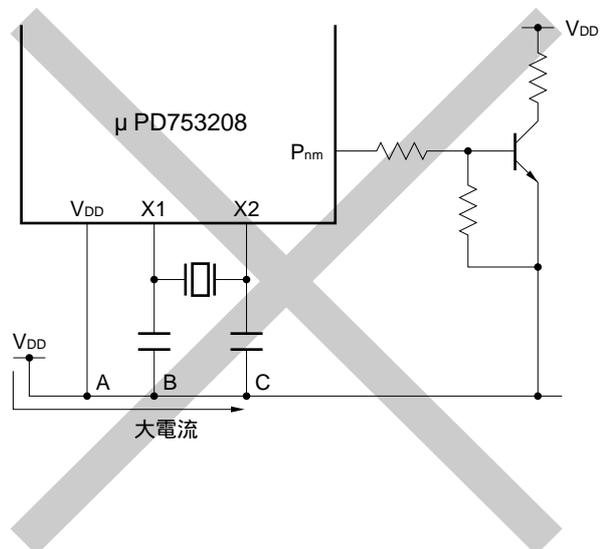
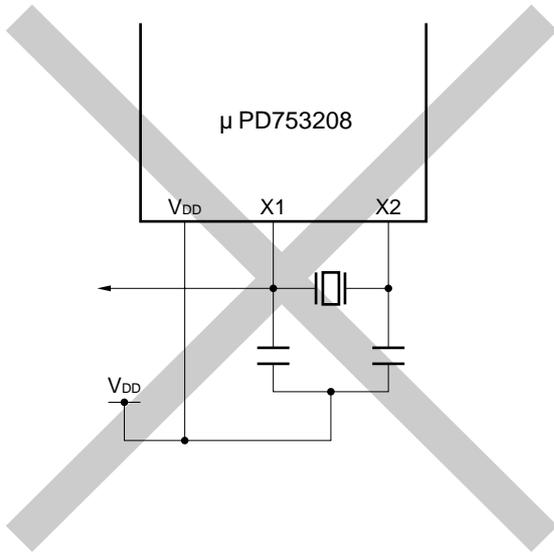


図5 - 14 発振子の接続回路の悪い例 (2/2)

(e) 信号を取り出している



(3) 分周回路

分周回路は、システム・クロック発振回路出力 (fx) を分周して、各種クロックを生成します。

5.2.3 CPUクロックの設定

(1) CPUクロックの切り替えに要する時間

CPUクロックはPCCの下位2ビットにより切り替えることができますが、この切り替えは、レジスタの書き換え後すぐには行われず、一定のマシン・サイクルの間は切り替え前のクロックで動作します。したがって、システム・クロックの発振を停止させる場合には、この切り替え時間経過後、STOP命令を実行する必要があります。

表5 - 5 CPUクロックの切り替えに要する最大時間

切り替え前の設定値		切り替え後の設定値							
PCC1	PCC0	PCC1	PCC0	PCC1	PCC0	PCC1	PCC0	PCC1	PCC0
		0	0	0	1	1	0	1	1
0	0	/		1マシン・サイクル		1マシン・サイクル		1マシン・サイクル	
0	1			4マシン・サイクル		4マシン・サイクル		4マシン・サイクル	
1	0	8マシン・サイクル		8マシン・サイクル		/		8マシン・サイクル	
1	1	16マシン・サイクル		16マシン・サイクル				16マシン・サイクル	

注意 発振子の周囲温度、負荷容量の性能のばらつきなどの条件により、 f_x の値は変化します。特に f_x が公称値より高い場合には、表中のマシン・サイクルは、 f_x の公称値で求めたマシン・サイクルより大きくなります。

したがって、CPUクロックの切り替えに必要なウェイト時間を設定するときは、 f_x の公称値で求めたマシン・サイクルよりも長くしてください。

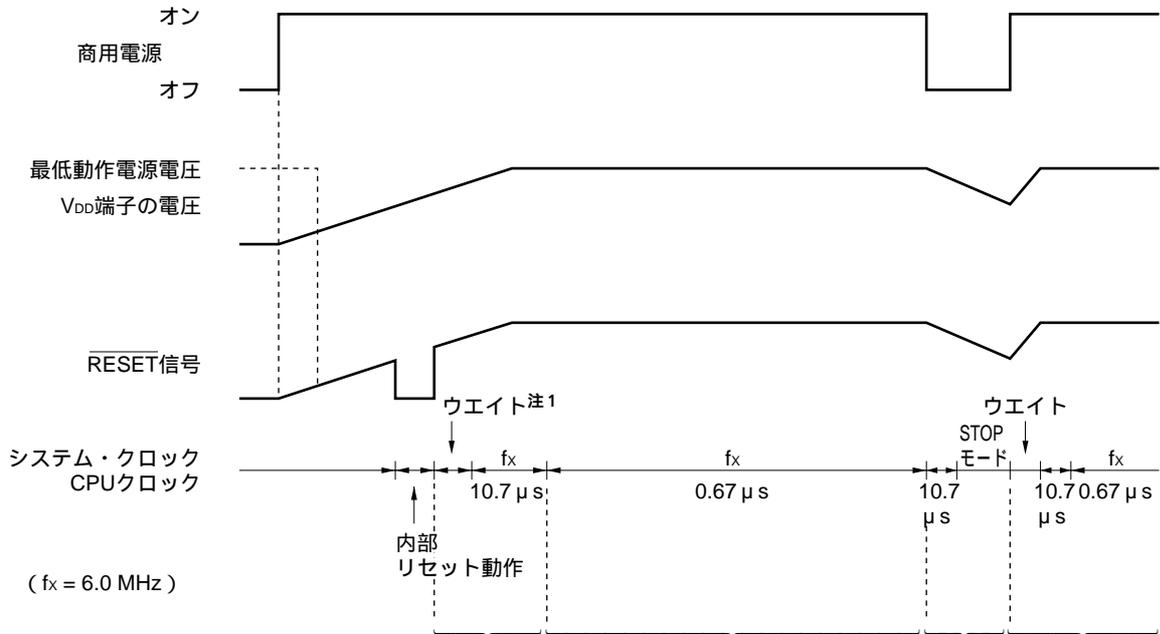
備考 CPUクロック は、内部CPUに供給されるクロックで、この逆数が最小命令時間（このマニュアルでは1マシン・サイクルと定義する）となります。

(2) CPUクロックの切り替え手順

CPUクロックの切り替えを図5 - 15に従って説明します。

★

図5 - 15 CPUクロックの切り替え



RESET信号発生により、発振安定時間確保のためのウエイト時間^{注1}後、CPUはシステム・クロックの最低速（10.7 μs：6.0 MHz動作時，15.3 μs：4.19 MHz動作時）で動作をスタートします。

VDD端子の電圧が、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、PCCを書き換え最高速で動作します。

商用電源がオフしたことを割り込み入力^{注2}により検出し、PCCを書き換え最低速で動作するのに必要な時間経過後、STOPモードにします。

商用電源が復帰したことを割り込み入力^{注2}で検出して、STOPモードを解除します。発振安定時間確保のためのウエイト時間（BTMによる設定時間）後、CPUはシステム・クロックの最低速で動作を再スタートします。その後、再びVDD端子の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が経過したあと、PCCを書き換え最高速で動作します。

注1．マスク・オプションにより、次の2つの時間を選択することができます。

$2^{17}/f_x$ （21.8 ms：6.0 MHz動作時，31.3 ms：4.19 MHz動作時）

$2^{15}/f_x$ （5.46 ms：6.0 MHz動作時，7.81 ms：4.19 MHz動作時）

ただし、μPD75P3216にはマスク・オプションがなく、 $2^{15}/f_x$ に固定されています。

2．INT4を使うと効果的です。

5.2.4 クロック出力回路

(1) クロック出力回路の構成

クロック出力回路は、図5-16のように構成されています。

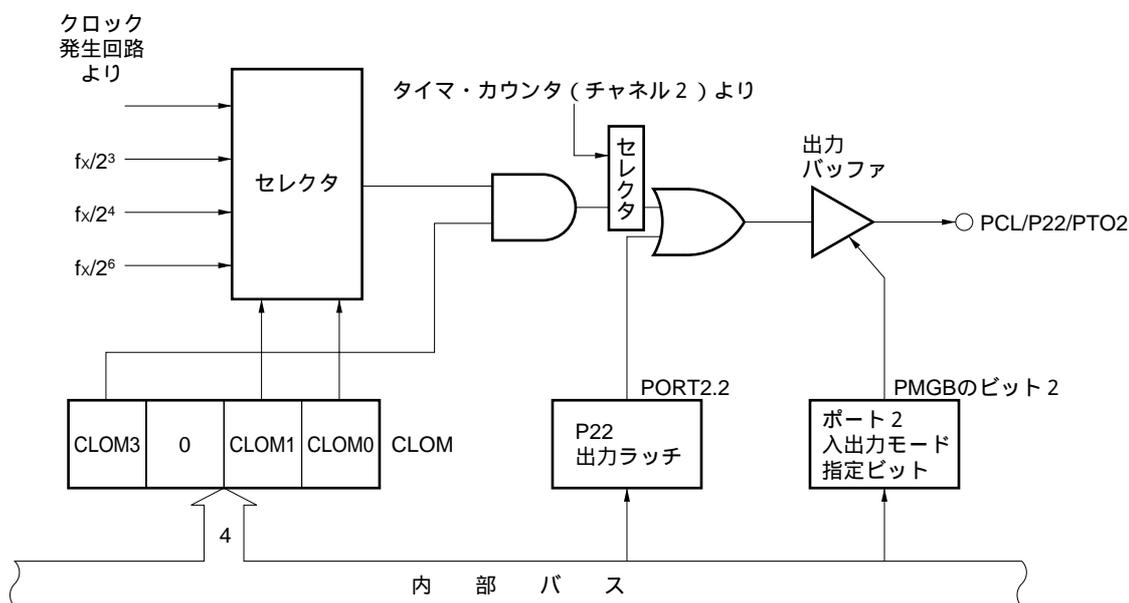
(2) クロック出力回路の機能

クロック出力回路は、P22/PTO2/PCL端子からクロック・パルスを出力するための回路で、リモコン波形出力への応用や周辺LSIにクロック・パルスを供給する場合などに利用します。

クロック・パルスを出力するときは、次の手順で行います。

- (a) クロック出力周波数を選択する。クロックの出力は禁止。
- (b) P22の出力ラッチに0を書き込む。
- (c) ポート2の入出力モードを出力に設定。
- (d) タイマ・カウンタ(チャンネル2)の出力を禁止。
- (e) クロックの出力を許可。

図5-16 クロック出力回路のブロック図



備考 クロックの出力の許可/禁止を切り替えるときに、幅の短いパルスが出力されないように考慮されています。

(3) クロック出力モード・レジスタ (CLOM)

CLOMは、クロック出力を制御する4ビット・レジスタです。

CLOMは、4ビット・メモリ操作命令で設定します。

例 CPUクロック をPCL/PTO2/P22端子より出力

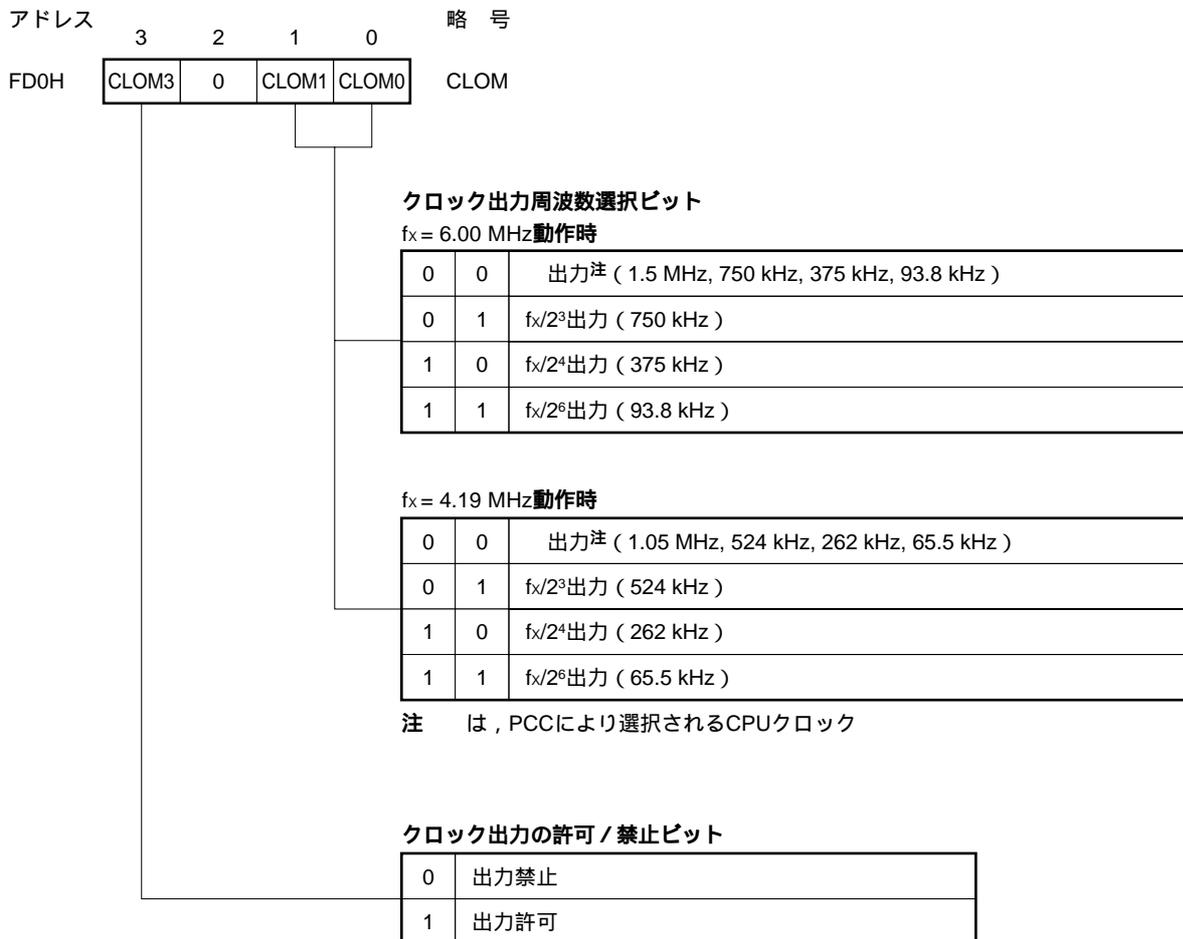
SEL MB15 ;またはCLR1 MBE

MOV A, #1000B

MOV CLOM, A

RESET信号発生によりCLOMは“0”にクリアされ、クロック出力は禁止状態になります。

図5 - 17 クロック出力モード・レジスタのフォーマット



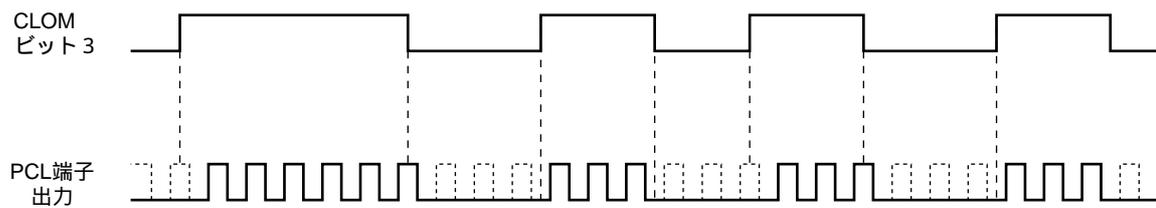
注意 CLOMのビット2には、必ず0を設定してください。

(4) リモコン波形出力への応用例

μPD753208のクロック出力機能は、リモコン波形出力に応用可能です。リモコン波形出力のキャリア周波数は、クロック出力モード・レジスタのクロック周波数選択ビットにより選択します。パルスの出力の許可/禁止はクロック出力の許可/禁止ビットをソフトウェアで制御することにより行います。

クロック出力の許可/禁止切り替え時に、幅の狭いパルスは出力されないよう考慮されています。

図5 - 18 リモコン波形出力応用例



5.3 ベーシック・インターバル・タイマ/ウォッチドッグ・タイマ

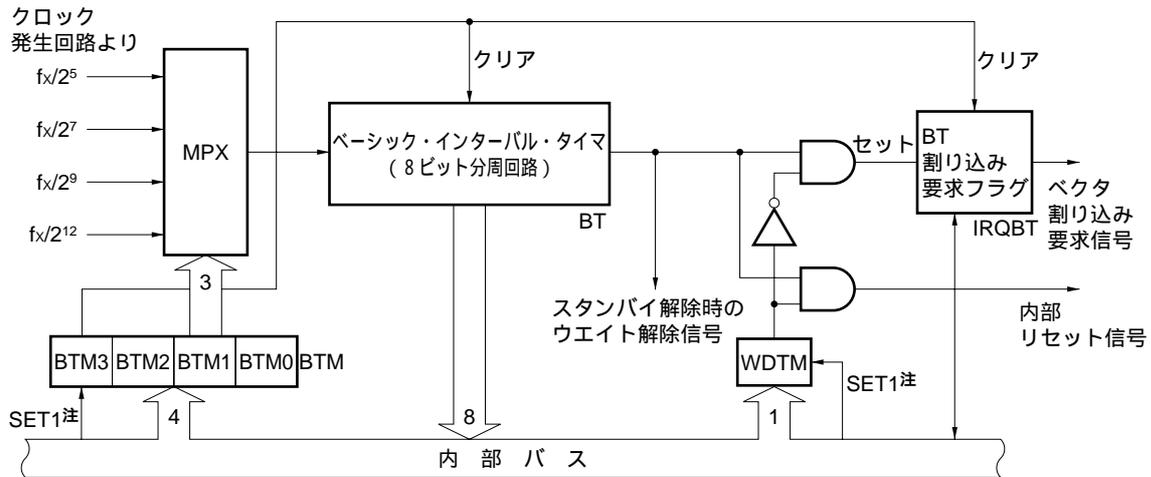
μPD753208は8ビット・ベーシック・インターバル・タイマ/ウォッチドッグ・タイマを備えており、次のような機能があります。

- (a) 基準時間割り込みを発生するインターバル・タイマ動作
- (b) プログラムの暴走を検出し、CPUをリセットするウォッチドッグ・タイマ動作
- (c) スタンバイ・モード解除時のウエイト時間の選択とカウント
- (d) カウント内容の読み出し

5.3.1 ベーシック・インターバル・タイマ/ウォッチドッグ・タイマの構成

ベーシック・インターバル・タイマ/ウォッチドッグ・タイマは、図5-19のように構成されています。

図5-19 ベーシック・インターバル・タイマ/ウォッチドッグ・タイマのブロック図



注 命令の実行

5.3.2 ベーシック・インターバル・タイマ・モード・レジスタ (BTM)

BTMは、ベーシック・インターバル・タイマ (BT) の動作を制御する4ビット・レジスタです。

BTMは、4ビット・メモリ操作命令により設定します。

ビット3は、ビット操作命令により単独で操作できます。

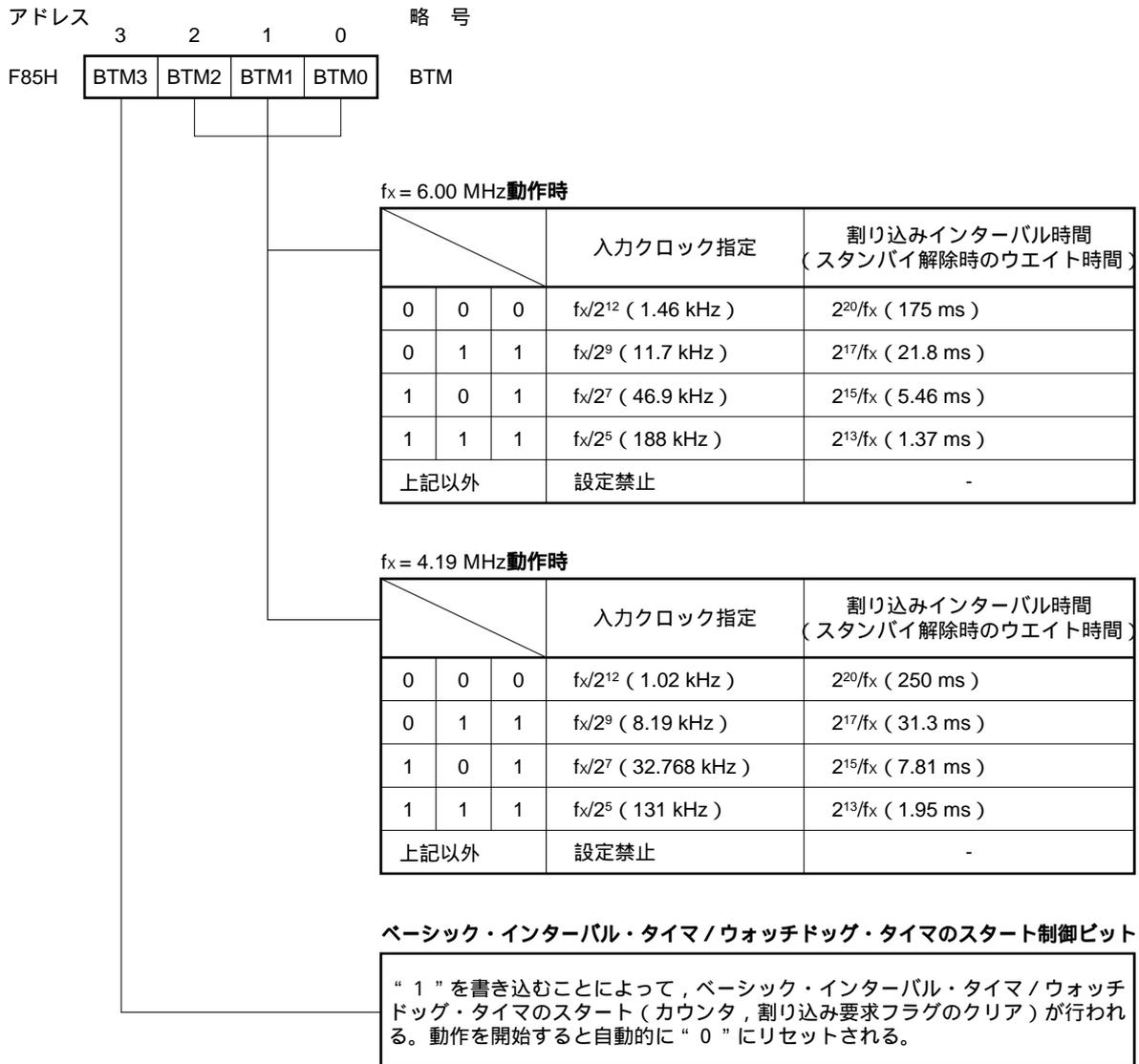
例 割り込み発生インターバルを1.37 ms (6.00 MHz) に設定

```
SEL  MB15          ;またはCLR1 MBE
CLR1  WDTM
MOV   A, #1111B
MOV   BTM, A       ;BTM 1111B
```

ビット3を“1”に設定すると、BTの内容がクリアされると同時に、ベーシック・インターバル・タイマ/ウォッチドッグ・タイマ割り込み要求フラグ (IRQBT) もクリアされます (ベーシック・インターバル・タイマ/ウォッチドッグ・タイマのスタート)。

$\overline{\text{RESET}}$ 信号発生により内容は“0”にクリアされ、割り込み要求信号の発生インターバル時間は最長に設定されます。

図5 - 20 ベーシック・インターバル・タイマ・モード・レジスタのフォーマット



5.3.3 ウォッチドッグ・タイマ許可フラグ (WDTM)

WDTMは、オーバーフローによりリセット信号を発生することを許可するフラグです。

WDTMは、ビット操作命令により設定します。一度セットすると、命令ではクリアできません。

例 ウォッチドッグ・タイマ機能の設定

```
SEL  MB15      ;またはCLR1 MBE
SET1 WDTM
   ⋮
SET1 BTM.3     ;BTMのビット3を“1”にセット
```

$\overline{\text{RESET}}$ 信号発生により内容は0にクリアされます。

図5 - 21 ウォッチドッグ・タイマ許可フラグ (WDTM) のフォーマット



5.3.4 ベーシック・インターバル・タイマの動作

WDTMに“0”をセットすると、ベーシック・インターバル・タイマ(BT)のオーバフローにより割り込み要求フラグ(IRQBT)をセットし、インターバル・タイマとして動作します。BTは、クロック発生回路からのクロックによって常にインクリメントされ、カウント動作を停止することはできません。

割り込み発生インターバルは、BTMの設定により4通りの時間が選択できます(図5-20参照)。

BTMのビット3を“1”にセットすることにより、BTとIRQBTをクリアすることができます(インターバル・タイマとしてのスタート指示)。

BTは、8ビット操作命令でカウント状態を読み出すことができます。なお、データの書き込みはできません。

タイマ動作は、次のように行ってください(、の設定は同時に行ってもかまいません)。

BTMにインターバル時間をセットする。

BTMのビット3に“1”をセットする。

例 1.37 ms (6.00 MHz動作時) ごとに割り込みを発生する。

```
SET1  MBE
SEL   MB15
MOV   A, #1111B
MOV   BTM, A      ; 時間設定とスタート
EI    ; 割り込みを許可
EI    IEBT        ; BT割り込みを許可
```

5.3.5 ウォッチドッグ・タイマの動作

ベーシック・インターバル・タイマ/ウォッチドッグ・タイマは、WDTMに“1”をセットすると、ベーシック・インターバル・タイマ(BT)のオーパフローにより内部リセット信号を発生するウォッチドッグ・タイマとして動作します。ただしSTOP命令解除後の発振ウエイト時間によるタイマのオーパフローでは、リセットはかかりません(なお、WDTMは一度“1”にセットすると、リセット以外にクリアすることはできません)。BTは、クロック発生回路からのクロックによって常にインクリメントされ、カウント動作を停止することはできません。

ウォッチドッグ・タイマ・モードでは、BTのオーパフローするインターバル時間を利用して、プログラムの暴走を検出します。このインターバル時間は、BTMのビット2-0の設定により4通りの時間が選択できます(図5-20参照)。これらの中からユーザ・システムに応じて暴走検出に必要な時間を決めてください。インターバル時間を設定しておいて、プログラムをその時間内に実行できる単位に分割し、それぞれの単位の最後でBTをクリアする命令を実行させるようにします。そうすると、設定時間内にこのBTクリアを実行する命令にたどりつかなければ(順調にプログラムの実行が進んでいなければ=暴走)BTはオーパフローし、内部リセット信号が発生してプログラムを強制終了させてしまいます。この結果、内部リセットがかかったということはプログラムの暴走が起きたことを示し、その検出ができたことになります。

ウォッチドッグ・タイマの設定手順は、次のように行ってください(、 の設定は同時に行ってもかまいません)。

BTMにインターバル時間をセットする。	} 初期設定
BTMのビット3に“1”をセットする。	
WDTMに“1”をセットする。	

～ を設定したあとは、インターバル時間以内にBTMのビット3に“1”をセットする。

例 5.46 msのウォッチドッグ・タイマとして使用する（6.00 MHz動作時）。

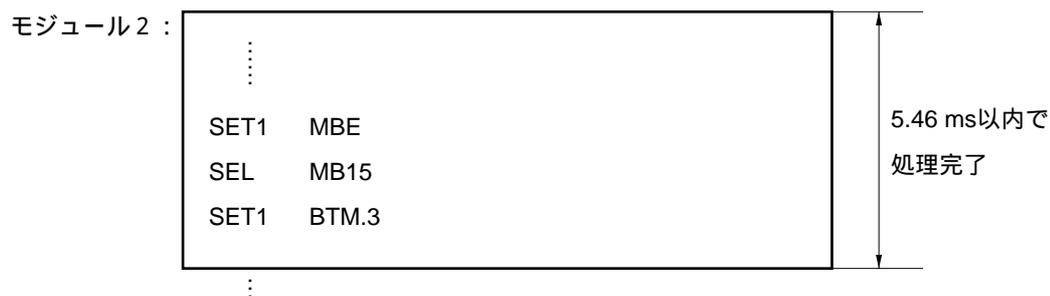
プログラムをBTMの設定時間（5.46 ms）以内に処理が終了するいくつかのモジュールに分割し、各モジュールの終わりでBTをクリアする。暴走した場合、BTが設定時間内にクリアされないためオーバフローしてしまい、内部リセット信号が発生する。

初期設定：

```

SET1  MBE
SEL   MB15
MOV   A, #1101B
MOV   BTM, A      ; 時間設定とスタート
SET1  WDTM       ; ウォッチドッグ・タイマを許可
      ⋮
    
```

（以後、5.46 msごとにBTMのビット3に“1”をセットする。）



5.3.6 その他の機能

ベーシック・インターバル・タイマ/ウォッチドッグ・タイマは、ベーシック・インターバル・タイマ動作、ウォッチドッグ・タイマ動作に関係なく、次の機能があります。

スタンバイ・モード解除後ウェイト時間の選択とカウント
 カウント内容の読み出し

(1) STOPモード解除後のウェイト時間の選択とカウント

STOPモードの解除時には、システム・クロックの発振が安定するまでの発振安定時間を得るため、ベーシック・インターバル・タイマ (BT) のオーバフローまでの間CPUの動作を停止するウェイト機能があります。

$\overline{\text{RESET}}$ 信号発生後のウェイト時間は、マスク・オプションにより固定ですが、割り込み発生によるSTOPモード解除時は、BTMの設定によりウェイト時間を選択できます。その場合のウェイト時間は図5-20のインターバル時間と同じになります。BTMへの設定は、STOPモードに設定する前に行います (詳しくは第7章 スタンバイ機能を参照)。

例 STOPモードを割り込みで解除したときのウェイト時間を5.46 msに設定する (6.00 MHz動作時)。

```
SET1 MBE
SEL MB15
MOV A, #1101B
MOV BTM, A ; 時間設定
STOP ; STOPモード設定
NOP
```

(2) カウント動作の読み出し

ベーシック・インターバル・タイマ (BT) は、8ビット操作命令でカウント状態を読み出すことができます。なお、データの書き込みはできません。

注意 BTのカウント内容を読み出す場合は、カウント更新中の不安定なデータを読み出してしまいうのを防ぐため、読み出し命令は2回実行してください。そして、読み出した2つの内容を比較して、妥当な値であればあの方を読み出し結果とし、まったく異なれば初めからやり直すようにしてください。

例1 . BTのカウンタ内容の読み出し。

```

SET1   MBE
SEL    MB15
MOV    HL, #BT    ; BTのアドレスをHLにセット
LOOP:  MOV    XA, @HL ; 1回目の読み出し
      MOV    BC, XA
      MOV    XA, @HL ; 2回目の読み出し
      SKE   XA, BC
      BR    LOOP

```

2 . INT4割り込み（両エッジ検出）へ入力されるパルスのハイ・レベルの幅を設定する（パルス幅はBTの設定値を越えないものとする。また、BTMの設定値は5.46 ms以上（6.00 MHz動作時）とする）。

INT4割り込みルーチン（MBE = 0）

```

LOOP:  MOV    XA, BT    ; 1回目の読み出し
      MOV    BC, XA    ; データを格納
      MOV    XA, BT    ; 2回目の読み出し
      SKE   A, C
      BR    LOOP
      MOV    A, X
      SKE   A, B
      BR    LOOP
      SKT   PORT0.0    ; P00 = 1 ?
      BR    AA        ; NO
      MOV    XA, BC    ; データをデータ・メモリに格納
      MOV    BUFF, XA
      CLR1  FLAG      ; データあり。フラグをクリア
      RETI
AA:    MOV    HL, #BUFF
      MOV    A, C
      SUBC  A, @HL
      INCS  L
      MOV    C, A
      MOV    A, B
      SUBC  A, @HL
      MOV    B, A
      MOV    XA, BC
      MOV    BUFF, XA ; データを格納
      SET1  FLAG      ; データあり。フラグをセット
      RETI

```

5.4 時計用タイマ

μPD753208は時計用タイマを1チャンネル内蔵しています。時計用タイマには、次のような機能があります。

(a) 0.5秒の時間間隔でテスト・フラグ (IRQW) をセットします。

IRQWによりスタンバイ・モードの解除ができます。

(b) システム・クロックで0.5秒の時間間隔を作ることができます。この場合は、システム・クロック周波数として $f_x = 4.194304$ MHzを使用してください。

(c) 早送りモードにより128倍 (3.91 ms) の時間間隔となり、プログラムのディバグや検査に便利です。

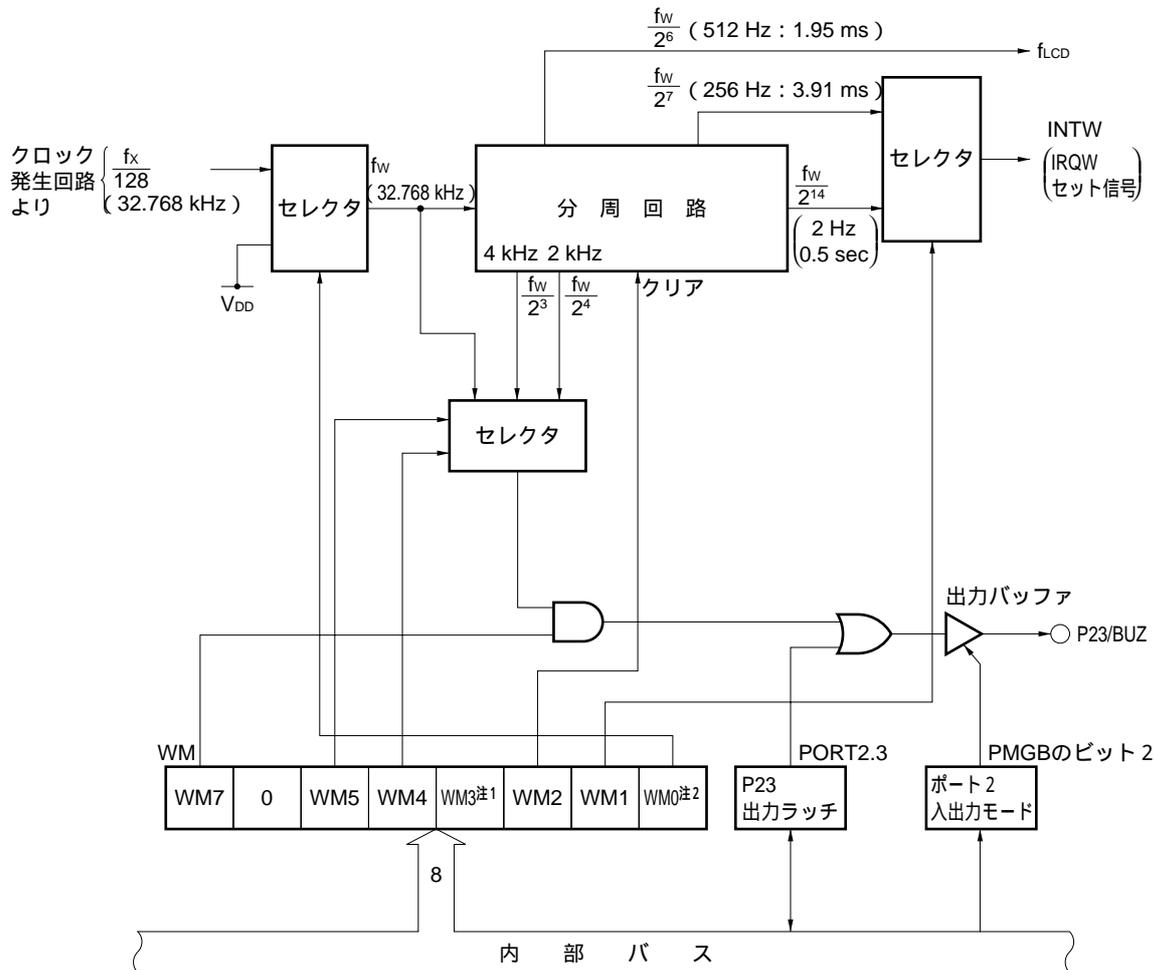
(d) 任意の周波数 (2.048, 4.096, 32.768 kHz) をP23/BUZ端子に出力することができ、ブザー音発生や、システム・クロック発振周波数のトリミングに使用できます。

(e) 分周回路のクリアをすることによって、時計をゼロ秒スタートできます。

5.4.1 時計用タイマの構成

時計用タイマは図5 - 22のように構成されています。

図5-22 時計用タイマのブロック図



注1. WM3は、データの読み出し時には不定となります。

2. WM0には、必ず0を設定してください。

備考 ()内は $f_x = 4.194304$ MHzの場合です。

5.4.2 時計モード・レジスタ

時計モード・レジスタ (WM) は、時計用タイマを制御する8ビット・レジスタです。8ビット・メモリ操作命令で操作します。図5-23にそのフォーマットを示します。

時計モード・レジスタのビット0には、必ず0を設定してください。また、ビット6は0に固定されています。ビット3は、データの読み出し時には不定となります。

RESET信号発生により、ビット3を除く全ビットは“0”にクリアされます。

例 システム・クロック (4.19 MHz) で時間を作る。ブザー出力を許可。

```
CLR1 MBE
MOV XA, #84H
MOV WM, XA ; WMをセット
```

図5 - 23 時計モード・レジスタのフォーマット

アドレス	7	6	5	4	3	2	1	0	略号
F98H	WM7	0	WM5	WM4	WM3 ^{注1}	WM2	WM1	WM0 ^{注2}	WM

BUZ出力 許可/禁止ビット

WM7	0	BUZ出力禁止
	1	BUZ出力許可

BUZ出力周波数選択ビット

WM5	WM4	BUZ出力周波数
0	0	$\frac{f_w}{2^4}$ (2.048 kHz)
0	1	$\frac{f_w}{2^3}$ (4.096 kHz)
1	0	設定禁止
1	1	f_w (32.768 kHz)

時計動作 許可/禁止ビット

WM2	0	時計動作停止 (分周回路クリア)
	1	時計動作可能

動作モード選択ビット

WM1	0	通常時計モード ($\frac{f_w}{2^{14}}$: 0.5 秒でIRQWをセット)
	1	早送り時計モード ($\frac{f_w}{2^7}$: 3.91 msでIRQWをセット)

注1 . WM3は、データの読み出し時には不定となります。

2 . WM0には、必ず0を設定してください。

備考 ()内は、 $f_w = 32.768$ kHz時

5.5 タイマ/イベント・カウンタ

μPD753208は、タイマ/イベント・カウンタを1チャンネル(チャンネル0)、タイマ・カウンタを2チャンネル(チャンネル1、チャンネル2)内蔵しています。その構成を図5-24~5-26に示します。

この項では、タイマ/イベント・カウンタとタイマ・カウンタをあわせてタイマ/イベント・カウンタと呼びます。チャンネル1、チャンネル2の説明としてお読みになる場合は、タイマ/イベント・カウンタをタイマ・カウンタと読み替えてください。

タイマ/イベント・カウンタには、次のような機能があります。

- (a) プログラマブル・インターバル・タイマ動作
- (b) PTO_n端子への任意の周波数の方形波出力
- (c) イベント・カウンタ動作(チャンネル0のみ)
- (d) TI₀端子入力をN分周してPTO₀端子へ出力(分周回路動作)
- (e) シリアル・インタフェース回路へのシリアル・シフト・クロック供給
- (f) カウント状態の呼び出し機能

また、タイマ/イベント・カウンタの動作は、モード・レジスタの設定によって、次に示す4種類のモードでの動作が可能です。

表5-6 使用可能モード一覧表

モード	チャンネル	チャンネル0	チャンネル1	チャンネル2
8ビット・タイマ/イベント・カウンタ・モード ^{注1}				
	ゲート制御機能	× ^{注2}	×	
PWMパルス・ジェネレータ・モード		×	×	
16ビット・タイマ・カウンタ・モード		×		
	ゲート制御機能	× ^{注2}		
キャリア・ジェネレータ・モード		×		

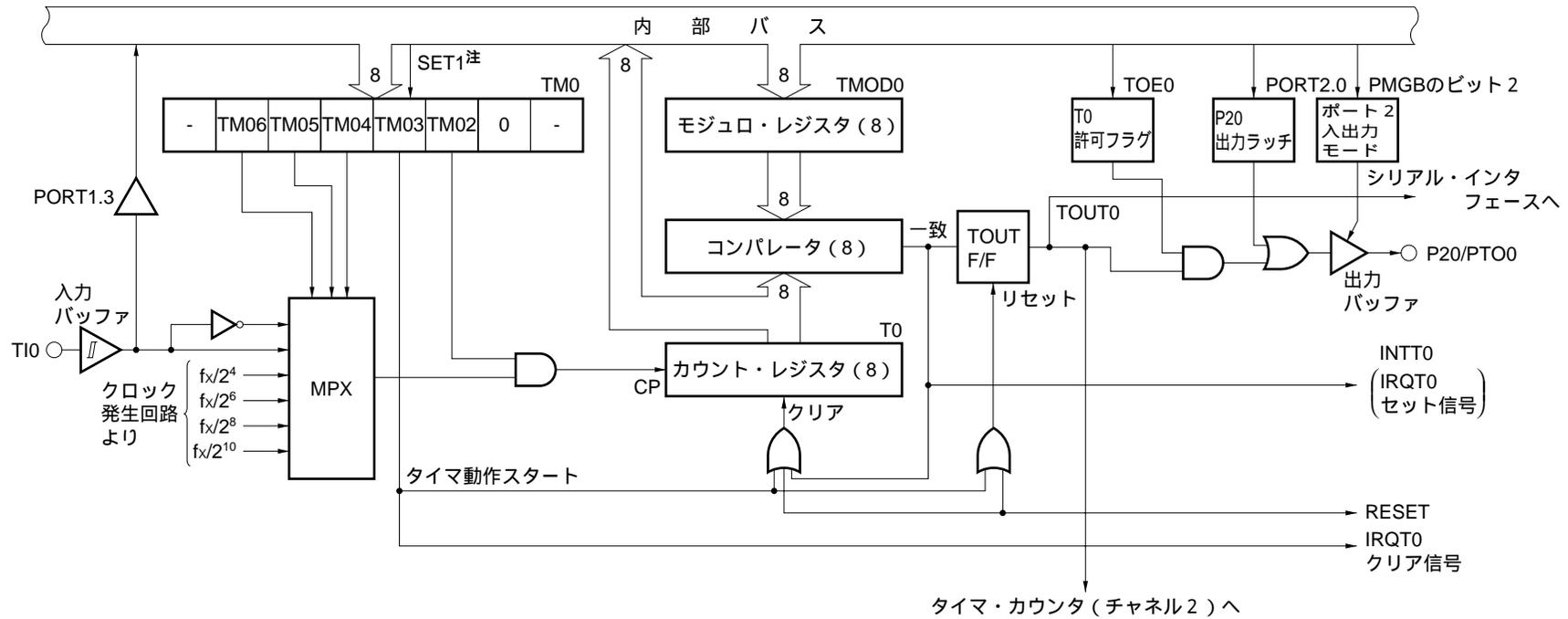
注1．チャンネル0のみです。チャンネル1、チャンネル2では、8ビット・タイマ・カウンタ・モードになります。

2．ゲート制御信号発生用に使用します。

5.5.1 タイマ/イベント・カウンタの構成

タイマ/イベント・カウンタは図5-24~5-26のように構成されています。

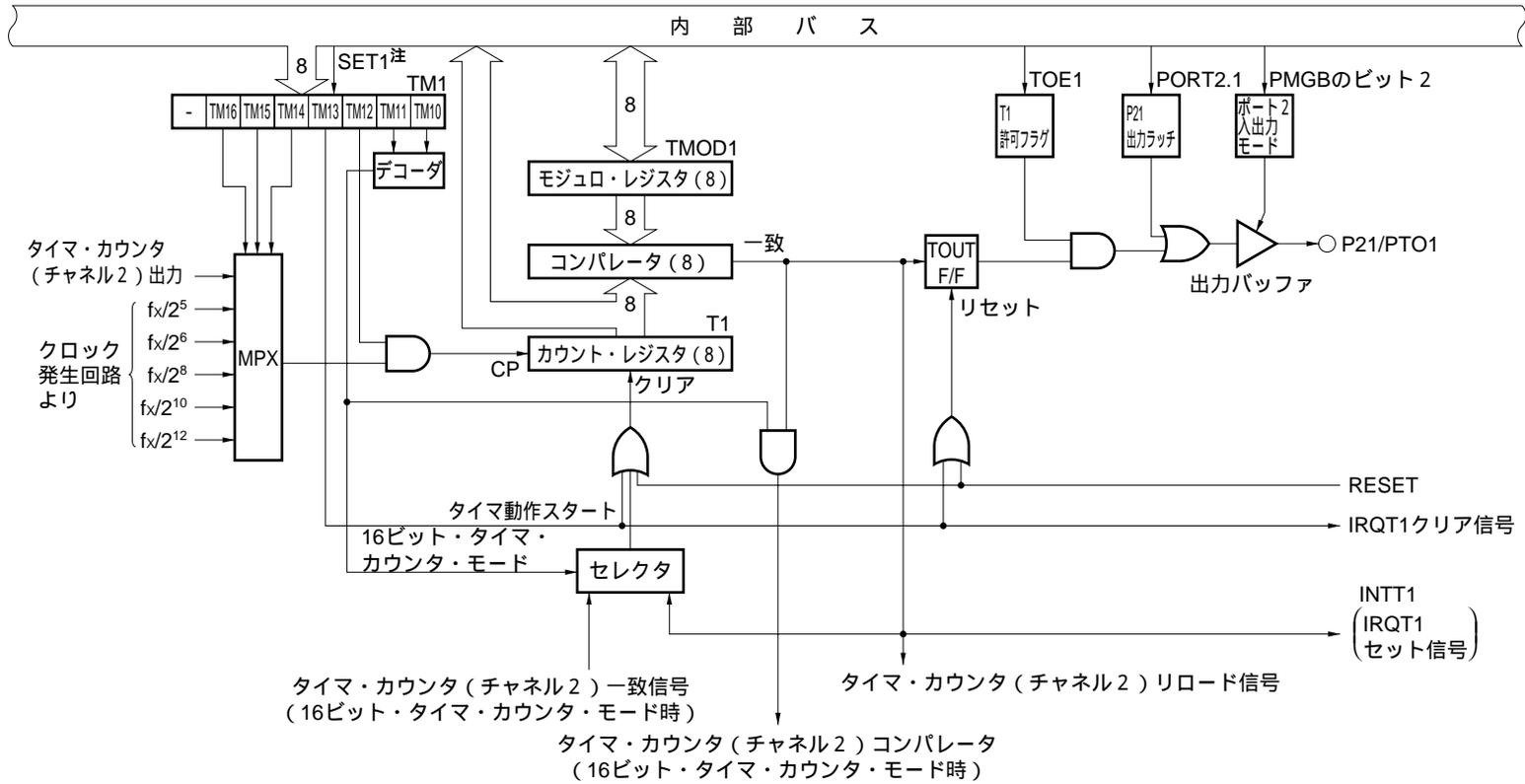
図5-24 タイマ/イベント・カウンタ(チャンネル0)のブロック図



注 命令の実行

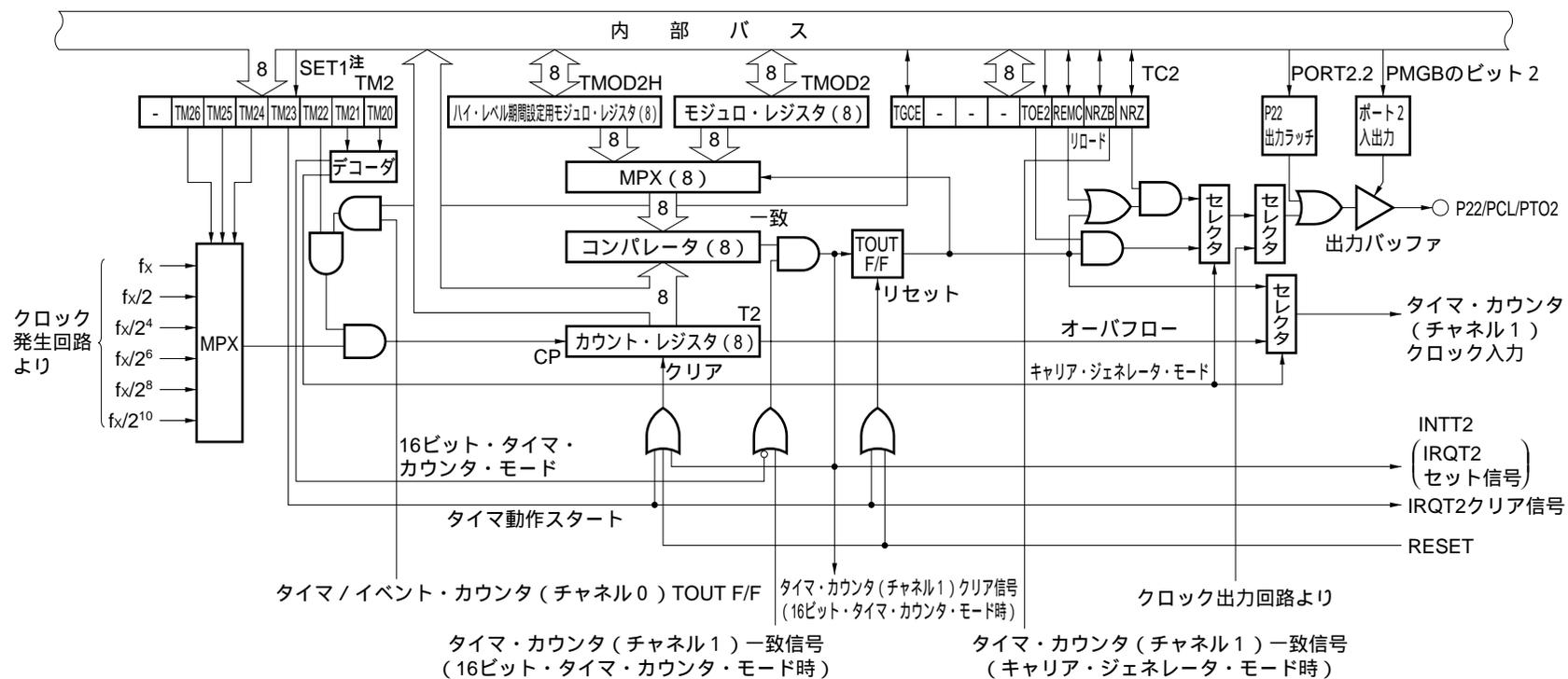
注意 TM0にデータを設定するときは、ビット1を必ず0に設定してください。

図5 - 25 タイマ・カウンタ (チャンネル1) のブロック図



注 命令の実行

図5-26 タイマ・カウンタ(チャンネル2)のブロック図



注 命令の実行

(1) タイマ/イベント・カウンタ・モード・レジスタ (TM0, TM1, TM2)

モード・レジスタ (TMn) は、タイマ/イベント・カウンタを制御する 8 ビット・レジスタです。

図 5 - 27 ~ 5 - 29 にそのフォーマットを示します。

タイマ/イベント・カウンタ・モード・レジスタは 8 ビット・メモリ操作命令により設定します。

ビット 3 は、タイマ・スタート・ビットで、1 ビット単独で操作することができ、タイマ動作を開始すると自動的に “ 0 ” にリセットされます。

$\overline{\text{RESET}}$ 信号発生により、タイマ/イベント・カウンタ・モード・レジスタは全ビット “ 0 ” にクリアされます。

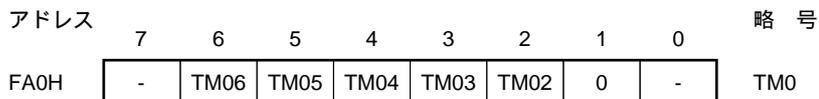
例 1 . CP = 5.86 kHz (6.00 MHz 動作時) のインターバル・タイマ・モードでタイマをスタート。

```
SEL    MB15                ;またはCLR1 MBE
MOV    XA, #01001100B
MOV    TMn, XA             ;TMn 4CH
```

2 . タイマ/イベント・カウンタ・モード・レジスタの設定に従い、タイマを再スタート。

```
SEL    MB15                ;またはCLR1 MBE
SET1   TMn. 3             ;TMn. bit3 1
```

図5 - 27 タイマ/イベント・カウンタ・モード・レジスタ(チャンネル0)のフォーマット



カウント・パルス (CP) 選択ビット

fx = 6.00 MHz時の場合

TM06	TM05	TM04	カウント・パルス (CP)
0	0	0	T10立ち上がりエッジ
0	0	1	T10立ち下がりエッジ
1	0	0	$f_x/2^{10}$ (5.86 kHz)
1	0	1	$f_x/2^8$ (23.4 kHz)
1	1	0	$f_x/2^6$ (93.8 kHz)
1	1	1	$f_x/2^4$ (375 kHz)
上記以外			設定禁止

fx = 4.19 MHz時の場合

TM06	TM05	TM04	カウント・パルス (CP)
0	0	0	T10立ち上がりエッジ
0	0	1	T10立ち下がりエッジ
1	0	0	$f_x/2^{10}$ (4.10 kHz)
1	0	1	$f_x/2^8$ (16.4 kHz)
1	1	0	$f_x/2^6$ (65.5 kHz)
1	1	1	$f_x/2^4$ (262 kHz)
上記以外			設定禁止

タイマ・スタート指示ビット

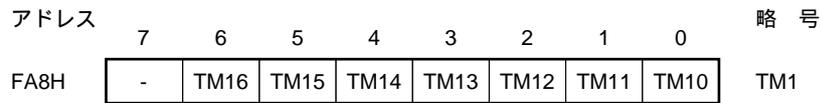
TM03	“ 1 ” を書き込むことによって、カウンタ、IRQT0フラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM02	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

★ **注意** TM0にデータを設定するときは、ビット1を必ず0に設定してください。

図5 - 28 タイマ・カウンタ・モード・レジスタ (チャンネル1) のフォーマット (1/2)



カウント・パルス (CP) 選択ビット

$f_x = 6.00 \text{ MHz}$ 時の場合

TM16	TM15	TM14	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	タイマ・カウンタ・チャンネル2のタイマ出力
0	1	1	$f_x/2^5$ (188 kHz)
1	0	0	$f_x/2^{12}$ (1.46 kHz)
1	0	1	$f_x/2^{10}$ (5.86 kHz)
1	1	0	$f_x/2^8$ (23.4 kHz)
1	1	1	$f_x/2^6$ (93.8 kHz)

$f_x = 4.19 \text{ MHz}$ 時の場合

TM16	TM15	TM14	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	タイマ・カウンタ・チャンネル2のタイマ出力
0	1	1	$f_x/2^5$ (131 kHz)
1	0	0	$f_x/2^{12}$ (1.02 kHz)
1	0	1	$f_x/2^{10}$ (4.10 kHz)
1	1	0	$f_x/2^8$ (16.4 kHz)
1	1	1	$f_x/2^6$ (65.5 kHz)

タイマ・スタート指示ビット

TM13	“1”を書き込むことによって、カウンタ、IRQT1フラグをクリア。ビット2が“1”にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM12	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

図5 - 28 タイマ・カウンタ・モード・レジスタ (チャンネル1) のフォーマット (2/2)

動作モード選択ビット

TM11	TM10	モード
0	0	8ビット・タイマ・カウンタ・モード ^注
1	0	16ビット・タイマ・カウンタ・モード
上記以外		設定禁止

注 タイマ・カウンタ・モード・レジスタ (チャンネル2) のTM20, TM21 (= 11) と組み合わせて使用する場合は, キャリア・ジェネレータ・モードとなります。

図5 - 29 タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマット (1/2)

アドレス	7	6	5	4	3	2	1	0	略号
F90H	-	TM26	TM25	TM24	TM23	TM22	TM21	TM20	TM2

カウント・パルス (CP) 選択ビット

$f_x = 6.00 \text{ MHz}$ 時の場合

TM26	TM25	TM24	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	$f_x/2$ (3.00 MHz)
0	1	1	f_x (6.00 MHz)
1	0	0	$f_x/2^{10}$ (5.86 kHz)
1	0	1	$f_x/2^8$ (23.4 kHz)
1	1	0	$f_x/2^6$ (93.8 kHz)
1	1	1	$f_x/2^4$ (375 kHz)

$f_x = 4.19 \text{ MHz}$ 時の場合

TM26	TM25	TM24	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	$f_x/2$ (2.10 MHz)
0	1	1	f_x (4.19 MHz)
1	0	0	$f_x/2^{10}$ (4.10 kHz)
1	0	1	$f_x/2^8$ (16.4 kHz)
1	1	0	$f_x/2^6$ (65.5 kHz)
1	1	1	$f_x/2^4$ (262 kHz)

図5 - 29 タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマット (2/2)

タイマ・スタート指示ビット

TM23	“ 1 ” を書き込むことによって、カウンタ、IRQT2フラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM22	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM21	TM20	モード
0	0	8ビット・タイマ・カウンタ・モード
0	1	PWMパルス・ジェネレータ・モード
1	0	16ビット・タイマ・カウンタ・モード
1	1	キャリア・ジェネレータ・モード

(2) タイマ/イベント・カウンタ出力許可フラグ (TOE0, TOE1)

タイマ/イベント・カウンタ出力許可フラグ (TOE0, TOE1) は、タイマ・アウトF/F (TOUT F/F) の状態でPTO0, PTO1端子へ出力許可/禁止を制御するフラグです。

タイマ・アウトF/Fは、コンパレータからの一致信号で反転するF/Fです。タイマ/イベント・カウンタ・モード・レジスタ (TM0, TM1) のビット3を“ 1 ” に設定すると、タイマ・アウトF/Fは、“ 0 ” にクリアされます。

RESET信号発生により、TOE0, TOE1とタイマ・アウトF/Fは“ 0 ” にクリアされます。

図5 - 30 タイマ/イベント・カウンタ出力許可フラグのフォーマット



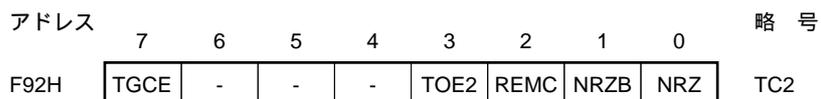
(3) タイマ・カウンタ・コントロール・レジスタ (TC2)

タイマ・カウンタ・コントロール・レジスタ (TC2) は、タイマ・カウンタ (チャンネル2) を制御する 8 ビット・レジスタです。図 5 - 31 にそのフォーマットを示します。

TC2 は、8 ビット、4 ビット・メモリ操作命令、ビット操作命令により設定します。

内部リセット信号発生により、TC2 は全ビット 0 にクリアされます。

図 5 - 31 タイマ・カウンタ・コントロール・レジスタのフォーマット



ゲート制御許可フラグ

TGCE	ゲート制御
0	禁止 (TM2のビット2が“1”にセットされていれば、サンプリング・クロックの状態にかかわらず、カウント動作を行う)
1	許可 (TM2のビット2が“1”にセットされていれば、サンプリング・クロックがハイ・レベルのときカウント動作を行い、ロウ・レベルのときカウント動作を停止する)

タイマ出力許可フラグ

TOE2	タイマ出力
0	禁止 (ロウ・レベルを出力する)
1	許可

リモコン出力制御フラグ

REMC	リモコン出力
0	NRZ = 1 のとき、キャリア・パルスを出力する
1	NRZ = 1 のとき、ハイ・レベルを出力する

ノー・リターン・ゼロ・バッファ・フラグ

NRZB	次に出力するノー・リターン・ゼロ・データ。 タイマ・カウンタ (チャンネル1) の割り込み発生時、NRZへ転送される。
------	--

ノー・リターン・ゼロ・フラグ

NRZ	ノー・リターン・ゼロ・データ
0	ロウ・レベルを出力する
1	キャリア・パルスまたはハイ・レベルを出力する

5.5.2 8ビット・タイマ/イベント・カウンタ・モードの動作

8ビット・タイマ/イベント・カウンタとして使用するモードです。8ビット長でのプログラマブル・インターバル・タイマ動作およびイベント・カウンタ動作（チャンネル0のみ）を行います。

(1) レジスタの設定

8ビット・タイマ/イベント・カウンタ・モードは、次の4つのレジスタを使用します。

- ・タイマ/イベント・カウンタ・モード・レジスタ (TMn)
- ・タイマ・カウンタ・コントロール・レジスタ (TC2)^注
- ・タイマ/イベント・カウンタ・カウント・レジスタ (Tn)
- ・タイマ/イベント・カウンタ・モジュロ・レジスタ (TMODn)

注 タイマ/イベント・カウンタのチャンネル0とチャンネル1は、タイマ/イベント・カウンタ出力許可フラグ (TOE0, TOE1) を使用します。

(a) タイマ/イベント・カウンタ・モード・レジスタ (TMn)

8ビット・タイマ/イベント・カウンタ・モードを使用する場合はTMnを図5 - 32のように設定します (TMnのフォーマットについては図5 - 27~5 - 29を参照してください)。

TMnは、8ビット操作命令により操作します。ビット3は、タイマ・スタート指示ビットで、ビット単位の操作もでき、タイマが動作を開始すると自動的に0クリアされます。

TMnは、内部リセット信号発生時に00Hにクリアされます。

図5-32 タイマ/イベント・カウンタ・モード・レジスタの設定(1/3)

(a) タイマ/イベント・カウンタ(チャンネル0)の場合

アドレス	7	6	5	4	3	2	1	0	略号
FA0H	-	TM06	TM05	TM04	TM03	TM02	0	-	TM0

カウント・パルス(CP)選択ビット

TM06	TM05	TM04	カウント・パルス(CP)
0	0	0	T10立ち上がりエッジ
0	0	1	T10立ち下がりエッジ
1	0	0	$f_x/2^{10}$
1	0	1	$f_x/2^8$
1	1	0	$f_x/2^6$
1	1	1	$f_x/2^4$
上記以外			設定禁止

タイマ・スタート指示ビット

TM03	“1”を書き込むことによって、カウンタ、IRQT0フラグをクリア。ビット2が“1”にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM02	カウント動作
0	停止(カウント内容保持)
1	カウント動作

★ 注意 TM0にデータを設定するときは、ビット1を必ず0に設定してください。

図5 - 32 タイマ/イベント・カウンタ・モード・レジスタの設定 (2/3)

(b) タイマ・カウンタ (チャンネル1) の場合

アドレス	7	6	5	4	3	2	1	0	略号
FA8H	-	TM16	TM15	TM14	TM13	TM12	TM11	TM10	TM1

カウント・パルス (CP) 選択ビット

TM16	TM15	TM14	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	タイマ・カウンタ・チャンネル2のタイマ出力
0	1	1	$f_x/2^5$
1	0	0	$f_x/2^{12}$
1	0	1	$f_x/2^{10}$
1	1	0	$f_x/2^8$
1	1	1	$f_x/2^6$

タイマ・スタート指示ビット

TM13	“ 1 ” を書き込むことによって、カウンタ、IRQT1フラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM12	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM11	TM10	モード
0	0	8ビット・タイマ・カウンタ・モード

図5 - 32 タイマ/イベント・カウンタ・モード・レジスタの設定 (3/3)

(c) タイマ・カウンタ (チャンネル2) の場合

アドレス	7	6	5	4	3	2	1	0	略号
F90H	-	TM26	TM25	TM24	TM23	TM22	TM21	TM20	TM2

カウント・パルス (CP) 選択ビット

TM26	TM25	TM24	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	$f_x/2$
0	1	1	f_x
1	0	0	$f_x/2^{10}$
1	0	1	$f_x/2^8$
1	1	0	$f_x/2^6$
1	1	1	$f_x/2^4$

タイマ・スタート指示ビット

TM23	“ 1 ” を書き込むことによって、カウンタ、IRQT2フラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM22	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM21	TM20	モード
0	0	8ビット・タイマ・カウンタ・モード

(b) タイマ・カウンタ・コントロール・レジスタ (TC2)

タイマ・カウンタ (チャンネル2) を 8 ビット・タイマ・カウンタ・モードで使用する場合の設定を図 5 - 33 に示します (TC2 のフォーマットについては、図 5 - 31 **タイマ・カウンタ・コントロール・レジスタのフォーマット** を参照してください)。

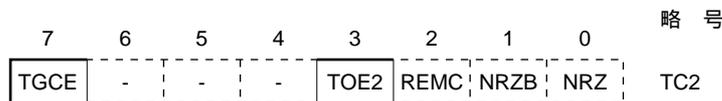
TC2 は、8 ビット、4 ビット操作命令、ビット操作命令により操作します。

TC2 は、内部リセット信号発生により 00H にクリアされます。

実線で示されるフラグは、8 ビット・タイマ・カウンタ・モードにおける使用フラグを表します。

点線で示されるフラグは、8 ビット・タイマ・カウンタ・モードでは使用しないでください (0 を設定してください)。

図 5 - 33 タイマ・カウンタ・コントロール・レジスタの設定



ゲート制御許可フラグ

TGCE	ゲート制御
0	禁止 (TM2 のビット 2 が “ 1 ” にセットされていれば、サンプリング・クロックの状態にかかわらず、カウント動作を行う)
1	許可 (TM2 のビット 2 が “ 1 ” にセットされていれば、サンプリング・クロックがハイ・レベルのときカウント動作を行い、ロウ・レベルのときカウント動作を停止する)

タイマ出力許可フラグ

TOE2	タイマ出力
0	禁止 (ロウ・レベルを出力する)
1	許可

図 5 - 34 タイマ/イベント・カウンタ出力許可フラグの設定

アドレス



タイマ/イベント・カウンタ出力許可フラグ (W)

0	禁止 (ロウ・レベルを出力する)
1	許可

(2) タイマ/イベント・カウンタの時間設定

[タイマ設定時間] (周期) は, [モジユロ・レジスタの内容 + 1] をモード・レジスタの設定により選択された [カウント・パルス (CP) 周波数] で割った値となります。

$$T(\text{sec}) = \frac{n+1}{f_{CP}} = (n+1) \cdot (\text{分解能})$$

- T (sec) : タイマ設定時間 (秒)
 f_{CP} (Hz) : CP周波数 (Hz)
 n : モジユロ・レジスタの内容 (n = 0)

タイマは一度設定すれば, その間隔で毎回割り込み要求フラグ (IRQ_{Tn}) をセットします。

表5 - 7にタイマ/イベント・カウンタの各カウント・パルスでの分解能と最長設定時間 (モジユロ・レジスタにFFHをセットしたときの時間) を示します。

★

表5 - 7 分解能と最長設定時間 (8ビット・タイマ時) (1/2)

(a) タイマ/イベント・カウンタ (チャンネル0) の場合

モード・レジスタ			6.0 MHz動作時		4.19 MHz動作時	
TM06	TM05	TM04	分解能	最長設定時間	分解能	最長設定時間
1	0	0	171 μs	43.7 ms	244 μs	62.5 ms
1	0	1	42.7 μs	10.9 ms	61.0 μs	15.6 ms
1	1	0	10.7 μs	2.73 ms	15.3 μs	3.91 ms
1	1	1	2.67 μs	683 μs	3.81 μs	977 μs

(b) タイマ・カウンタ (チャンネル1) の場合

モード・レジスタ			6.0 MHz動作時		4.19 MHz動作時	
TM16	TM15	TM14	分解能	最長設定時間	分解能	最長設定時間
0	1	1	5.33 μs	1.37 ms	7.64 μs	1.95 ms
1	0	0	683 μs	175 ms	977 μs	250 ms
1	0	1	171 μs	43.7 ms	244 μs	62.5 ms
1	1	0	42.7 μs	10.9 ms	61.0 μs	15.6 ms
1	1	1	10.7 μs	2.73 ms	15.3 μs	3.91 ms

★

表5 - 7 分解能と最長設定時間 (8ビット・タイマ時) (2/2)

(c) タイマ・カウンタ (チャンネル2) の場合

モード・レジスタ			6.0 MHz動作時		4.19 MHz動作時	
TM26	TM25	TM24	分解能	最長設定時間	分解能	最長設定時間
0	1	0	333 ns	85.3 μ s	477 ns	122 μ s
0	1	1	167 ns	42.7 μ s	239 ns	61.1 μ s
1	0	0	171 μ s	43.7 ms	244 μ s	62.5 ms
1	0	1	42.7 μ s	10.9 ms	61.0 μ s	15.6 ms
1	1	0	10.7 μ s	2.73 ms	15.3 μ s	3.91 ms
1	1	1	2.67 μ s	683 μ s	3.81 μ s	977 μ s

(3) タイマ/イベント・カウンタ動作

タイマ/イベント・カウンタ動作は、次のようになります。タイマ・カウンタ (チャンネル2) でこの動作を行うときは、タイマ・カウンタ・コントロール・レジスタ (TC2) のゲート制御許可フラグ (TGCE) を、0 に設定してください。

図5 - 35にタイマ/イベント・カウンタ動作時の構成を示します。

モード・レジスタ (TMn) の設定によりカウント・パルス (CP) が選択され、カウント・レジスタ (Tn) に入力されます。

Tnとモジュロ・レジスタ (TMODn) の内容を比較し、一致すると一致信号を発生し、割り込み要求フラグ (IRQTn) をセットします。同時にタイマ・アウト・フリップフロップ (TOUT F/F) は反転します。

タイマ/イベント・カウンタ動作のタイミングは、図5 - 36のようになります。

タイマ/イベント・カウンタ動作は、通常、次の手順により開始します。

TMODnにカウント数を設定します。

TMnに動作モード、カウント・パルス、スタート指示を設定します。

注意 モジュロ・レジスタ (TMODn) には00H以外の値を設定してください。

タイマ/イベント・カウンタ出力端子 (PTOn) を使用する場合は、兼用端子のP2nを次のように設定してください。

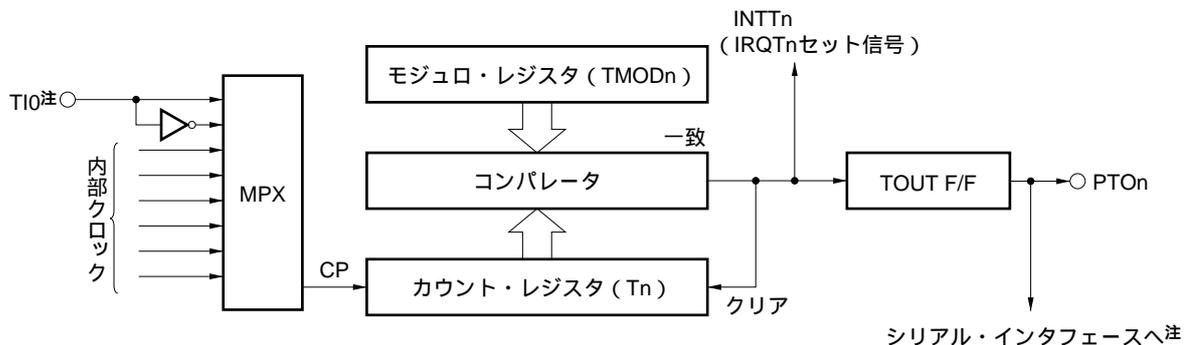
P2nの出力ラッチをクリアする。

ポート2を出力モードに設定する。

ポート2の内蔵プルアップ抵抗を接続しない状態にする (PTO2を出力する場合は、PCLを出力禁止にする)。

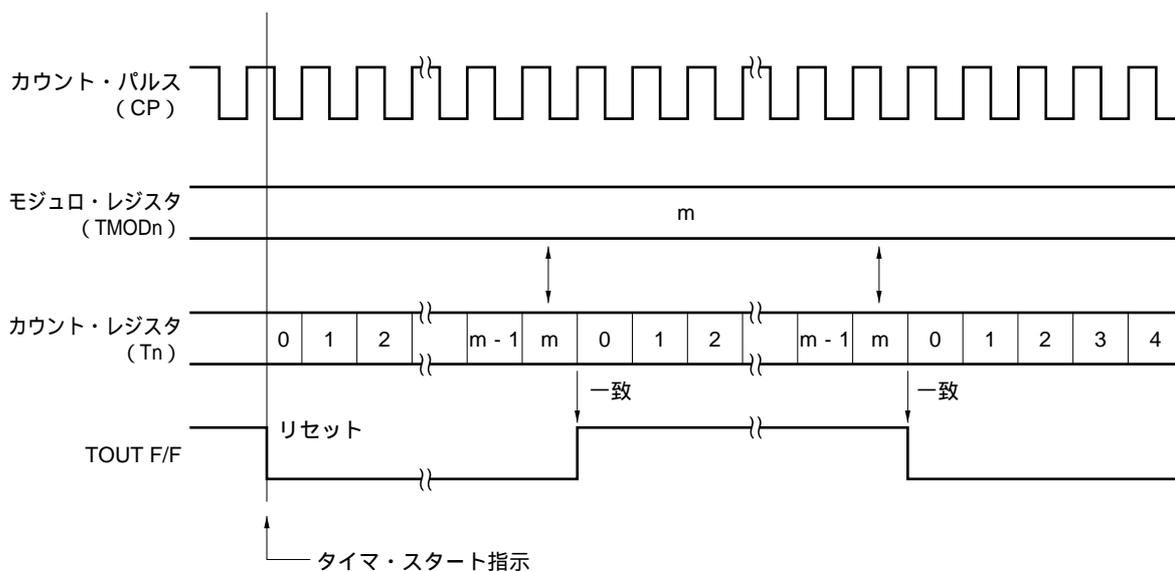
タイマ/イベント・カウンタ出力許可フラグ (TOEn) を1にする。

図5 - 35 タイマ/イベント・カウンタ動作時の構成



注 タイマ/イベント・カウンタのチャンネル0のみです。

図5 - 36 カウント動作時のタイミング



備考 m : モジュール・レジスタの設定値

$n = 0-2$

(4) ゲート制御機能を持つカウンタ動作 (8ビット時)

タイマ・カウンタ (チャンネル2) は、ゲート制御機能を持つカウンタとして使用可能です。この機能を使用するときは、タイマ・カウンタ・コントロール・レジスタのゲート制御許可フラグ (TGCE) を1に設定してください。

タイマ/イベント・カウンタ (チャンネル0) をカウントすることにより、ゲート信号を生成します。

図5-38のようにゲート信号 (T0のTOUT F/Fの出力) がハイ・レベルのとき、タイマ・カウンタ (チャンネル2) のカウント・パルスのカウントをすることができます (詳細は、(3) タイマ/イベント・カウンタ動作を参照してください)。

モード・レジスタ (TM2) の設定によりカウント・パルス (CP) が選択され、ゲート信号がハイ・レベルのときCPがカウント・レジスタ (T2) に入力されます。

ゲート信号の立ち上がりエッジと立ち下がりエッジで割り込みが発生します。通常は、立ち下がりエッジの割り込みサブルーチンでT2の内容を読み出し、T2をクリアして次のカウント動作に備えます。

カウンタ動作のタイミングは、図5-38のようになります。

カウンタ動作は、通常、次の手順により開始します。

TM2に動作モード、カウント・パルス、カウンタ・クリア指示を設定します。

TMOD0にカウント数を設定します。

TM0に動作モード、スタート指示を設定します。

注意 モジユロ・レジスタ (TMOD0, TMOD2) には00H以外の値を設定してください。

図5-37 カウント動作時の構成

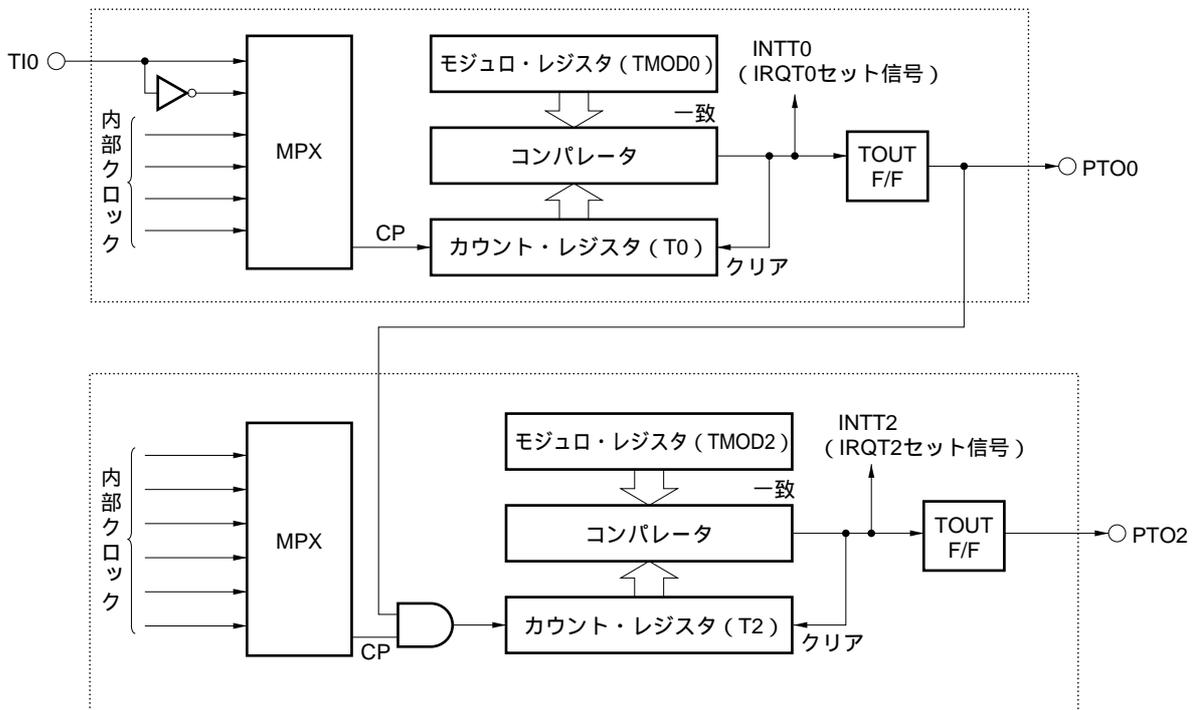
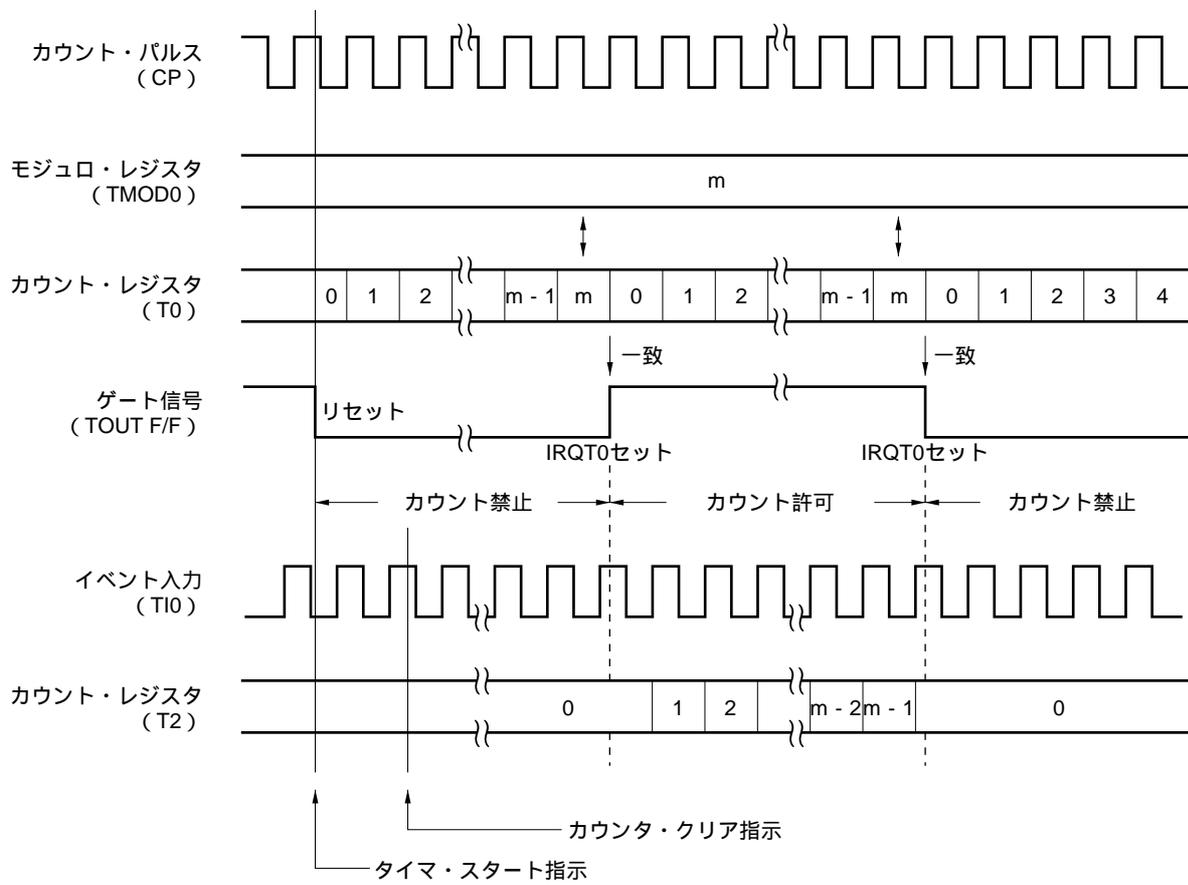


図5 - 38 カウント動作時のタイミング



備考 m : モジュロ・レジスタの設定値

(5) 8ビット・タイマ/イベント・カウンタ・モードの応用

(a) 50 ms間隔で割り込みを発生するインターバル・タイマとして応用します。

- ・モード・レジスタ (TM2) の上位4ビットは0100Bとし、最長設定時間62.5 ms ($f_x = 4.19 \text{ MHz}$ 時) を選択します。
- ・TM2の下位4ビットは1100Bとします。
- ・モジュロ・レジスタ (TMOD2) の設定値は次のとおりです。

$$\frac{50 \text{ ms}}{244 \mu\text{s}} = 205, 205 - 1 = \text{CCH}$$

プログラム例

```
SEL    MB15                ;またはCLR1 MBE
MOV    XA, #0CCH
MOV    TMOD2, XA          ;モジュロを設定
MOV    XA, #01001100B
MOV    TM2, XA            ;モードを設定, タイマ・スタート
EI
EI    IET2                ;タイマの割り込みを許可
```

備考 この応用の場合, TI0端子は入力端子として使用できます。

(b) TI0端子から入力されるパルス数が100になったら割り込みを発生します (チャンネル0のみ, パルスはハイ・アクティブ)。

- ・モード・レジスタ (TM0) の上位4ビットは0000で立ち上がりエッジを選択。
- ・TM0の下位4ビットは1100B。
- ・モジュロ・レジスタ (TMOD0) は99 = 100 - 1。

プログラム例

```
SEL    MB15                ;またはCLR1 MBE
MOV    XA, #100 - 1
MOV    TMOD0, XA          ;モジュロを設定
MOV    XA, #00001100B
MOV    TM0, XA            ;モードを設定, カウント・スタート
EI
EI    IET0                ;INTT0を許可
```

5.5.3 PWMパルス・ジェネレータ・モード (PWMモード) の動作

PWMパルス・ジェネレータとして使用するモードです。8ビット長でのPWMパルス・ジェネレータ動作を行います。

(1) レジスタの設定

PWMモードは、次の5つのレジスタを使用します。

- ・タイマ・カウンタ・モード・レジスタ (TM2)
- ・タイマ・カウンタ・コントロール・レジスタ (TC2)
- ・タイマ・カウンタ・カウント・レジスタ (T2)
- ・タイマ・カウンタ・ハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H)
- ・タイマ・カウンタ・モジュロ・レジスタ (TMOD2)

(a) タイマ・カウンタ・モード・レジスタ (TM2)

PWMモードを使用する場合のTM2の設定を図5 - 39に示します (TM2のフォーマットについては、**図5 - 29 タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマット**を参照してください)。

TM2は、8ビット操作命令により操作します。ビット3は、タイマ・スタート指示ビットで、ビット単位の操作もでき、タイマが動作を開始すると自動的に0にクリアされます。

TM2は、内部リセット信号発生時に00Hにクリアされます。

図5 - 39 タイマ・カウンタ・モード・レジスタの設定

アドレス	7	6	5	4	3	2	1	0	略号
F90H	-	TM26	TM25	TM24	TM23	TM22	TM21	TM20	TM2

カウント・パルス (CP) 選択ビット

TM26	TM25	TM24	カウント・パルス (CP)
0	0	0	設定禁止
0	0	1	設定禁止
0	1	0	$f_x/2$
0	1	1	f_x
1	0	0	$f_x/2^{10}$
1	0	1	$f_x/2^8$
1	1	0	$f_x/2^6$
1	1	1	$f_x/2^4$

タイマ・スタート指示ビット

TM23	“ 1 ” を書き込むことによって、カウンタ、IRQT2フラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM22	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM21	TM20	モード
0	1	PWMパルス・ジェネレータ・モード

(b) タイマ・カウンタ・コントロール・レジスタ (TC2)

PWMモードを使用する場合のTC2の設定を図5 - 40に示します (TC2のフォーマットについては、図5 - 31 タイマ・カウンタ・コントロール・レジスタのフォーマットを参照してください)。

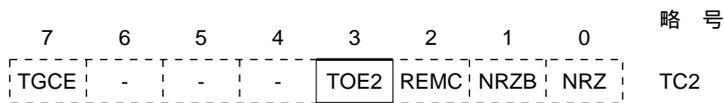
TC2は、8ビット、4ビット操作命令、ビット操作命令により操作します。

TC2は、内部リセット信号発生により00Hにクリアされます。

実線で示されるフラグは、PWMモードにおける使用フラグを表します。

点線で示されるフラグは、PWMモードでは使用しないでください (0を設定してください)。

図5 - 40 タイマ・カウンタ・コントロール・レジスタの設定



タイマ出力許可フラグ

TOE2	タイマ出力
0	禁止 (ロウ・レベルを出力する)
1	許可

(2) PWMパルス・ジェネレータ動作

PWMパルス・ジェネレータ動作は、次のようになります。図5 - 41にその構成を示します。

モード・レジスタ (TM2) の設定によりカウント・パルス (CP) が選択され、カウント・レジスタ (T2) に入力されます。

T2とハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H) の内容を比較し、一致すると一致信号を発生し、タイマ・アウト・フリップフロップ (TOUT F/F) は反転します。

T2とモジュロ・レジスタ (TMOD2) の内容を比較し、一致すると一致信号を発生し、割り込み要求フラグ (IRQT2) をセットします。同時にTOUT F/Fは反転します。

と の動作を交互に繰り返します。

PWMパルス・ジェネレータ動作のタイミングは、図5 - 42のようになります。

PWMパルス・ジェネレータ動作は、通常、次の手順により開始します。

TMOD2Hにカウント数を設定します。

TMOD2にはロウ・レベルのカウント数を設定します。

TM2に動作モード、カウント・パルス、スタート指示を設定します。

注意 モジュロ・レジスタ (TMOD2) とハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H) には00H以外の値を設定してください。

タイマ・カウンタ出力端子 (PTO2) を使用する場合は、兼用端子のP22とPCLを次のように設定してください。

P22の出力ラッチをクリアする。

ポート2を出力モードに設定する。

ポート2の内蔵プルアップ抵抗を接続しない状態に、また、PCLを出力禁止にする。

タイマ・カウンタ出力許可フラグ (TOE2) を1にする。

図5 - 41 PWMパルス・ジェネレータ動作時の構成

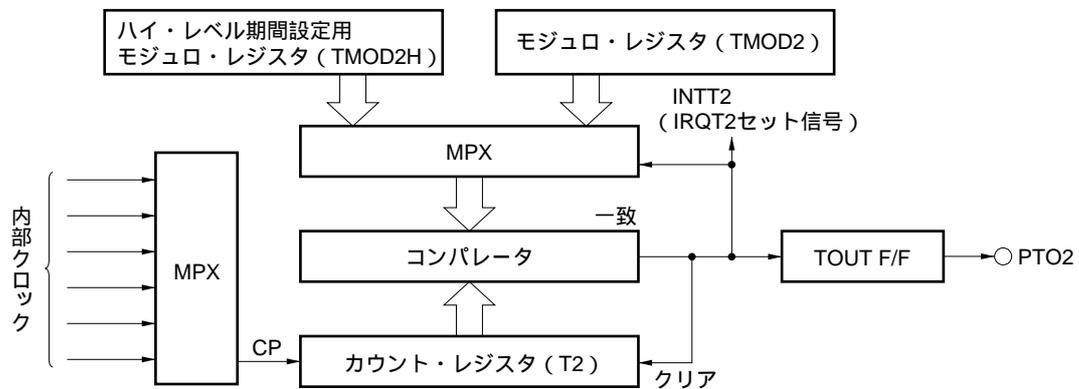
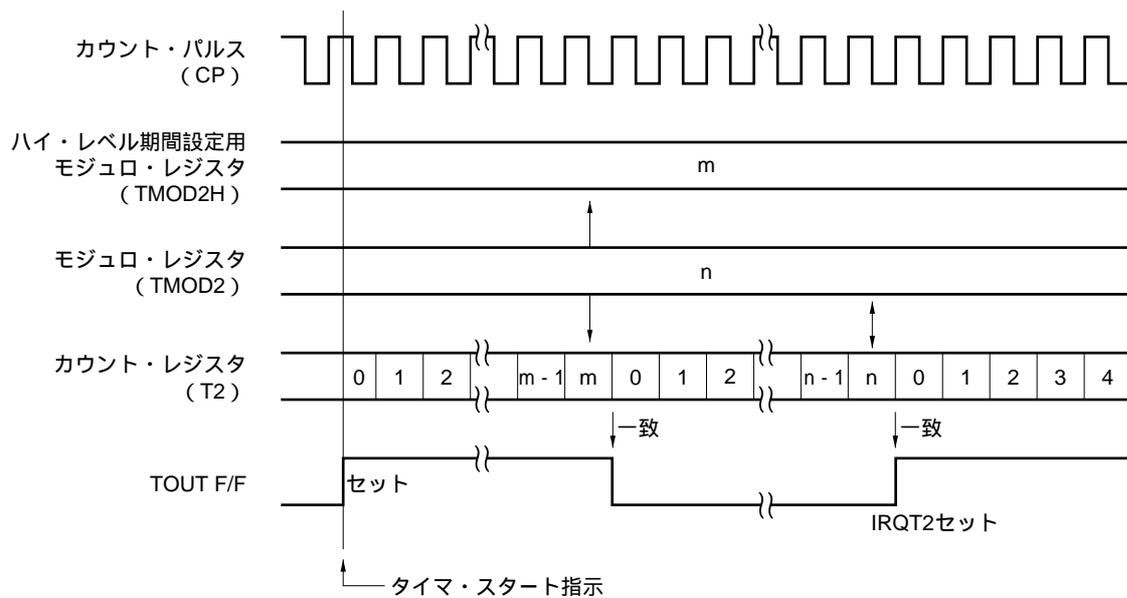


図5 - 42 PWMパルス・ジェネレータ動作時のタイミング



備考 m : ハイ・レベル期間設定用モジュール・レジスタの設定値

n : モジュール・レジスタの設定値

(3) PWMモードの応用

周波数が38.0 kHz (周期が26.3 μ s) , デューティ比が1/3のパルスをPTO2端子に出力します。

- ・モード・レジスタ (TM2) の上位4ビットは0011Bとし, 最長設定時間61.1 μ sを選択します。
- ・TM2の低位4ビットは1101Bとし, PWMモード, カウント動作を選択, タイマ・スタートを指示します。
- ・タイマ出力許可フラグ (TOE2) は “ 1 ” とし, タイマ出力を許可します。
- ・ハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H) の設定値は, 次のとおりです。

$$\frac{1}{3} \cdot \frac{26.3 \mu\text{s}}{239 \text{ ns}} - 1 = 36.7 - 1 \quad 36 = 24\text{H}$$

- ・モジュロ・レジスタ (TMOD2) の設定値は, 次のとおりです。

$$\frac{2}{3} \cdot \frac{26.3 \mu\text{s}}{239 \text{ ns}} - 1 = 73.4 - 1 \quad 72 = 48\text{H}$$

プログラム例

```
SEL    MB15                ;またはCLR1 MBE
SET1   TOE2                ;タイマの出力を許可
MOV    XA, #024H
MOV    TMOD2H, XA         ;モジュロを設定 (ハイ・レベル期間)
MOV    XA, #48H
MOV    TMOD2, XA         ;モジュロを設定 (ロウ・レベル期間)
MOV    XA, #00111101B
MOV    TM2, XA           ;モードを設定, タイマ・スタート
```

備考 この応用の場合, TI0は入力端子として使用できます。

5.5.4 16ビット・タイマ・カウンタ・モードの動作

16ビット・タイマ・カウンタとして使用するモードです。16ビット長でのプログラマブル・インターバル・タイマ動作およびカウント動作を行います。

16ビット・タイマ・カウンタ・モードを使用する場合は、タイマ・カウンタのチャンネル1とチャンネル2を組み合わせて使用します。

(1) レジスタの設定

16ビット・タイマ・カウンタ・モードは、次の7つのレジスタを使用します。

- ・タイマ・カウンタ・モード・レジスタ (TM1, TM2)
- ・タイマ・カウンタ・コントロール・レジスタ (TC2) 注
- ・タイマ・カウンタ・カウント・レジスタ (T1, T2)
- ・タイマ・カウンタ・モジュロ・レジスタ (TMOD1, TMOD2)

注 タイマ・カウンタのチャンネル1は、タイマ・カウンタ出力許可フラグ (TOE1) を使用しません。

(a) タイマ・カウンタ・モード・レジスタ (TM1, TM2)

16ビット・タイマ・カウンタ・モードを使用する場合のTM1, TM2の設定を図5 - 43に示します (TM1, TM2のフォーマットについては図5 - 28 タイマ・カウンタ・モード・レジスタ (チャンネル1) のフォーマット, 図5 - 29 タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマットを参照してください)。

TM1, TM2は、8ビット操作命令により操作します。ビット3は、タイマ・スタート指示ビットで、ビット単位の操作もでき、タイマが動作を開始すると自動的に0クリアされます。

TM1, TM2は、内部リセット信号発生時に00Hにクリアされます。

実線で示されるフラグは、16ビット・タイマ・カウンタ・モードにおける使用ビットを表します。

点線で示されるフラグは、16ビット・タイマ・カウンタ・モードでは使用しないでください (0を設定してください)。

図5 - 43 タイマ・カウンタ・モード・レジスタの設定

アドレス	7	6	5	4	3	2	1	0	略号
FA8H	-	TM16	TM15	TM14	TM13	TM12	TM11	TM10	TM1
F90H	-	TM26	TM25	TM24	TM23	TM22	TM21	TM20	TM2

カウント・パルス (CP) 選択ビット (n = 1, 2)

TMn6	TMn5	TMn4	TM1	TM2
0	0	0	設定禁止	設定禁止
0	0	1	設定禁止	設定禁止
0	1	0	タイマ・カウンタ (チャンネル2) のタイマ出力	$f_x/2$
0	1	1	$f_x/2^5$	f_x
1	0	0	$f_x/2^{12}$	$f_x/2^{10}$
1	0	1	$f_x/2^{10}$	$f_x/2^8$
1	1	0	$f_x/2^8$	$f_x/2^6$
1	1	1	$f_x/2^6$	$f_x/2^4$

タイマ・スタート指示ビット

TM23	“ 1 ” を書き込むことによって、カウンタ、IRQTnフラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TM22	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM21	TM20	TM11	TM10	モード
1	0	1	0	16ビット・タイマ・カウンタ・モード

(b) タイマ・カウンタ・コントロール・レジスタ (TC2)

16ビット・タイマ・カウンタ・モードを使用する場合のTC2の設定を図5 - 44に示します (TC2のフォーマットについては図5 - 31 タイマ・カウンタ・コントロール・レジスタのフォーマットを参照してください)。

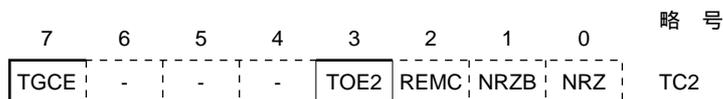
TC2は、8ビット、4ビット操作命令、ビット操作命令により操作します。

TC2は、内部リセット信号発生により00Hにクリアされます。

実線で示されるフラグは、16ビット・タイマ・カウンタ・モードにおける使用フラグを表します。

点線で示されるフラグは、16ビット・タイマ・カウンタ・モードでは使用しないでください (0を設定してください)。

図5 - 44 タイマ・カウンタ・コントロール・レジスタの設定



ゲート制御許可フラグ

TGCE	ゲート制御
0	禁止 (TM2のビット2が“1”にセットされていれば、サンプリング・クロックの状態にかかわらず、カウント動作を行う)
1	許可 (TM2のビット2が“1”にセットされていれば、サンプリング・クロックがハイ・レベルのときカウント動作を行い、ロウ・レベルのときカウント動作を停止する)

タイマ出力許可フラグ

TOE2	タイマ出力
0	禁止 (ロウ・レベルを出力する)
1	許可

(2) タイマ/イベント・カウンタの時間設定

[タイマ設定時間] (周期) は, [モジユロ・レジスタの内容 + 1] をモード・レジスタの設定により選択された [カウント・パルス (CP) 周波数] で割った値となります。

$$T(\text{sec}) = \frac{n+1}{f_{CP}} = (n+1) \cdot (\text{分解能})$$

T (sec) : タイマ設定時間 (秒)

f_{CP} (Hz) : CP周波数 (Hz)

n : モジユロ・レジスタの内容 (n = 0)

タイマは一度設定すれば, その間隔で毎回割り込み要求フラグ (IRQ2) をセットします。

表5 - 8にタイマ・カウンタの各カウント・パルスでの分解能と最長設定時間 (モジユロ・レジスタにFFHをセットしたときの時間) を示します。

★

表5 - 8 分解能と最長設定時間 (16ビット・タイマ時)

(a) タイマ・カウンタ (チャンネル1) の場合

モード・レジスタ			6.0 MHz動作時		4.19 MHz動作時	
TM16	TM15	TM14	分解能	最長設定時間	分解能	最長設定時間
0	1	1	5.33 μs	350 ms	7.63 μs	500 ms
1	0	0	683 μs	44.7 s	977 μs	64.0 s
1	0	1	171 μs	11.2 s	244 μs	16.0 s
1	1	0	42.7 μs	2.80 s	61.0 μs	4.0 s
1	1	1	10.7 μs	699 ms	15.3 μs	1.0 s

(b) タイマ・カウンタ (チャンネル2) の場合

モード・レジスタ			6.0 MHz動作時		4.19 MHz動作時	
TM26	TM25	TM24	分解能	最長設定時間	分解能	最長設定時間
0	1	0	333 ns	21.8 ms	477 ns	31.3 ms
0	1	1	167 ns	10.9 ms	238 ns	15.6 ms
1	0	0	171 μs	11.2 s	244 μs	16.0 s
1	0	1	42.7 μs	2.80 s	61.0 μs	4.0 s
1	1	0	10.7 μs	699 ms	15.3 μs	1.0 s
1	1	1	2.67 μs	175 ms	3.81 μs	250 ms

(3) タイマ・カウンタ動作

タイマ・カウンタ動作は、次のようになります。この動作を行うときは、タイマ・カウンタ・コントロール・レジスタ (TC2) のゲート制御許可フラグ (TGCE) を、0 に設定してください。

図5 - 45にタイマ・カウンタ動作時の構成を示します。

モード・レジスタ (TM1, TM2) の設定によりカウント・パルス (CP) が選択され、カウント・レジスタ (T2) に入力されます。カウント・レジスタ (T1) には、T2のオーバフローが入力されます。

T1とモジュロ・レジスタ (TMOD1) の内容を比較し、一致すると一致信号を発生します。

T2とモジュロ・レジスタ (TMOD2) の内容を比較し、一致すると一致信号を発生します。

と の一致信号が重複すると、割り込み要求フラグ (IRQT2) をセットします。同時にタイマ・アウト・フリップフロップ (TOUT F/F) は反転します。

タイマ・カウンタ動作のタイミングは、図5 - 46のようになります。

タイマ・カウンタ動作は、通常、次の手順により開始します。

TMOD1に16ビット幅で示されるカウント数のうち、上位8ビット分を設定します。

TMOD2に16ビット幅で示されるカウント数のうち、下位8ビット分を設定します。

TM1に動作モード、カウント・パルスを設定します。

TM2に動作モード、カウント・パルス、スタート指示を設定します。

注意 モジュロ・レジスタ (TMOD2) には00H以外の値を設定してください。

タイマ・カウンタ出力端子 (PTO2) を使用する場合は、兼用端子のP22とPCLを次のように設定してください。

P22の出力ラッチをクリアする。

ポート2を出力モードに設定する。

ポート2の内蔵プルアップ抵抗を接続しない状態に、また、PCLを出力禁止にする。

タイマ・カウンタ出力許可フラグ (TOE2) を1にする。

図5 - 45 タイマ・カウンタ動作時の構成

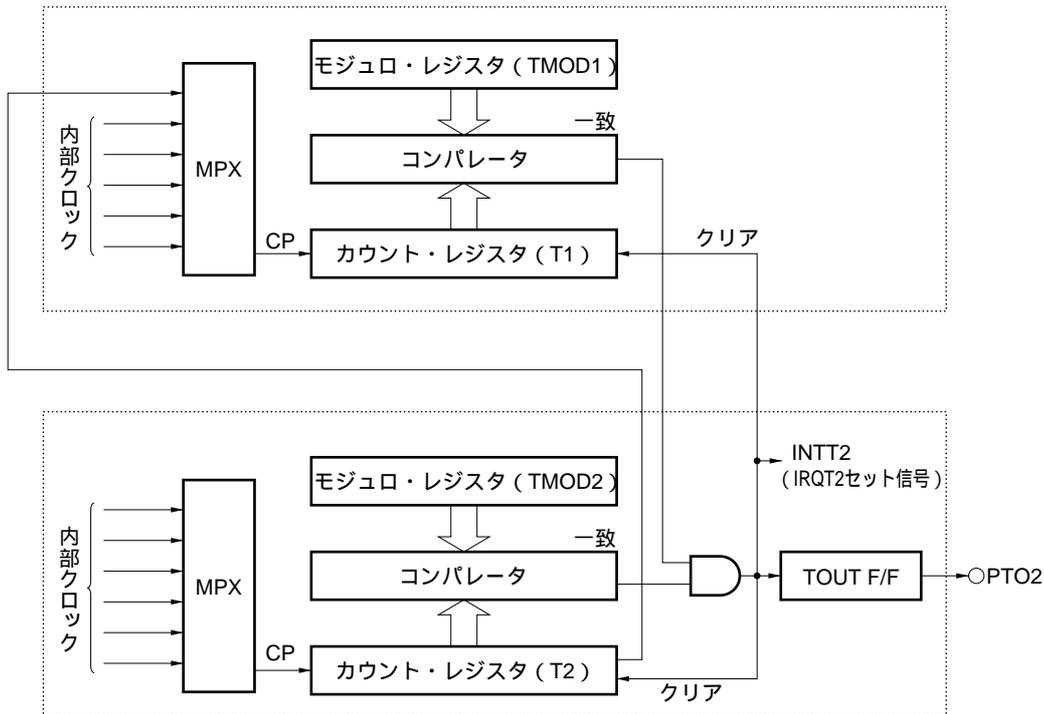
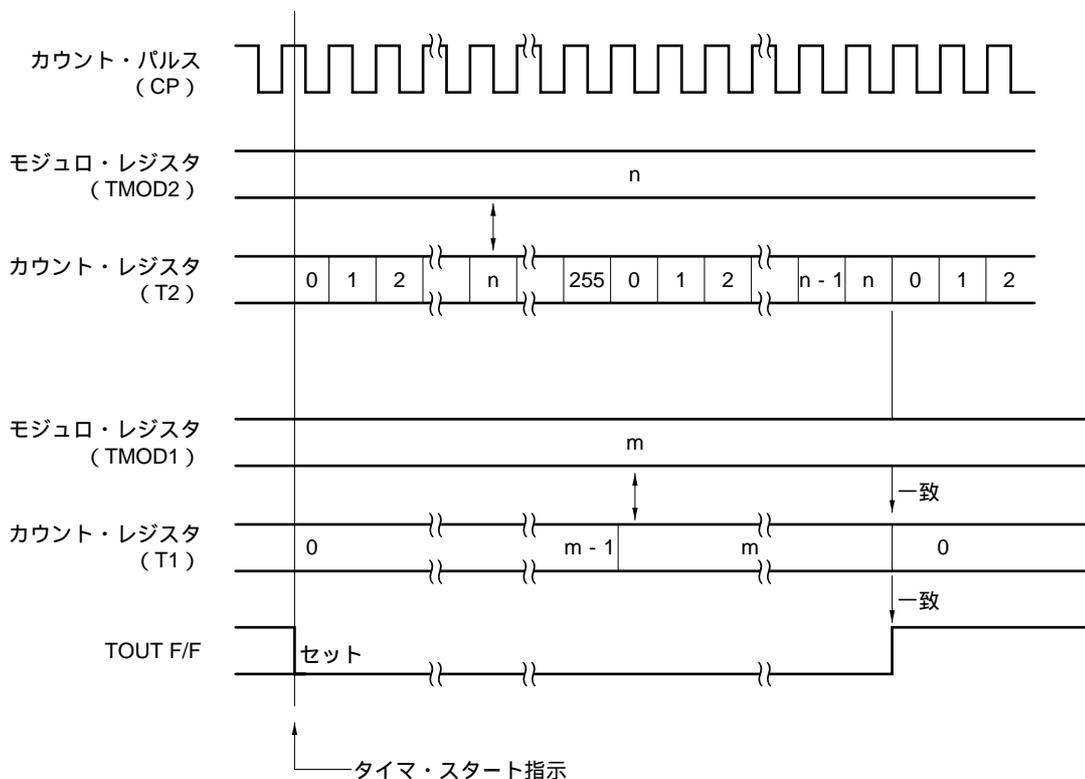


図5 - 46 カウント動作時のタイミング



備考 m : モジユロ・レジスタ (TMOD1) の設定値

n : モジユロ・レジスタ (TMOD2) の設定値

(4) ゲート制御機能を持つカウンタ動作 (16ビット時)

タイマ・カウンタ (チャンネル1) とタイマ・カウンタ (チャンネル2) は、ゲート制御機能を持つカウンタとして使用できます。この機能を使用するときは、タイマ・カウンタ・コントロール・レジスタのゲート制御許可フラグ (TGCE) を 1 に設定してください。

タイマ/イベント・カウンタ (チャンネル0) をカウントすることにより、ゲート信号を生成します。

図5 - 48のようにゲート信号 (T0のTOUT F/Fの出力) がハイ・レベルのとき、タイマ・カウンタ (チャンネル1, チャンネル2) のカウント・パルスのカウントをすることができます (詳細は、(3) **タイマ・カウンタ動作**を参照してください)。

モード・レジスタ (TM1, TM2) の設定によりカウント・パルス (CP) が選択され、ゲート信号がハイ・レベルのときCPがカウント・レジスタ (T2) に入力されます。カウント・レジスタ (T1) には、T2のオーバフローが入力されます。

ゲート信号の立ち上がりエッジと立ち下がりエッジで割り込みが発生します。通常は、立ち下がりエッジの割り込みサブルーチンでT1, T2の内容を読み出し、T1, T2をクリアして次のカウント動作に備えます。

カウンタ動作のタイミングは、図5 - 48のようになります。

カウンタ動作は、通常、次の手順により開始します。

TM1に動作モード、カウント・パルスを設定します。

TM2に動作モード、カウント・パルス、カウンタ・クリア指示を設定します。

TMOD0にカウント数を設定します。

TM0に動作モード、カウント・パルス、スタート指示を設定します。

注意1．モジュロ・レジスタ (TMOD0, TMOD1, TMOD2) には00H以外の値を設定しないでください。

2．タイマ・カウンタ割り込み許可フラグ (IET1) には、1を設定しないでください。

図5-47 カウント動作時の構成

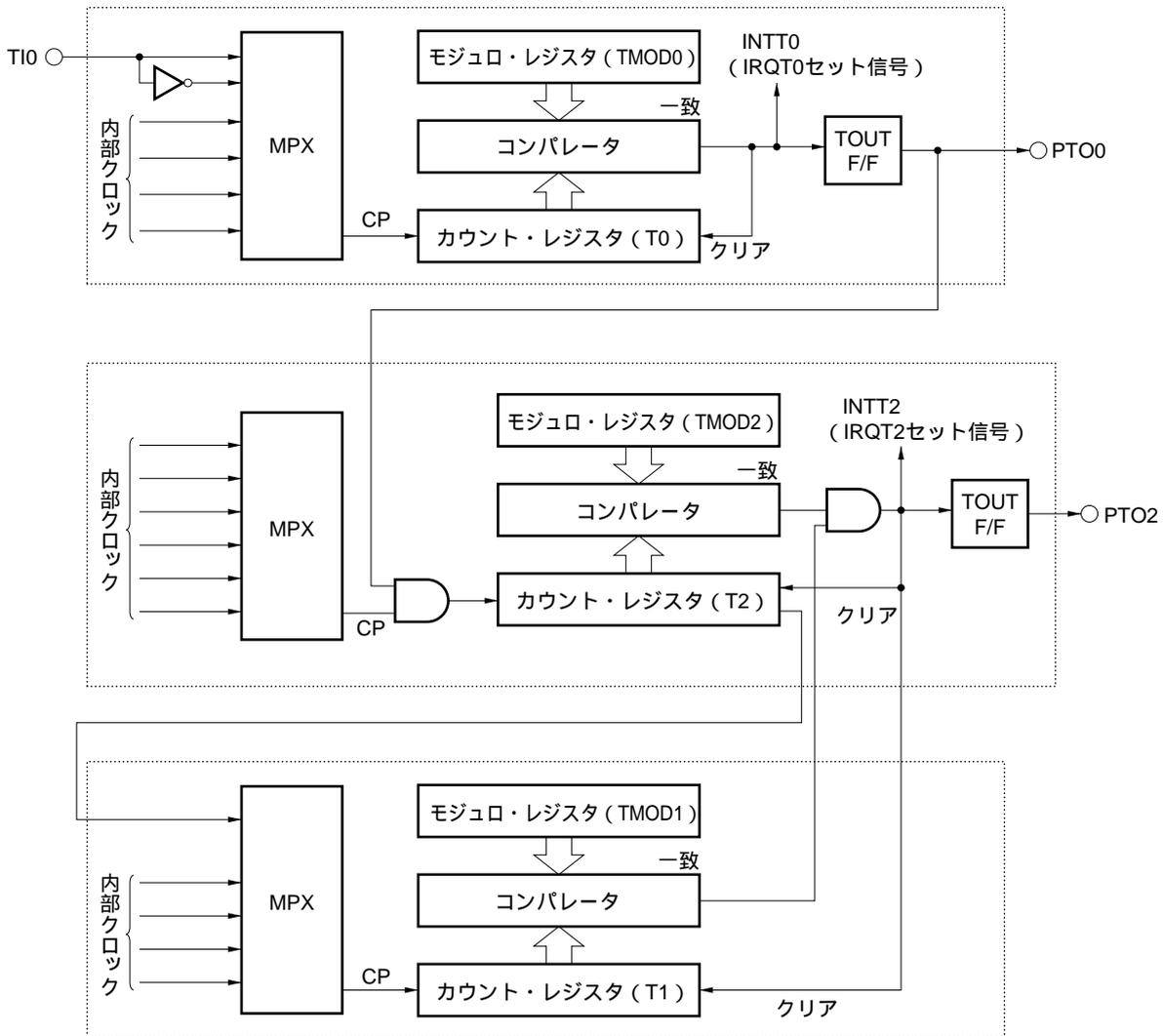
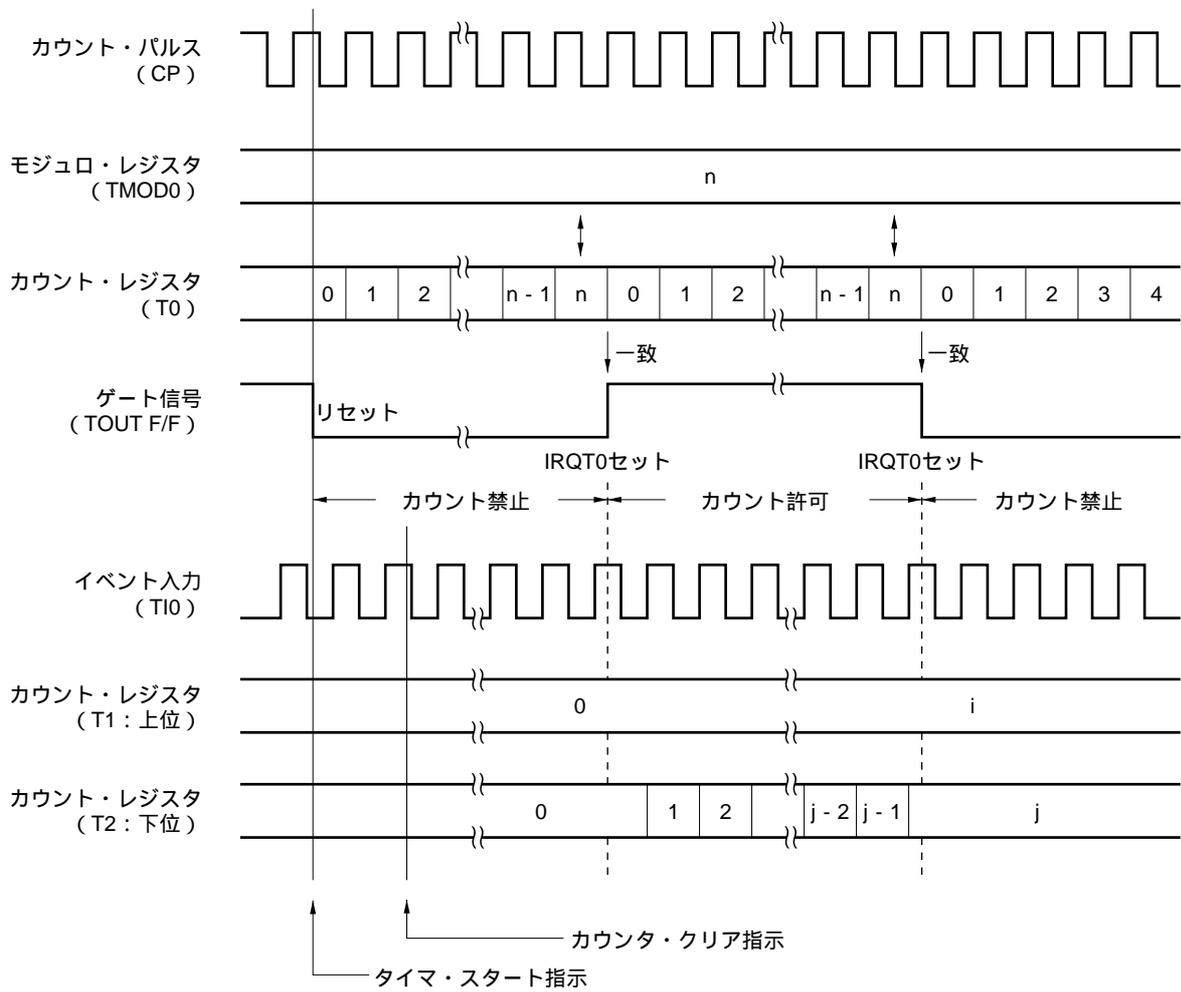


図5 - 48 カウント動作時のタイミング



備考 i : カウント・レジスタ (T1 : 上位) の設定値
 j : カウント・レジスタ (T2 : 下位) の設定値
 n : モジュロ・レジスタ (TMOD0) の設定値

(5) 16ビット・タイマ・カウンタ・モードの応用

例 5 sec間隔で割り込みを発生するインターバル・タイマとして応用します。

- ・モード・レジスタ (TM1) の上位 4 ビットは0010Bとし、タイマ・カウンタ (チャンネル 2) のタイマ出力を選択します。
- ・TM2の上位 4 ビットは0100Bとし、最長設定時間16.0 secを選択します。
- ・TM1の下位 4 ビットは0010Bとし、16ビット・タイマ・カウンタ・モードを選択します。
- ・TM2の下位 4 ビットは1110Bとし、16ビット・タイマ・カウンタ・モード、カウント動作を選択、タイマ・スタートを指示します。
- ・モジュロ・レジスタ (TMOD1, TMOD2) の設定値は、次のとおりです。

$$\frac{5 \text{ sec}}{244 \mu\text{s}} = 20491.8 - 1 = 500\text{BH}$$

プログラム例

```

SEL      MB15                ;またはCLR1 MBE
MOV      XA, #050H
MOV      TMOD1, XA          ;モジュロを設定 (上位 8 ビット分)
MOV      XA, #00BH
MOV      TMOD2, XA          ;モジュロを設定 (下位 8 ビット分)
MOV      XA, #00100010B
MOV      TM1, XA            ;モードを設定
MOV      XA, #01001110B
MOV      TM2, XA            ;モードを設定, タイマ・スタート
DI       IET1                ;タイマ (チャンネル 1) の割り込みを禁止
EI       IET1                ;割り込みを許可
EI       IET2                ;タイマ (チャンネル 2) の割り込みを許可

```

備考 この応用の場合、TI0は入力端子として使用できます。

5.5.5 キャリア・ジェネレータ・モード (CGモード) の動作

キャリア・ジェネレータとして使用するモードです。8ビット長でのキャリア・ジェネレータ動作を行います。

CGモードを使用する場合は、タイマ・カウンタのチャンネル1とチャンネル2を組み合わせて使用します。

タイマ・カウンタ (チャンネル1) は、リモコン信号を生成します。

タイマ・カウンタ (チャンネル2) は、キャリア・クロックを生成します。

(1) レジスタの設定

CGモードは、次の8つのレジスタを使用します。

- ・タイマ・カウンタ・モード・レジスタ (TM1, TM2)
- ・タイマ・カウンタ・コントロール・レジスタ (TC2)^注
- ・タイマ・カウンタ・カウント・レジスタ (T1, T2)
- ・タイマ・カウンタ・モジュロ・レジスタ (TMOD1, TMOD2)
- ・タイマ・カウンタ・ハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H)

注 タイマ・カウンタのチャンネル1は、タイマ・カウンタ出力許可フラグ (TOE1) を使用しません。

(a) タイマ・カウンタ・モード・レジスタ (TM1, TM2)

CGモードを使用する場合のTM1, TM2の設定を図5-49に示します (TM1, TM2のフォーマットについては図5-28 タイマ・カウンタ・モード・レジスタ (チャンネル1) のフォーマット, 図5-29 タイマ・カウンタ・モード・レジスタ (チャンネル2) のフォーマットを参照してください)。

TM1, TM2は、8ビット操作命令により操作します。ビット3は、タイマ・スタート指示ビットで、ビット単位の操作もでき、タイマが動作を開始すると自動的に0クリアされます。

TM1, TM2は、内部リセット信号発生時に00Hにクリアされます。

図5 - 49 タイマ・カウンタ・モード・レジスタの設定 (n = 1, 2)

アドレス	7	6	5	4	3	2	1	0	略号
FA8H	-	TM16	TM15	TM14	TM13	TM12	TM11	TM10	TM1
F90H	-	TM26	TM25	TM24	TM23	TM22	TM21	TM20	TM2

カウント・パルス (CP) 選択ビット

TMn6	TMn5	TMn4	TM1	TM2
0	0	0	設定禁止	設定禁止
0	0	1	設定禁止	設定禁止
0	1	0	タイマ・カウンタ (チャンネル2) のタイマ出力	$f_x/2$
0	1	1	$f_x/2^5$	f_x
1	0	0	$f_x/2^{12}$	$f_x/2^{10}$
1	0	1	$f_x/2^{10}$	$f_x/2^8$
1	1	0	$f_x/2^8$	$f_x/2^6$
1	1	1	$f_x/2^6$	$f_x/2^4$

タイマ・スタート指示ビット

TMn3	“ 1 ” を書き込むことによって、カウンタ、IRQTnフラグをクリア。ビット2が“ 1 ” にセットされていれば、カウント動作をスタート。
------	--

動作モード

TMn2	カウント動作
0	停止 (カウント内容保持)
1	カウント動作

動作モード選択ビット

TM21	TM20	TM11	TM10	モード
1	1	0	0	キャリア・ジェネレータ・モード

(b) タイマ・カウンタ・コントロール・レジスタ (TC2)

CGモードを使用する場合は、タイマ出力許可フラグ (TOE1) とTC2を図5 - 50, 5 - 51のように設定します (TC2のフォーマットについては図5 - 31 **タイマ・カウンタ・コントロール・レジスタのフォーマット**を参照してください)。

TOE1は、ビット操作命令により操作します。またTC2は、8ビット、4ビット操作命令、ビット操作命令により操作します。

TOE1, TC2は、内部リセット信号発生により00Hにクリアされます。

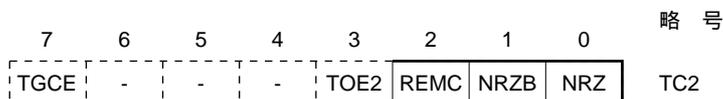
実線で示されるフラグは、CGモードにおける使用フラグを表します。

点線で示されるフラグは、CGモードでは使用しないでください (0を設定してください)。

図5 - 50 タイマ・カウンタ出力許可フラグの設定



図5 - 51 タイマ・カウンタ・コントロール・レジスタの設定



リモコン出力制御フラグ

REMC	リモコン出力
0	NRZ = 1 のとき、キャリア・パルスを出力する
1	NRZ = 1 のとき、ハイ・レベルを出力する

ノー・リターン・ゼロ・パツファ・フラグ

NRZB	次に出力するノー・リターン・ゼロ・データ。 タイマ・カウンタ (チャンネル1) の割り込み発生時、NRZへ転送される。
------	--

ノー・リターン・ゼロ・フラグ

NRZ	ノー・リターン・ゼロ・データ
0	ロウ・レベルを出力する
1	キャリア・パルスまたはハイ・レベルを出力する

(2) キャリア・ジェネレータ動作

キャリア・ジェネレータ動作は、次のようになります。図5 - 52にその構成を示します。

(a) タイマ・カウンタ(チャンネル1)の動作

タイマ・カウンタ(チャンネル1)は、ノー・リターン・ゼロ・パツファ・フラグ(NRZB)からノー・リターン・ゼロ・フラグ(NRZ)へのリロード間隔を決定します。タイマ・カウンタ(チャンネル1)の動作は、次のようになります(詳細は5.5.2 8ビット・タイマ/イベント・カウンタ・モードの動作を参照してください)。

モード・レジスタ(TM1)の設定によりカウント・パルス(CP)が設定され、カウント・レジスタ(T1)に入力されます。

T1とモジュロ・レジスタ(TMOD1)の内容を比較し、一致すると一致信号を発生し、割り込み要求フラグ(IRQT1)をセットします。同時にタイマ・アウト・フリップフロップ(TOUT F/F)は反転します。

(b) タイマ・カウンタ(チャンネル2)の動作

タイマ・カウンタ(チャンネル2)は、キャリア・クロックを発生し、ノー・リターン・ゼロ・データに従って、キャリアを出力します。タイマ・カウンタ(チャンネル2)の動作は、次のようになります(詳細は5.5.3 PWMパルス・ジェネレータ・モード(PWMモード)の動作を参照してください)。

モード・レジスタ(TM2)の設定によりカウント・パルス(CP)が選択され、カウント・レジスタ(T2)に入力されます。

T2とハイ・レベル期間設定用モジュロ・レジスタ(TMOD2H)の内容を比較し、一致すると一致信号を発生し、タイマ・アウト・フリップフロップ(TOUT F/F)は反転します。

T2とモジュロ・レジスタ(TMOD2)の内容を比較し、一致すると一致信号を発生し、割り込み要求フラグ(IRQT2)をセットします。同時にTOUT F/Fは反転します。

この動作を繰り返します。

ノー・リターン・ゼロ・データは、タイマ・カウンタ(チャンネル1)に割り込みが発生すると、NRZBからNRZへリロードされます。

リモコン出力制御フラグ(REMC)の設定によりNRZ=1のとき、キャリア・クロックまたはハイ・レベルを出力します。また、NRZ=0のときロウ・レベルを出力します。

キャリア・ジェネレータ動作のタイミングは、図5 - 53のようになります。

キャリア・ジェネレータ動作は、通常、次の手順により開始します。

TMOD2Hにキャリア・クロックのハイ・レベルのカウント数を設定します。

TMOD2にキャリア・クロックのロウ・レベルのカウント数を設定します。

REMCに出力波形の形態を設定します。

TM2に動作モード，カウント・パルス，スタート指示を設定します。

TMOD1にカウント数を設定します。

TM1に動作モード，カウント・パルス，スタート指示を設定します。

タイマ・カウンタ（チャンネル1）に割り込みが発生する前に随時，次に出力するノー・リターン・ゼロ・データをNRZBに設定します。

注意 モジュロ・レジスタ（TMOD1, TMOD2, TMOD2H）には00H以外の値を設定してください。

タイマ・カウンタ出力端子（PTO1）を使用する場合は，兼用端子のP21を次のように設定してください。

P21の出力ラッチをクリアする。

ポート2を出力モードに設定する。

ポート2の内蔵プルアップ抵抗を接続しない状態にする。

タイマ・カウンタ出力許可フラグ（TOE1）を1にする。

図5 - 52 キャリア・ジェネレータ動作時の構成

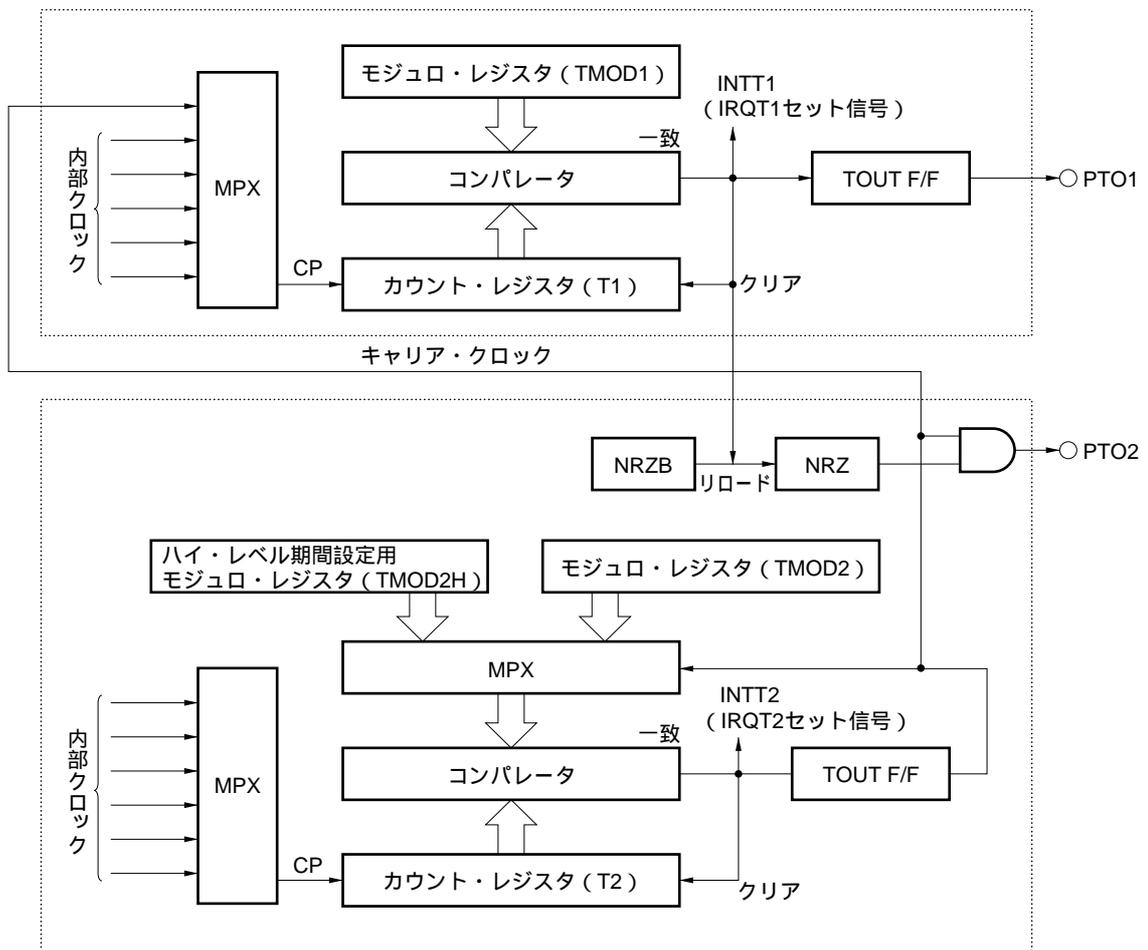
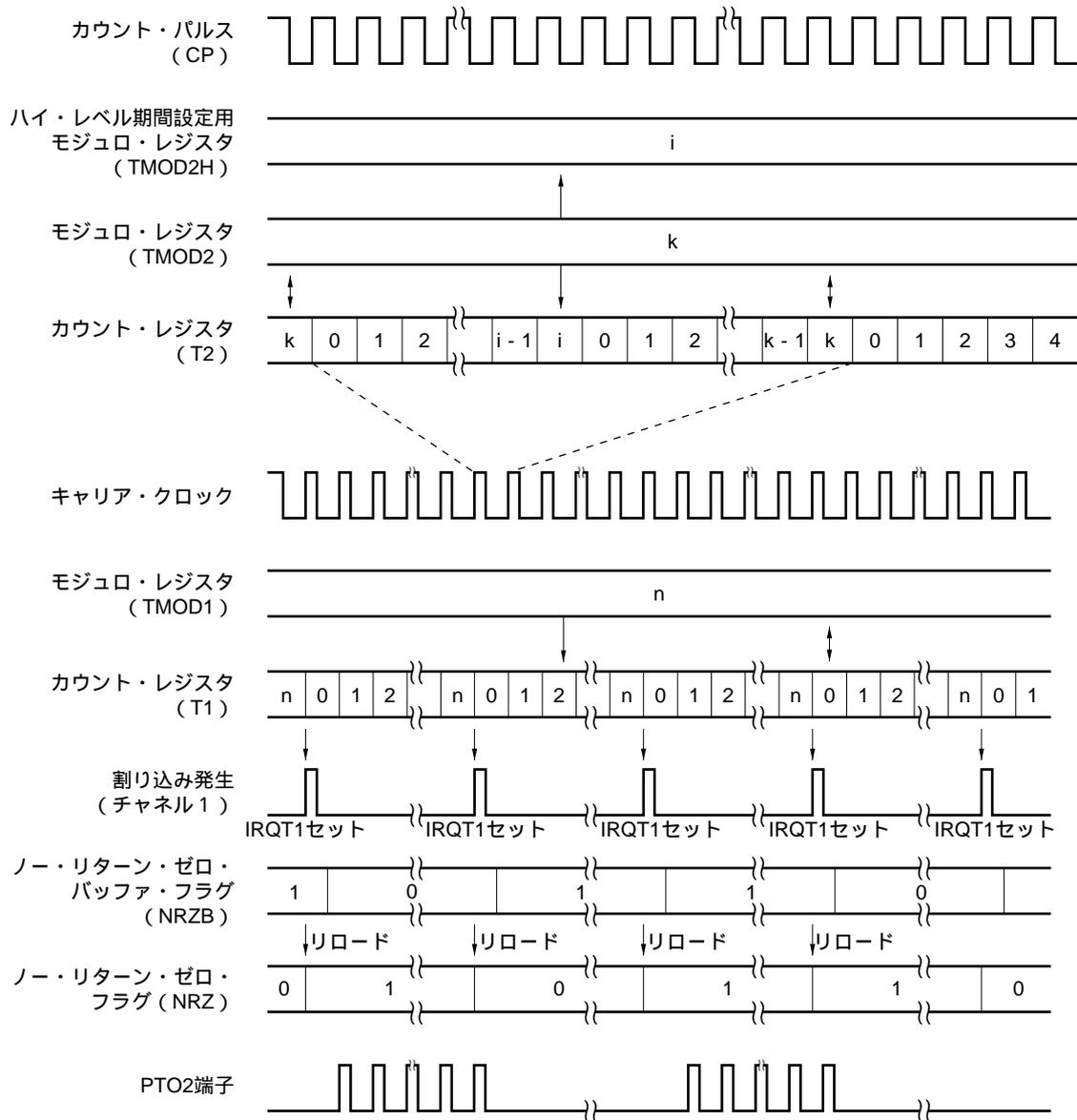


図5 - 53 キャリア・ジェネレータ動作時のタイミング



備考 i : ハイ・レベル期間設定用モジュール・レジスタ (TMOD2H) の設定値

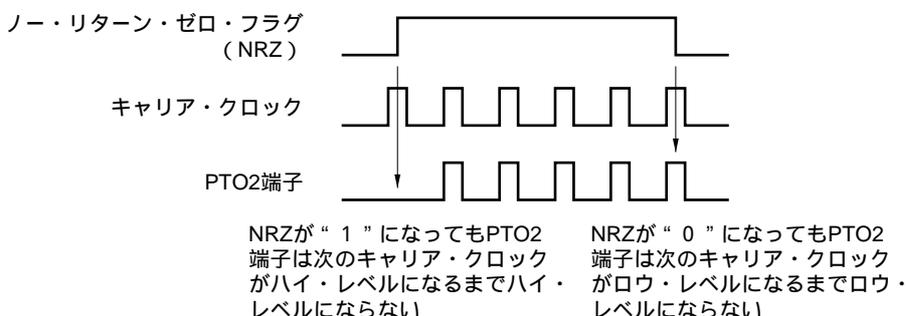
k : モジュール・レジスタ (TMOD2) の設定値

n : モジュール・レジスタ (TMOD1) の設定値

備考 PTO2端子がハイ・レベルのとき（ノー・リターン・ゼロ・フラグ（NRZ）が“0”かつキャリア・クロックがハイ・レベルのとき）、タイマ・カウンタ（チャンネル1）の割り込みが発生した場合は、PTO2端子の出力は、キャリア・クロックが次にハイ・レベルになるまで、更新されたNRZの内容に従いません。

またPTO2端子がハイ・レベルのとき（NRZが“1”かつキャリア・クロックがハイ・レベルのとき）、タイマ・カウンタ（チャンネル1）の割り込みが発生した場合は、PTO2端子の出力は、キャリア・クロックがロウ・レベルになるまで、更新されたNRZの内容に従いません。

この処理は、出力するキャリアのハイ・レベル・パルス幅を一定にするためです（次図参照）。



(3) CGモードの応用

リモコン送信用のキャリア・ジェネレータとして応用します。

周波数が38.0 kHz（周期が26.3 μs）、デューティ比が1/3のキャリア・クロックを発生します。

- ・モード・レジスタ（TM2）の上位4ビットは0011Bとし、最長設定時間61.1 μsを選択します。
- ・TM2の下位4ビットは1111Bとし、CGモード、カウント動作を選択、タイマ・スタートを指示します。
- ・タイマ出力許可フラグ（TOE2）は“1”とし、タイマ出力を許可します。
- ・ハイ・レベル期間設定用モジュロ・レジスタ（TMOD2H）の設定値は、次のとおりです。

$$\frac{1}{3} \cdot \frac{26.3 \mu s}{239 ns} - 1 = 36.7 - 1 \quad 36 = 24H$$

- ・モジュロ・レジスタ（TMOD2）の設定値は、次のとおりです。

$$\frac{2}{3} \cdot \frac{26.3 \mu s}{239 ns} - 1 = 73.4 - 1 \quad 72 = 48H$$

プログラム例

```

SEL    MB15                ;またはCLR1 MBE
MOV    XA, #024H
MOV    TMOD2H, XA         ;モジュロを設定(ハイ・レベル期間)
MOV    XA, #48H
MOV    TMOD2, XA          ;モジュロを設定(ロウ・レベル期間)
MOV    XA, #00111111B
MOV    TM2, XA            ;モードを設定, タイマ・スタート

```

キャリア・クロックを出力する期間が9 ms, ロウ・レベルを出力する期間が4.5 msのリーダー・コードを出力します(次頁の図参照)。

- ・モード・レジスタ(TM1)の上位4ビットは0110Bとし, 最長設定時間15.6 msを選択します。
- ・TM1の下部4ビットは1100Bとし, 8ビット・タイマ・カウンタ・モード, カウント動作を選択, タイマ・スタートを指示します。
- ・モジュロ・レジスタ(TMOD1)の初回の設定値は, 次のとおりです。

$$\frac{9 \text{ ms}}{61 \mu\text{s}} - 1 = 147.5 - 1 \quad 146 = 92\text{H}$$

- ・TMOD1の書き換え時の設定値は, 次のとおりです。

$$\frac{4.5 \text{ ms}}{61 \mu\text{s}} - 1 = 73.7 - 1 \quad 73 = 49\text{H}$$

- ・TC2の上位4ビットは0000Bとし, ゲート制御を禁止します。
- ・TC2の下部4ビットは0000Bとし, ノー・リターン・ゼロ・データが“1”のときキャリア・クロックを出力し, 次に出力するノー・リターン・ゼロ・データを“0”にします。

プログラム例

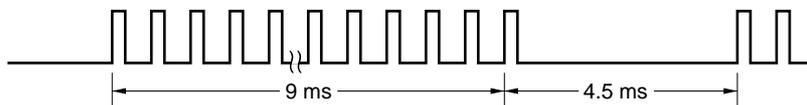
```

SEL    MB15                ;またはCLR1 MBE
MOV    XA, #092H
MOV    TMOD1, XA          ;モジュロを設定(キャリア・クロック出力期間)
MOV    XA, #00000000B
MOV    TC2, XA
SET1   NRZ                ;ノー・リターン・ゼロ・データを“1”に設定
MOV    XA, #01101100B
MOV    TM1, XA            ;モードを設定, タイマ・スタート
EI                                           ;割り込みを許可
EI    IET1                ;タイマ(チャンネル1)の割り込みを許可

```

； サブルーチン

```
MOV    XA, #049H
MOV    TMOD1, XA          ; モジユロを書き換える（ロウ・レベル出力期間）
RETI
```

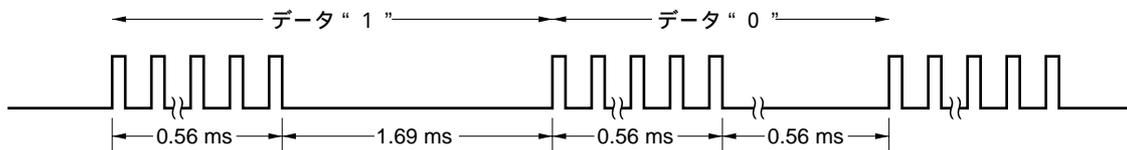


データが“ 1 ”のときキャリア・クロックを出力する期間が0.56 ms，ロウ・レベルを出力する期間が1.69 ms，データが“ 0 ”のときキャリア・クロックを出力する期間が0.56 ms，ロウ・レベルを出力する期間が0.56 msのカスタム・コードを出力します（下図参照）。

- ・モード・レジスタ（TM1）の上位4ビットは0011Bとし，最長設定時間1.95 msを選択します。
- ・TM1の下位4ビットは1100Bとし，8ビット・タイマ・カウンタ・モード，カウント動作を選択，タイマ・スタートを指示します。
- ・モジユロ・レジスタ（TMOD1）の初回の設定値は，次のとおりです。

$$\frac{0.56 \text{ ms}}{7.64 \text{ } \mu\text{s}} - 1 = 73.3 - 1 = 72 = 48\text{H}$$

- ・TMOD1のキャリア出力を行わない期間は，データ“ 0 ”のときは出力期間と同じ長さ，データ“ 1 ”のときは出力期間の3倍の長さで処理をします。
- ・TC2の上位4ビットは0000Bとし，ゲート制御を禁止します。
- ・TC2の下位4ビットは0000Bとし，ノー・リターン・ゼロ・データが“ 1 ”のときキャリア・クロックを出力し，次に出力するノー・リターン・ゼロ・データを“ 0 ”にします。
- ・送信データ（“ 0 ”または“ 1 ”）をビット・シーケンシャル・バッファに設定しておきます。



プログラム例

PTO2端子の出力ラッチは“0”に固定され、かつ端子の入出力モードは出力モードに設定された状態を想定しています。また、キャリア・クロックの生成は、前例のプログラムで発生中の状態を想定しています。

```

; SEND_CARRIER_DATA_PRO

        SEL    MB15          ; または CLR1 MBE
        MOV    HL, #00H     ; LにBSB (ビット・シーケンシャル・バッ
                           ; ファ)のポインタを設定, HはBSBのビッ
                           ; ト・データ一時退避エリアとして使用

; CG_Init & Send_1st_Data
        MOV    XA, #48H
        MOV    TMOD1, XA   ; モジュロ・レジスタ (キャリア・クロック出
                           ; 力期間)を設定
        MOV    XA, #0000000B ; ゲート制御を禁止, キャリア・クロックの出
                           ; 力可能状態, NRZB, NRZを0に初期化

        MOV    TC2, XA
        SET1   NRZ         ; ノー・リターン・ゼロ・フラグに“1”を設
                           ; 定
        MOV    XA, #01101100B ; カウント・パルスを選択, 8ビット・タイマ・
                           ; カウンタ・モード
        MOV    TM1, XA     ; タイマ・カウント動作許可, タイマ・スター
                           ; トを指示

; Send_1st_Data
        CALL   !GET_DAT    ; BSBからデータをゲット
        CALL   !SEND_D_0  ; データが0および1のキャリア出力と1回分
                           ; のロウ・レベル出力期間設定処理
        SKE    H, #1H     ; ビット0が1なら, ロウ・レベル出力期間を
                           ; 2回分追加処理へ
        BR     SEND_1_F   ; ビット0が0なら, ロウ・レベル出力の状態
                           ; で, 次のデータのサーチへ処理を受け流す
        CALL   !SEND_D_1  ; ロウ・レベル出力期間を2回分追加処理
                           ; PTO2端子からロウ・レベル出力の状態で,
                           ; BSBのビット0-Fのデータ送信処理へ処理を
                           ; 移す

SEND_1_F :                               ; BSBのビット0-Fのデータ送信処理

        SET1   NRZB       ; 前回のデータのロウ・レベル出力期間中に,
                           ; 次に発生するIRQT1で, 次に送信するデー
                           ; タのキャリアが出力されるようにNRZB = 1にす
                           ; する
        INCS   L          ; 送信中のデータをカウントし, Lが0FH 0H
                           ; でデータ送信終了
        BR     LOOP_C_0
        BR     SEND_END

LOOP_C_0 : SKTCLR IRQT1   ; 前のデータのロウ・レベル出力をウエイト
                           ; (前のデータの終了を確認)
        BR     LOOP_C_0
        CLR1   NRZB       ; キャリア出力を開始
                           ; 次に発生するIRQT1で, 1回目のロウ・レベ
                           ; ル出力が行われるように, あらかじめNRZB
                           ; を0にクリアする

        CALL   !GET_DAT
        CALL   !SEND_D_0
        SKE    H, #1H     ; ゲットしたデータが1なら, ロウ・レベル出
                           ; 力期間を2回分追加処理 (SEND_D_1)へ
        BR     SEND_1_F   ; データが0なら, PTO2端子からロウ・レベ
                           ; ル出力の状態でのデータの送信処理へ

        CALL   !SEND_D_1
        BR     SEND_1_F

SEND_END :                               ; 16ビット分のデータ送信終了

```

; サブルーチン

```
GET_DAT : ; @Lで示されるBSBのデータをサーチ。Hレジスタに値を設定

        SKT    BSB0, @L
        MOV    A, #0
        MOV    A, #1
        MOV    H, A
        RET

SEND_D_0 : ; データが0および1のキャリア出力と1回分のロウ・レベル出力設定処理

LOOP_1st : SKTCLR IRQT1
        BR     LOOP_1st ; キャリア出力をウエイト
        RET     ; 1回目のロウ・レベル出力を開始

SEND_D_1 :
        CLR1   NRZB ; データが1のとき、2回目のロウ・レベル出力を設定

LOOP_2nd : SKTCLR IRQT1
        BR     LOOP_2nd ; 1回目のロウ・レベル出力をウエイト
        ; 2回目のロウ・レベル出力を開始
        CLR1   NRZB ; 3回目のロウ・レベル出力を設定

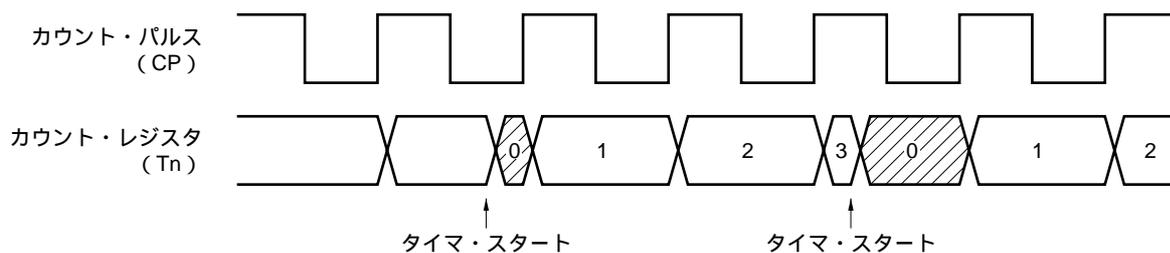
LOOP_3rd : SKTCLR IRQT1
        BR     LOOP_3rd ; 2回目のロウ・レベル出力をウエイト
        ; 3回目のロウ・レベル出力を開始

        RET
```

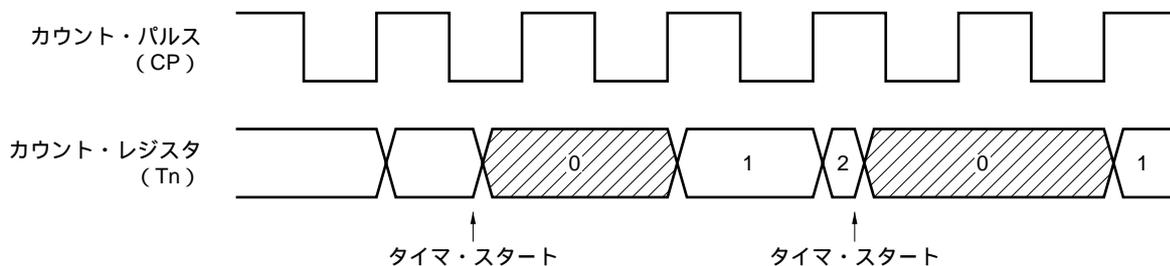
5.5.6 タイマ/イベント・カウンタ応用上の注意

(1) タイマ・スタート時の誤差

タイマ・スタート (TMnのビット3を“1”にセット) 後、一致信号が発生するまでの時間は (モジュロ・レジスタの内容 + 1) × 分解能の式で求めた値に対し、最大でカウント・パルス (CP) 1 クロック分の誤差が生じます。これは次の図に示すように、CPに対してカウント・レジスタTnのクリアが非同期で行われるためです。



またCPの周波数が1マシン・サイクル以上の場合、タイマ・スタート (TMnのビット3を“1”にセット) 後、一致信号が発生するまでの時間は、(モジュロ・レジスタの内容 + 1) × 分解能の式で求めた値に対し、最大でCP2クロック分の誤差が生じます。これは、次の図に示すようにCPに対してTnのクリアがCPUクロックを基に非同期で行われるためです。



(2) タイマ・スタート時の注意

通常タイマ・スタート (TMnのビット3を“1”にセット) によってカウント・レジスタTnと割り込み要求フラグIRQnはクリアされますが、タイマが動作モードにあり、IRQnのセットと、タイマ・スタートが同一タイミングで発生した場合には、IRQnをクリアできない場合が生じます。これは、IRQnをベクタ割り込みとして使用しているときは問題ありませんが、IRQnをテストする応用の場合には、タイマ・スタートを行ったにもかかわらず、IRQnがセットされていることになり不都合を生じます。したがって、IRQnが立つ可能性のあるタイミングでタイマ・スタートする場合にはいったん、タイマを停止 (TMnのビット2を“0”にする) させてから再スタートするか、タイマ・スタートを2回行うようにしてください。

例 IRQnが立つ可能性のあるタイミングでのタイマ・スタート

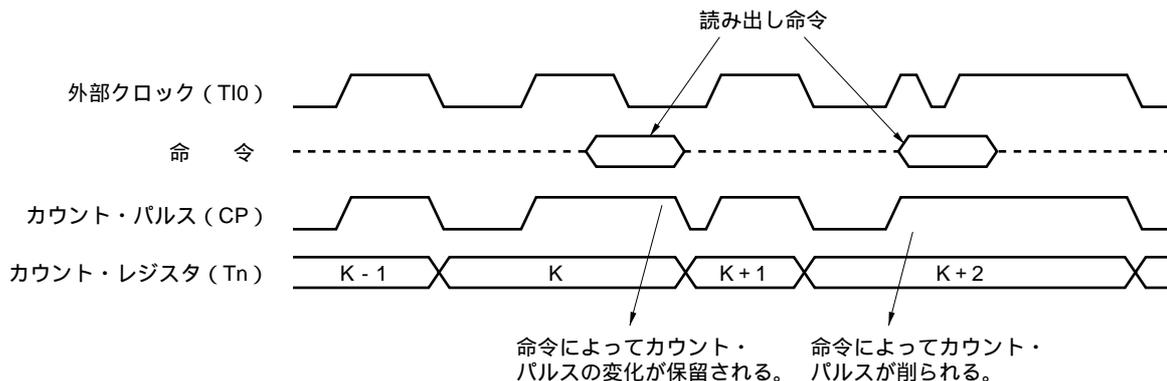
```

SEL    MB15
MOV    XA, #0
MOV    TMn, XA    ; タイマを停止
MOV    XA, #4CH
MOV    TMn, XA    ; 再スタート
または
SEL    MB15
SET1   TMn. 3
SET1   TMn. 3    ; 再スタート
    
```

(3) カウント・レジスタ読み出し時の誤差

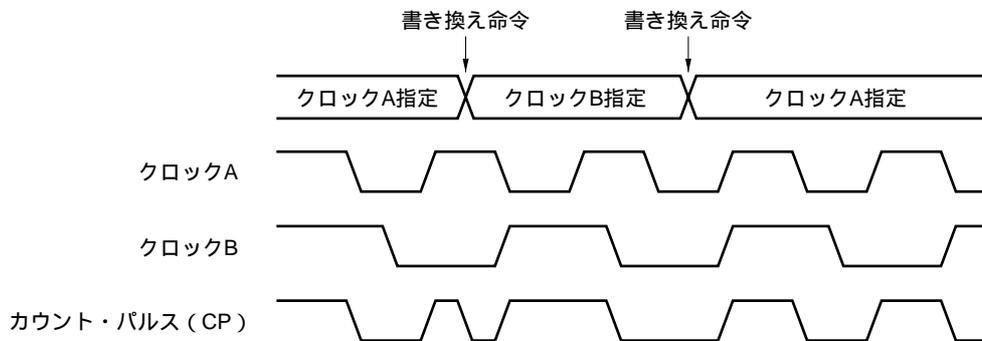
カウント・レジスタ (Tn) の内容は、8ビット・データ・メモリ操作命令によっていつでも読み出し可能ですが、この命令の動作中、カウント・パルス (CP) の変化が保留され、Tnが変化しないようになっています。このため、CPの信号源をTI0入力にした場合、命令実行時間だけCPが削られます (内部クロックをCPとした場合は、命令と同期するのでこの現象は起こりません)。

したがって、TI0入力をCPとして入力し、Tnの内容を読み取る場合は、CPが削られてもミス・カウントしないパルス幅の信号を入力しなければなりません。すなわち、読み出し命令によってカウントが保留される期間は、1マシン・サイクルとなっていますから、TI0に入力するパルスはそれ以上の幅が必要となります。

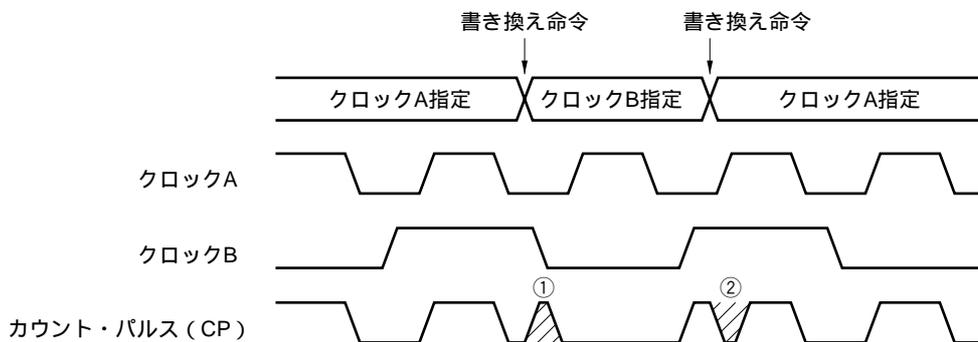


(4) カウント・パルス変更時の注意

タイマ/イベント・カウンタ・モード・レジスタ (TMn) を書き換えてカウント・パルス (CP) を変更した場合、その指定は命令実行直後から有効になります。

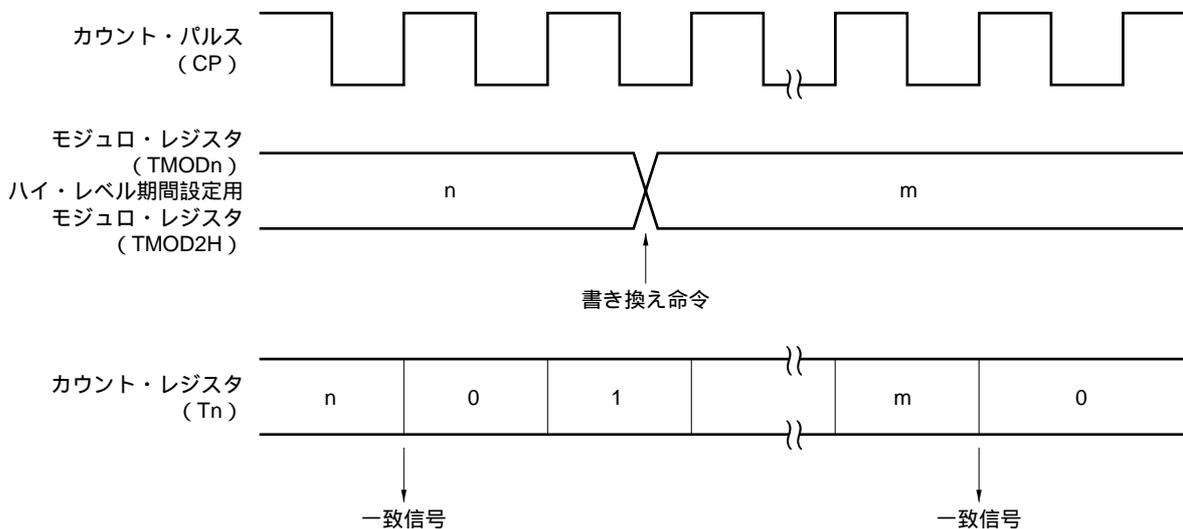


CP変更時のクロックの組み合わせにより、次図のようにヒゲ状 (または) のCPが発生することもあります。この場合、ミス・カウントあるいはカウンタ・レジスタ (Tn) の内容が破壊される可能性がありますので、CPを変更する場合は必ず、TMnのビット3を“1”にして、タイマの再スタートも同時に行ってください。

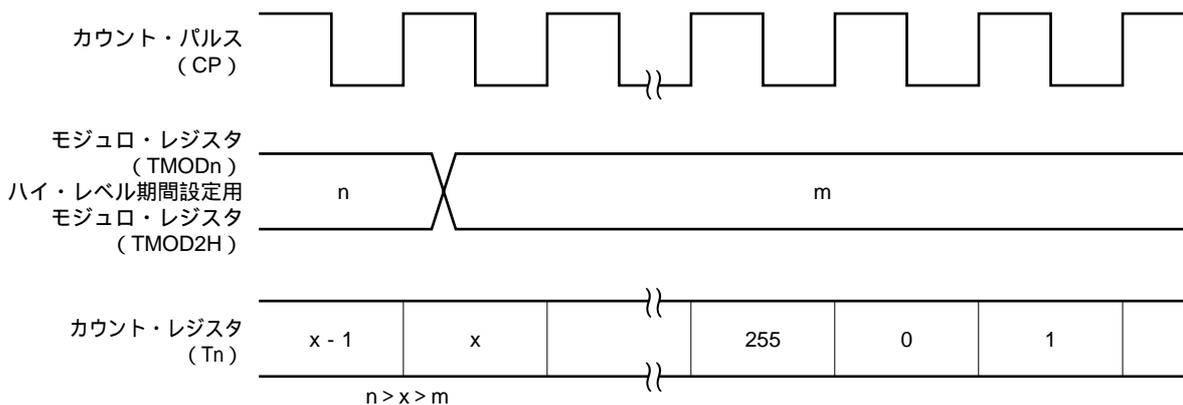


(5) モジュール・レジスタ変更後の動作

モジュール・レジスタ (TMODn) とハイ・レベル期間設定用モジュール・レジスタ (TMOD2H) の変更は、8ビット・データ・メモリ操作命令を実行した時点で行われます。



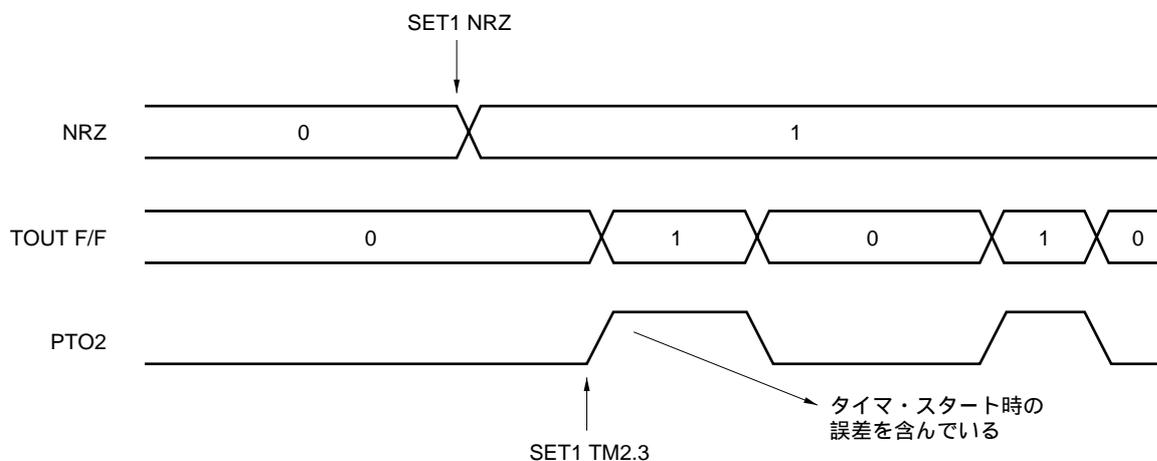
TMODnの変更後の値が、カウント・レジスタ (Tn) の値よりも小さい場合には、Tnはカウントを継続しオーバフローして0から再カウントします。したがって、TMODn, TMOD2Hの変更後の値 (m) が変更前の値 (n) より小さい場合には、TMODn, TMOD2Hを変更後、タイマを再スタートさせる必要があります。



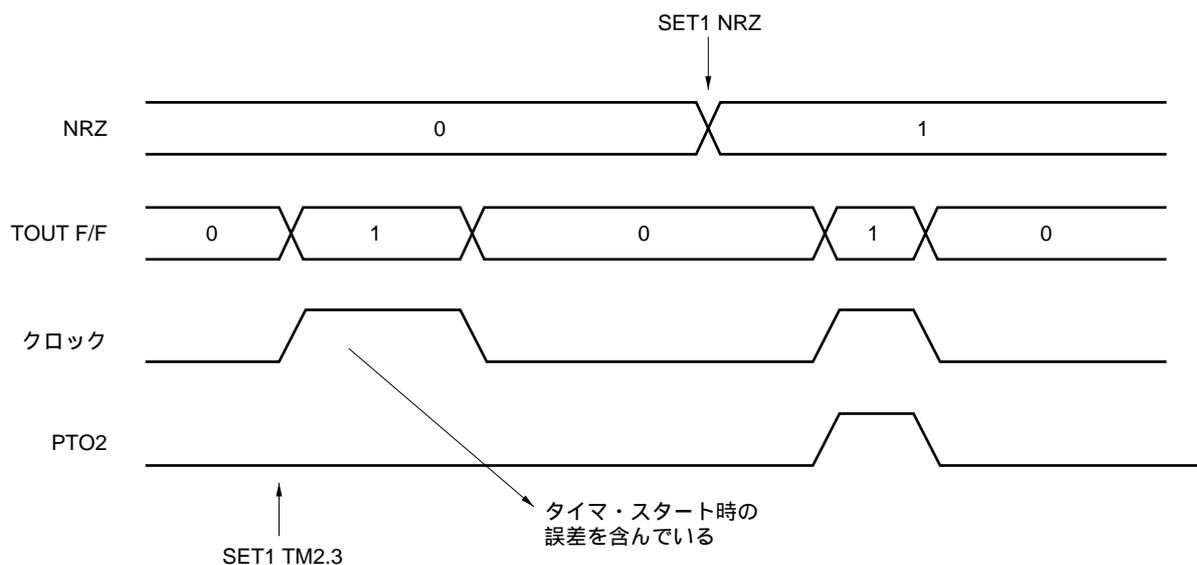
(6) キャリア・ジェネレータ応用上の注意 (スタート時)

キャリア・クロックを発生させるとき、タイマ・スタート (TM2のビット3を“1”にセット) 後、初回のキャリア・クロックのハイ・レベル期間は、(モジュロ・レジスタの内容+1) × 分解能の式で求めた値に対し、最大でカウント・パルス (CP) 1クロック分 (CPの周波数が1マシン・サイクル以上の場合、最大でCP2クロック分) の誤差が生じます (詳細は、(1) タイマ・スタート時の誤差を参照してください)。

初回のコードとしてキャリアを出力する場合、ノー・リターン・ゼロ・フラグ (NRZ) を“1”にセットしたあとにタイマ・スタート (TM2のビット3を“1”にセット) を行うと、初回のキャリア・クロックのハイ・レベル期間はタイマ・スタート時の誤差が含まれます。



したがって、初回のコードとしてキャリアを出力する場合、タイマ・スタート (TM2のビット3を“1”にセット) したあとに、NRZを“1”にセットするようにしてください。

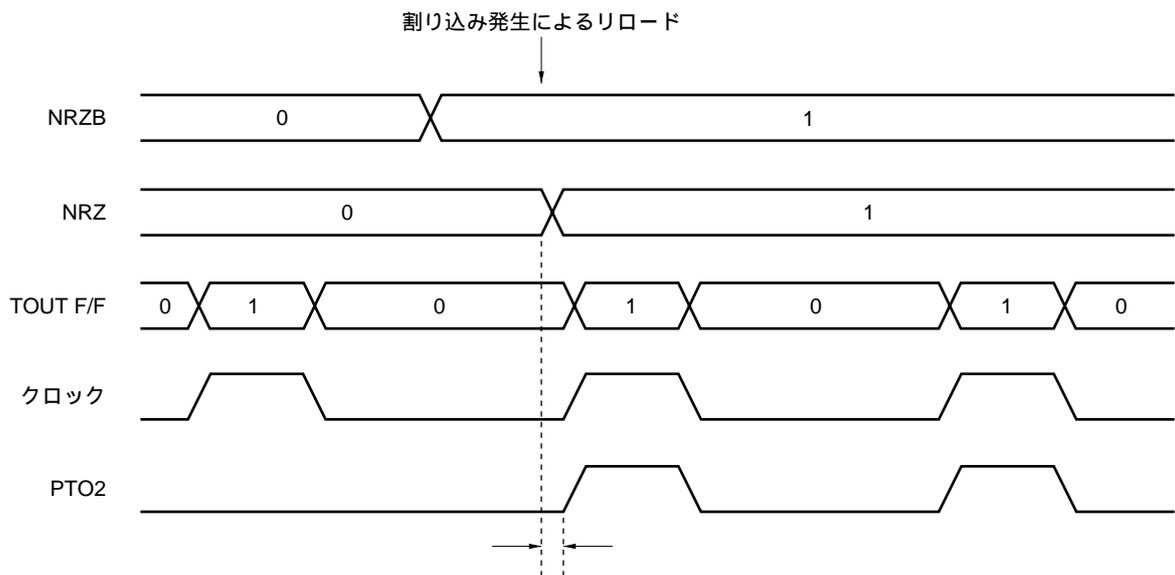


(7) キャリア・ジェネレータ応用上の注意 (リロード時)

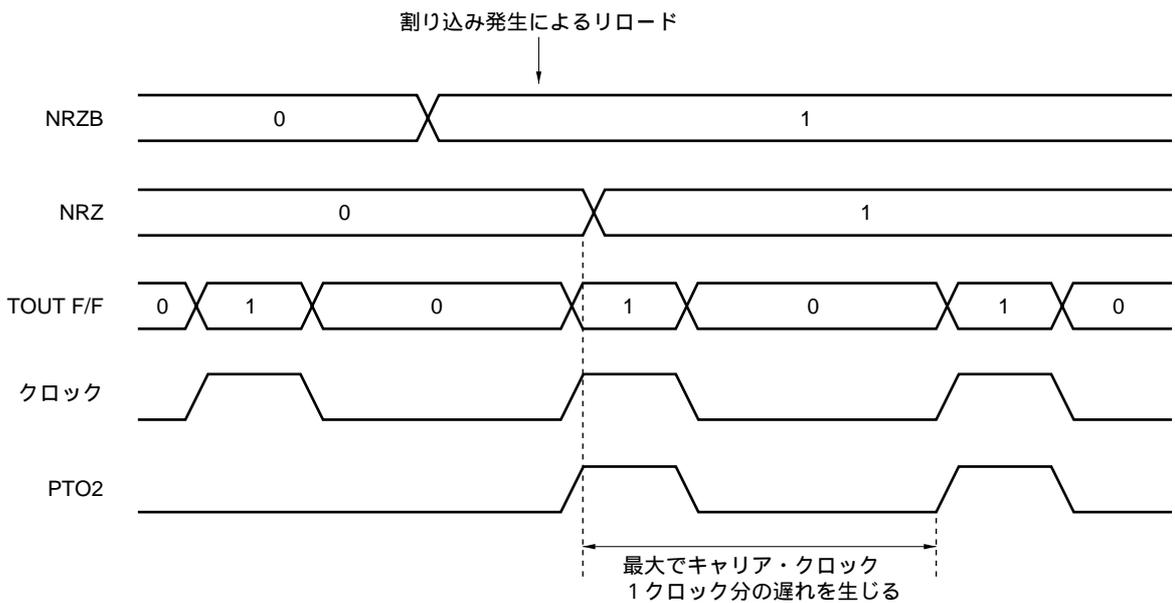
PTO2端子にキャリアを出力するとき、リロード (タイマ・カウンタ (チャンネル1) の割り込み発生により、ノー・リターン・ゼロ・バッファ・フラグ (NRZB) の内容がノー・リターン・ゼロ・フラグ (NRZ) へ転送され、NRZの内容が “ 1 ” に更新) 後、初回のキャリアが発生するまでの時間は、最大でキャリア・クロック 1クロック分の遅れを生じます。

これは、次の図に示すように、キャリア・クロックに対してリロードが非同期で行われるためであり、またキャリアのハイ・レベル期間を一定に保つためでもあります。

リロード後の遅れが最小の場合

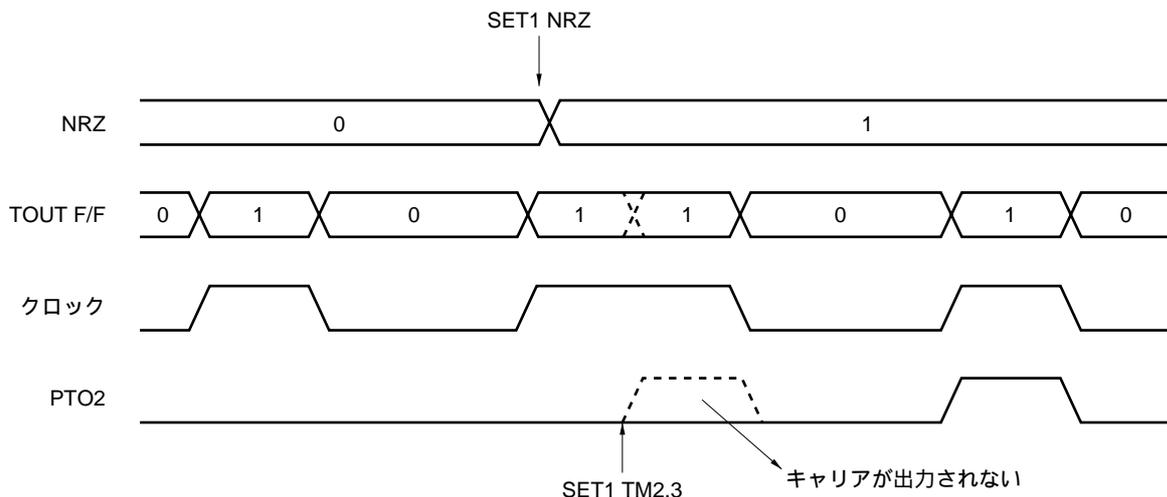


リロード後の遅れが最大の場合

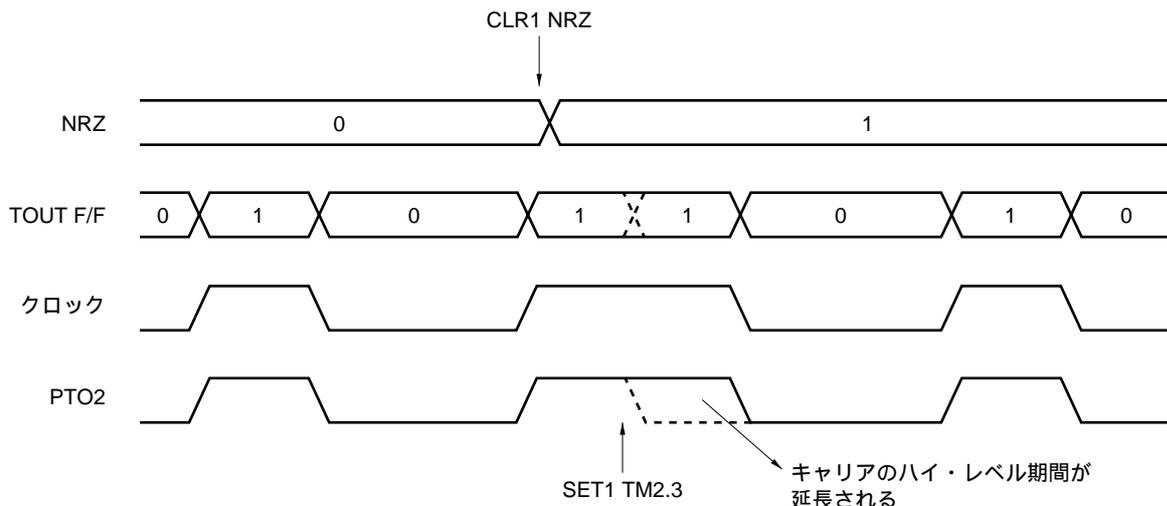


(8) キャリア・ジェネレータ応用上の注意(再スタート時)

キャリア・クロックがハイ・レベル(TOUT F/Fが“1”を保持)のとき、ノー・リターン・ゼロ・フラグ(NRZ)の内容を直接書き換えることにより強制的にリロードを行い、さらにタイマ再スタート(TM2のビット3を“1”にセット)を行う場合、次の図に示すようにPTO2端子にキャリアが出力されないことがあります。



同様にキャリア・クロックがハイ・レベル(TOUT F/Fが“1”を保持)のとき、NRZの内容を直接書き換えることにより強制的にリロードを行い、さらにタイマ再スタート(TM2のビット3を“1”にセット)を行う場合、次の図に示すようにPTO2端子に出力されるキャリアのハイ・レベル期間が長くなる場合があります。



5.6 シリアル・インタフェース

5.6.1 シリアル・インタフェースの機能

μPD753208は、クロック同期式8ビット・シリアル・インタフェースを内蔵しており、次の4種類のモードがあります。

以下に、各モードの機能概要を示します。

(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。消費電力を低減することができます。

(2) 3線式シリアルI/Oモード

シリアル・クロック (\overline{SCK})、シリアル出力 (SO)、シリアル入力 (SI) の3本のラインによって8ビット長のデータの転送を行います。

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は速くなります。

シリアル転送する8ビット・データの先頭ビットをMSBか、またはLSBかに切り替えることができるので、いずれの先頭ビットのデバイスとも接続が可能です。

3線式シリアルI/Oモードでは、75XLシリーズ、75Xシリーズ、78Kシリーズ、各種周辺I/Oデバイスとの接続が可能です。

(3) 2線式シリアルI/Oモード

シリアル・クロック (\overline{SCK})、シリアル・データ・バス (SB0、またはSB1) の2本のラインによって8ビット長のデータの転送を行います。2本のラインへの出力レベルをソフトウェアで操作することにより、複数のデバイスと通信できます。

また、 \overline{SCK} とSB0 (または、SB1) はソフトウェアによる出力レベルの操作が可能のため、任意の転送フォーマットに対応することができます。したがって、従来、複数のデバイスを接続する場合に必要なハンドシェイクのためのラインを削減することができ、入出力ポートの有効活用ができます。

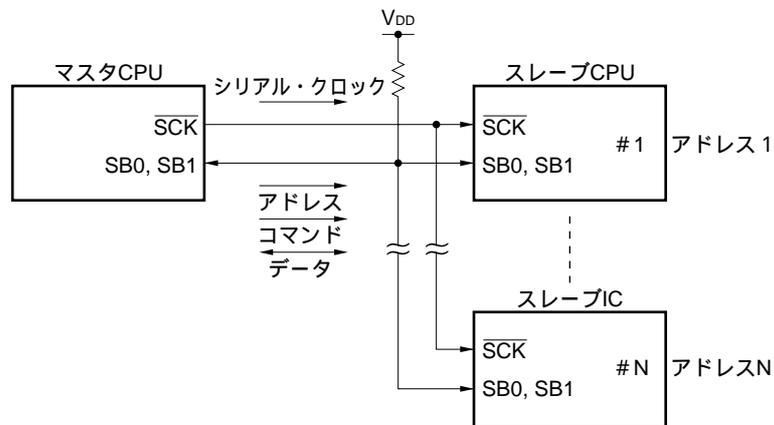
(4) SBIモード (シリアル・バス・インタフェース・モード)

シリアル・クロック (\overline{SCK})、シリアル・データ・バス (SB0、またはSB1) の2本のラインにより複数のデバイスと通信可能なモードです。

NECのシリアル・バス・フォーマットに準拠します。

SBIモードでは、送信側はシリアル・データ・バス上にシリアル通信の対象デバイス選択のための“アドレス”、対象デバイスに対して指令を与える“コマンド”および実際の“データ”を出力することができます。また、受信側は受信したデータをハードウェアにより、“アドレス”、“コマンド”、“データ”に判別することができます。この機能により、2線式シリアルI/Oモードと同様、入出力ポートの有効活用ができるほか、さらに、応用プログラムのシリアル・インタフェースの制御部分を簡単にすることができます。

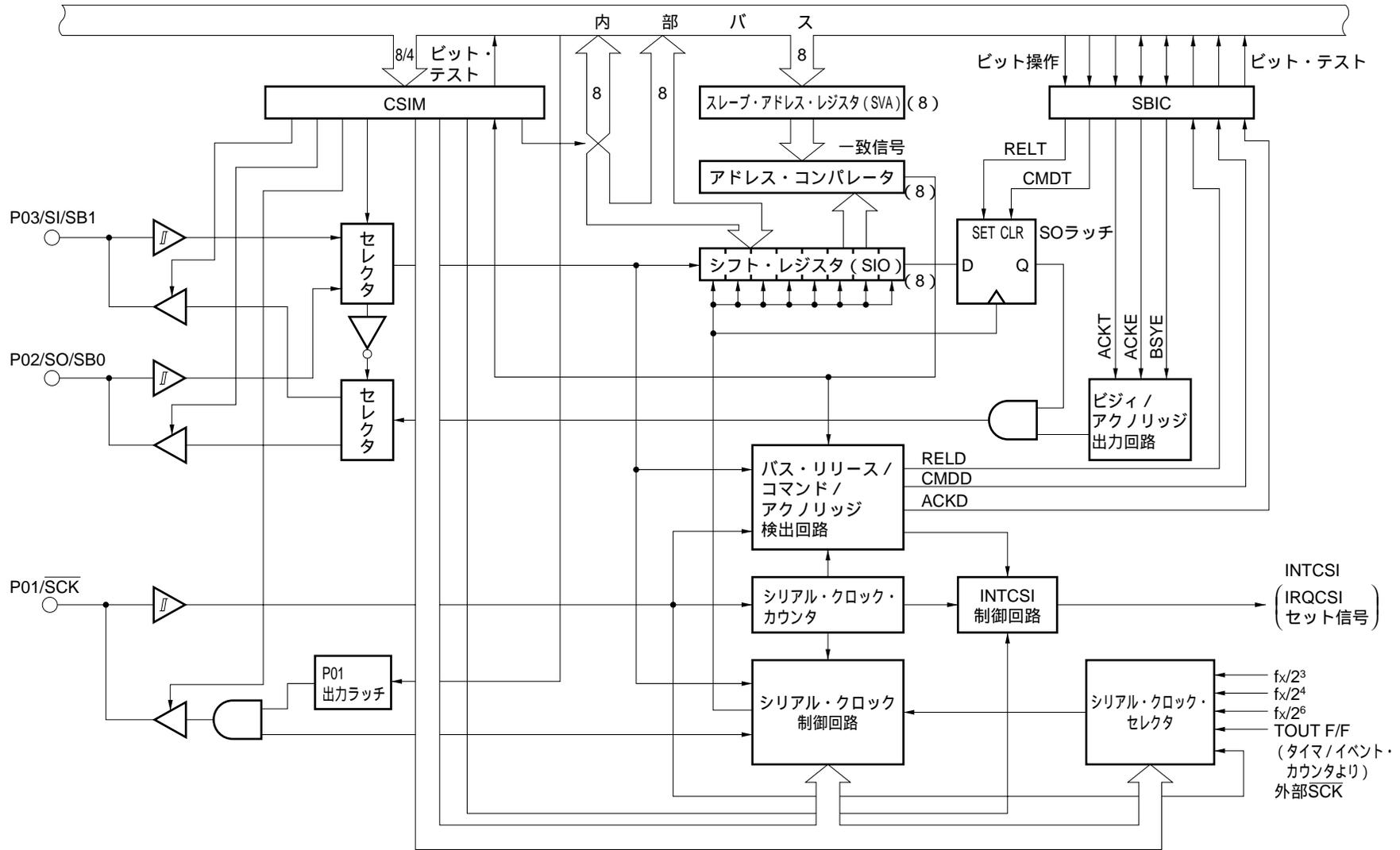
図5 - 54 SBIのシステム構成例



5.6.2 シリアル・インタフェースの構成

図5 - 55にシリアル・インタフェースのブロック図を示します。

図5 - 55 シリアル・インタフェースのブロック図



(1) シリアル動作モード・レジスタ (CSIM)

シリアル・インタフェースの動作モード、シリアル・クロック、ウエイク・アップ機能などを指定する8ビット・レジスタです(詳細は、5.6.3(1)シリアル動作モード・レジスタ(CSIM)を参照)。

(2) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

シリアル・バスの状態を制御するビットと、シリアル・バスからの入力データの各種状態を示すフラグから構成される8ビット・レジスタです。おもにSBIモードで使用します(詳細は5.6.3(2)シリアル・バス・インタフェース・コントロール・レジスタ(SBIC)を参照)。

(3) シフト・レジスタ (SIO)

8ビットのシリアル・データをパラレル・データに、8ビットのパラレル・データをシリアル・データに変換するレジスタです。シリアル・クロックに同期して送受信動作(シフト動作)を行います。SIOに対する書き込みにより、実際の送受信動作が制御されます(詳細は、5.6.3(3)シフト・レジスタ(SIO)を参照)。

(4) SOラッチ

SO/SB0、SI/SB1端子レベルを保持するラッチです。ソフトウェアにより直接制御することもできます。SBIモードでは、 $\overline{\text{SCK}}$ の8発目終了時にセットされます(詳細、5.6.3(2)シリアル・バス・インタフェース・コントロール・レジスタ(SBIC)を参照)。

(5) シリアル・クロック・セクタ

使用するシリアル・クロックを選択します。

(6) 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。

(7) スレープ・アドレス・レジスタ (SVA), アドレス・コンパレータ**SBIモード時**

μ PD753208をスレープ・デバイスとするときに使用します。スレープは、自分の指定番号(スレープ・アドレス値)をSVAにセットします。マスタは、特定のスレープを選択するためのスレープ・アドレスを出力します。

アドレス・コンパレータにより、マスタから受信したスレープ・アドレスとSVAの値を比較して、一致すると、そのスレープが選択されたことになります。

2線式シリアル/Oモード, SBIモード時

μPD753208をマスタまたはスレーブとして送信する場合に、エラー検出を行います（詳細は、5.6.3（4）スレーブ・アドレス・レジスタ（SVA）を参照）。

（8）INTCSI制御回路

割り込み要求の発生を制御します。次の場合に割り込み要求（INTCSI）を発生し、割り込み要求フラグ（IRQCSI）をセットします（図6-1 割り込み制御回路ブロック図参照）。

3線式および2線式シリアル/Oモード時

シリアル・クロックを8発カウントするたびに割り込み要求を発生します。

SBIモード時

WUP^注 = “ 0 ” のとき...シリアル・クロックを8発カウントするたびに割り込み要求を発生します。

WUP^注 = “ 1 ” のとき...アドレス受信後、SVAとSIOの値が一致したときに割り込み要求を発生します。

注 WUP...ウエイク・アップ機能指定ビット（CSIMのビット5）

（9）シリアル・クロック制御回路

シフト・レジスタへのシリアル・クロックの供給を制御します。また、内部システム・クロック使用時は、 $\overline{\text{SCK}}$ 端子へ出力するクロックの制御も行います。

（10）ビジィ/アクノリッジ出力回路, パス・リリース/コマンド/アクノリッジ検出回路

SBIモード時に各種制御信号の出力および検出を行います。

3線式および2線式シリアル/Oモード時には動作しません。

（11）P01出力ラッチ

シリアル・クロックが8発終了したあとに、ソフトウェアでシリアル・クロックを発生するためのラッチです。

リセット入力により“ 1 ”に設定されます。

シリアル・クロックとして内部システム・クロックを選択する場合は、P01出力ラッチを“ 1 ”に設定してください。

5.6.3 レジスタの機能

(1) シリアル動作モード・レジスタ (CSIM)

図5 - 56にシリアル動作モード・レジスタ (CSIM) のフォーマットを示します。

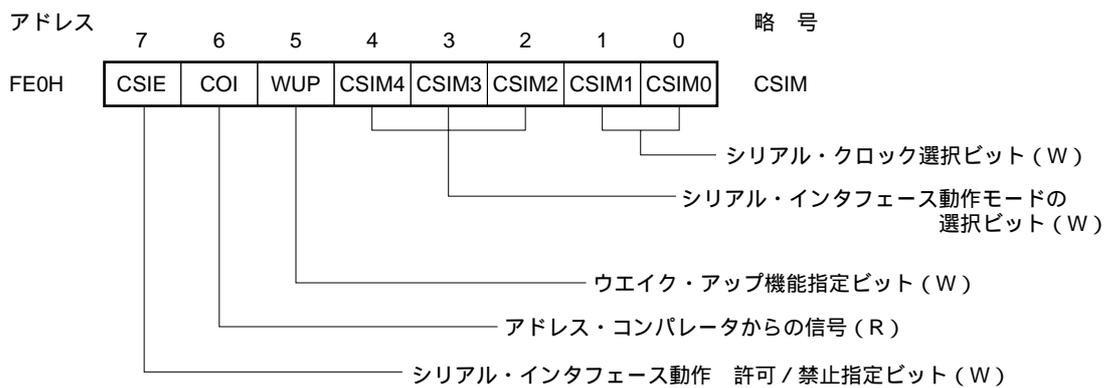
CSIMは、シリアル・インタフェースの動作モード、シリアル・クロック、ウエイク・アップ機能などを指定する8ビット・レジスタです。

操作は、8ビット・メモリ操作命令で行います。上位3ビットについては、1ビット単位の操作もできます。ビット操作の場合は各ビットの名称を用いて操作します。

ビットにより読み出し/書き込み動作の可否は異なります(図5 - 56参照)。ビット6は、ビット・テストのみ可能で、書き込んだデータは無効となります。

$\overline{\text{RESET}}$ 信号発生により、全ビット0にクリアされます。

図5 - 56 シリアル動作モード・レジスタ (CSIM) のフォーマット (1/4)



備考1 . (R) 読み出しのみ可能

2 . (W) 書き込みのみ可能

図5 - 56 シリアル動作モード・レジスタ (CSIM) のフォーマット (2/4)

シリアル・インタフェース動作 許可/禁止指定ビット (W)

		シフト・レジスタの動作	シリアル・クロック・カウンタ	IRQCSIフラグ	SO/SB0, SI/SB1端子
CSIE	0	シフト動作禁止	クリア	保持	ポート0機能専用
	1	シフト動作可能	カウント動作	セット可能	各モードでの機能とポート0兼用

備考1 . CSIE, CSIM3, CSIM2の設定により各モードを選択できます。

CSIE	CSIM3	CSIM2	動作モード
0	x	x	動作停止モード
1	0	x	3線式シリアルI/Oモード
1	1	0	SBIモード
1	1	1	2線式シリアルI/Oモード

2 . P01/SCK端子はCSIE, CSIM1, CSIM0の設定により次の状態になります。

CSIE	CSIM1	CSIM0	P01/SCK端子の状態
0	0	0	入力ポート (P01)
1	0	0	ハイ・インピーダンス (SCK入力)
0	1	0	ハイ・レベル出力
0	0	1	
0	1	1	
1	1	0	シリアル・クロック出力
1	0	1	(ハイ・レベル出力 : シリアル転送終了時)
1	1	1	

3 . シリアル転送中にCSIEをクリアするときは、次の手順で行ってください。

割り込み許可フラグ (IECSI) をクリアして割り込み禁止状態にする。

CSIEをクリアする。

割り込み要求フラグ (IRQCSI) をクリアする。

例1 . シリアル・クロックに $fx/2^4$ を選択し、各シリアル転送終了時にシリアル割り込みIRQCSIを発生させ、SB0端子をシリアル・データ・バスとしてSBIモードでシリアル転送を行うモードを選択する。

```
SEL    MB15          ; または CLR1 MBE
MOV    XA, #10001010B
MOV    CSIM, XA      ; CSIM 10001010B
```

2 . CSIMの内容に従うシリアル転送を可能な状態にする。

```
SEL    MB15          ; または CLR1 MBE
SET1   CSIE
```

図5 - 56 シリアル動作モード・レジスタ (CSIM) のフォーマット (3/4)

アドレス・コンパレータからの信号 (R)

COI ^注	クリアされる条件 (COI = 0)	セットされる条件 (COI = 1)
	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致しなかったとき	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致したとき

注 COIの読み出しはシリアル転送開始前、完了後にはのみ有効です。転送中には不定の値が読み出されます。また、8ビット操作命令によって書き込まれたCOIデータは無視されます。

ウエイク・アップ機能指定ビット (W)

WUP	0	各モードでのシリアル転送終了時に毎回IRQCSIをセットする。
	1	SBIモード時のみに使用。バス・リリース後に受信したアドレスがスレーブ・アドレス・レジスタのデータと一致したとき (ウエイク・アップ状態) のみIRQCSIをセットする。SB0, SB1はハイ・インピーダンス。

注意 $\overline{\text{BUSY}}$ 信号出力中にWUP = 1 にすると、 $\overline{\text{BUSY}}$ が解除されません。SBIでは $\overline{\text{BUSY}}$ 解除指示後、次のシリアル・クロック ($\overline{\text{SCK}}$) の立ち上がりまで $\overline{\text{BUSY}}$ 信号が出力されます。WUP = 1 とするときは、必ず $\overline{\text{BUSY}}$ を解除したのちにSB0 (またはSB1) 端子がハイ・レベルになったことを確認してから、WUP = 1 としてください。

シリアル・インタフェース動作モードの選択ビット (W)

CSIM4	CSIM3	CSIM2	動作モード	シフト・レジスタのビット順	SO/SB0/P02端子機能	SI/SB1/P03端子機能
x	0	0	3線式シリアル I/Oモード	SIO ₇₋₀ XA (MSB先頭で転送)	SO (CMOS出力)	SI (CMOS入力)
		1		SIO ₀₋₇ XA (LSB先頭で転送)		
0	1	0	SBIモード	SIO ₇₋₀ XA (MSB先頭で転送)	SB0 (N-chオープン・ドレイン) 入出力	P03 (CMOS入力)
1		P02 (CMOS入力)			SB1 (N-chオープン・ドレイン) 入出力	
0	1	1	2線式シリアル I/Oモード	SIO ₇₋₀ XA (MSB先頭で転送)	SB0 (N-chオープン・ドレイン) 入出力	P03 (CMOS入力)
1		P02 (CMOS入力)			SB1 (N-chオープン・ドレイン) 入出力	

備考 x : don't care

図5 - 56 シリアル動作モード・レジスタ (CSIM) のフォーマット (4/4)

シリアル・クロック選択ビット (W)

CSIM1	CSIM0	シリアル・クロック			SCK端子 モード
		3線式シリアルI/Oモード	SBIモード	2線式シリアルI/Oモード	
0	0	SCK端子への外部からの入力クロック			入 力
0	1	タイマ/イベント・カウンタ出力 (T0)			出 力
1	0	$f_x/2^4$ (375 kHz : 6.00 MHz時, 262 kHz : 4.19 MHz時)		$f_x/2^6$ (93.8 kHz : 6.00 MHz時, 65.5 kHz : 4.19 MHz時)	
1	1	$f_x/2^3$ (750 kHz : 6.00 MHz時, 524 kHz : 4.19 MHz時)			

(2) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

図5 - 57にシリアル・バス・インタフェース・コントロール・レジスタ (SBIC) のフォーマットを示します。

SBICはシリアル・バスを制御するビットとシリアル・バスからの入力データの各種状態を示すフラグから構成されている8ビット・レジスタで、おもにSBIモードで使用されます。

操作は、ビット操作命令で行います。4ビットまたは8ビット・メモリ操作命令で操作することはできません。

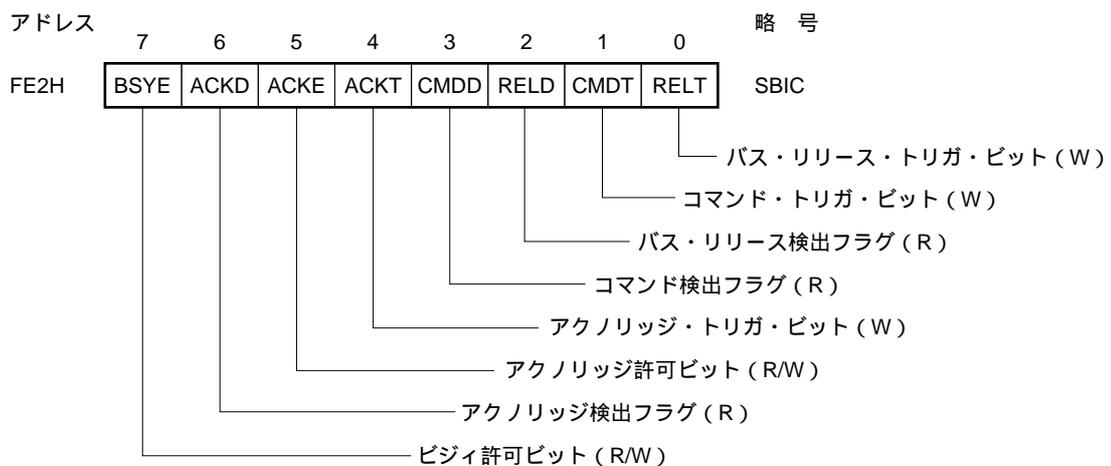
ビットにより読み出し/書き込み動作の可否は異なります (図5 - 57参照)。

RESET信号発生により全ビット0にクリアされます。

注意 3線式および2線式シリアルI/Oモードでは次のビット以外は使用できません。

- ・バス・リリース・トリガ・ビット (RELT) SOラッチのセット
- ・コマンド・トリガ・ビット (CMDT) SOラッチのクリア

図5 - 57 シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) のフォーマット (1/3)



- 備考 1 . (R) 読み出しのみ可能
 2 . (W) 書き込みのみ可能
 3 . (R/W) 読み出し/書き込みともに可能

図5 - 57 シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) のフォーマット (2/3)

ビジー許可ビット (R/W)

BSYE	0	ビジー信号の自動出力禁止 クリア命令実行直後の、 \overline{SCK} の立ち下がりに同期して、ビジー信号の出力を停止
	1	アクノリッジ信号に続き \overline{SCK} の立ち下がりに同期して、ビジー信号を出力

例1 . コマンド信号を出力する。

```
SEL MB15 ;またはCLR1 MBE
SET1 CMDT
```

2 . RELD, CMDDをテストし、受信データの種別を判別して処理を分ける。

なお、この割り込みルーチンはWUP = 1 とし、一致アドレスの場合のみ処理されるようにする。

```
SEL MB15
SKF RELD ;RELDをテスト
BR !ADRS
SKT CMDD ;CMDDをテスト
BR !DATA
BR !CMD

CMD ; ..... ; コマンドを解釈
DATA ; ..... ; データを処理
ADRS ; ..... ; アドレスをデコード
```

アクノリッジ検出フラグ (R)

ACKD	クリアされる条件 (ACKD = 0)	セットされる条件 (ACKD = 1)
	転送スタート時 \overline{RESET} 信号発生時	アクノリッジ信号 (\overline{ACK}) 検出時 (\overline{SCK} の立ち上がりに同期します)

アクノリッジ許可ビット (R/W)

ACKE	0	アクノリッジ信号 (\overline{ACK}) の自動出力を禁止します (ACKTIによる出力は可能)。	
	1	転送終了前にセットした場合	\overline{SCK} の9クロック目に同期して \overline{ACK} を出力します。
		転送終了後にセットした場合	セット命令実行直後の \overline{SCK} に同期して \overline{ACK} を出力します。

アクノリッジ・トリガ・ビット (W)

ACKT	転送終了後にセットすると、次の \overline{SCK} に同期して \overline{ACK} が出力されます。 \overline{ACK} 信号出力後、自動的にクリア (0) されます。
------	---

注意1 . シリアル転送終了前、および転送中にセット (1) しないでください。

- 2 . ACKTIは、ソフトウェアでクリアすることはできません。
- 3 . ACKTIをセットするときは、ACKE = 0 にしてください。

図5 - 57 シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) のフォーマット (3/3)

コマンド検出フラグ (R)

CMDD	クリアされる条件 (CMDD = 0)	セットされる条件 (CMDD = 1)
	転送スタート命令実行時 バス・リリース信号 (REL) 検出時 RESET信号発生時 CSIE = 0 (図5 - 56参照)	コマンド信号 (CMD) 検出時

バス・リリース検出フラグ (R)

RELD	クリアされる条件 (RELD = 0)	セットされる条件 (RELD = 1)
	転送スタート命令実行時 リセット信号発生時 CSIE = 0 (図5 - 56参照) アドレス受信時にSVAとSIOが一致しなかったとき	バス・リリース信号 (REL) 検出時

コマンド・トリガ・ビット (W)

CMDT	コマンド信号 (CMD) のトリガ出力制御ビットです。セット (CMDT = 1) することによりSOラッチがクリア (0) され、その後、CMDTビットは自動的にクリア (0) されます。
------	---

注意 SB0 (または、SB1) のクリアはシリアル転送中には行わないでください。必ず転送開始前、または転送終了後に行ってください。

バス・リリース・トリガ・ビット (W)

RELT	バス・リリース信号 (REL) のトリガ出力制御ビットです。セット (RELT = 1) することによりSOラッチがセット (1) され、その後、RELTビットは自動的にクリア (0) されます。
------	--

注意 SB0 (または、SB1) のクリアはシリアル転送中には行わないでください。必ず転送開始前、または転送終了後に行ってください。

(3) シフト・レジスタ (SIO)

図5 - 58にシフト・レジスタ周辺の構成を示します。SIOはパラレル-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）を行うための8ビット・レジスタです。

シリアル転送の開始はSIOにデータを書き込むことにより開始します。

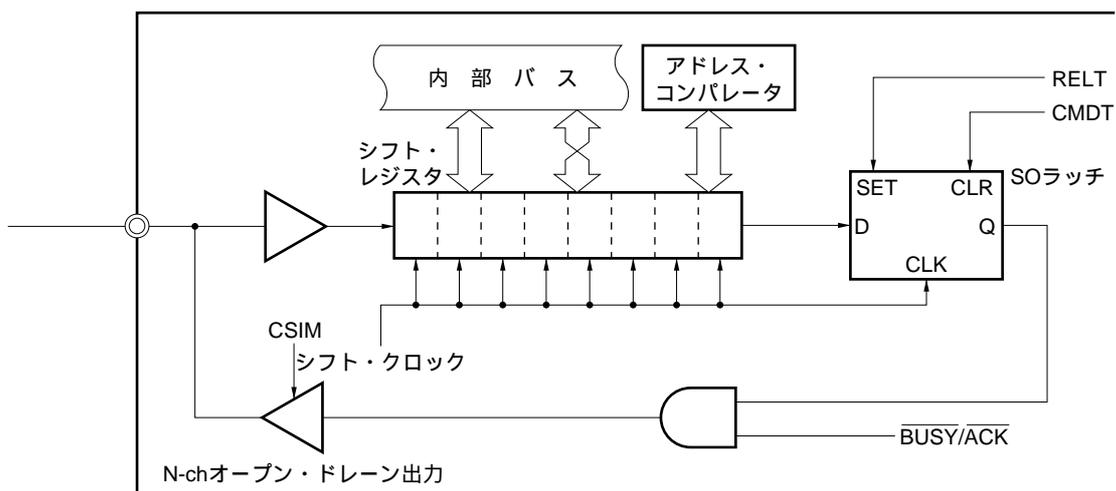
送信時はSIOに書き込まれたデータが、シリアル出力（SO）またはシリアル・データ・バス（SB0またはSB1）に出力されます。受信時はデータが、シリアル入力（SI）またはSB0またはSB1からSIOに読み込まれます。

8ビット操作命令で読み出し、書き込みが可能です。

動作中に $\overline{\text{RESET}}$ 信号を発生した場合、SIOの値は不定となります。スタンバイ・モード時に $\overline{\text{RESET}}$ 信号を発生した場合、SIOの値は保持されます。

8ビット送信 / 受信終了後、シフト動作は停止します。

図5 - 58 シフト・レジスタの周辺の構成



SIOの読み出しおよびシリアル転送スタート（書き込み）可能なタイミングは次の場合です。

- ・シリアル・インタフェース動作許可 / 禁止ビット（CSIE）= 1 のとき。ただし、シフト・レジスタにデータを書き込んだあと、CSIEを“1”に設定した場合を除く。
- ・8ビット・シリアル転送後で、シリアル・クロックがマスクされた状態のとき。
- ・ $\overline{\text{SCK}}$ がハイ・レベルのとき。

SIOへのデータの書き込み / 読み出しは、必ず $\overline{\text{SCK}}$ がハイ・レベルの状態で行ってください。

2線式シリアルI/Oモード、SBIモードのデータ・バス構成は、入力端子と出力端子が兼用になっています。出力端子はN-chオープン・ドレイン構成です。したがって、これから受信を行おうとするデバイスは、SIOにFFHを設定してください。

(4) スレーブ・アドレス・レジスタ (SVA)

SVAは、スレーブがスレーブ・アドレス値（自分の指定番号）をセットするための8ビット・レジスタです。

操作は、8ビット操作命令で行います。

$\overline{\text{RESET}}$ 信号発生により、SVAの値は不定になります。ただし、スタンバイ・モード時に $\overline{\text{RESET}}$ 信号を発生した場合は、SVAの値は保持されます。

(a) スレーブ・アドレスの検出 (SBIモード時)

μ PD753208をスレーブ・デバイスとしてシリアル・バスに接続するときを使用します。SVAはスレーブがスレーブ・アドレス値（自分の指定番号）をセットするための8ビット・レジスタです。マスタは接続されているスレーブに対して、特定のスレーブを選択するためのスレーブ・アドレスを出力します。アドレス・コンパレータによりこれらの2つのデータ（マスタから出力されたスレーブ・アドレスとSVAの値）を比較して、一致すると、そのスレーブが選択されたことになります。

このときシリアル動作モード・レジスタ (CSIM) のビット6 (COI) は“1”になります。

アドレス受信時には、一致が検出されなければ、バス・リリース検出フラグ (RELD) をクリア (0) します。WUP = 1 のときには一致が検出されたときのみ、IRQCSIをセットします。この割り込み要求により、マスタから μ PD753208に通信要求があったことを知ることができます。

(b) エラーの検出 (2線式シリアルI/Oモード, SBIモード時)

SVAは次の場合にエラーの検出を行います。

- ・ μ PD753208をマスタ・デバイスとしてアドレス、コマンド、データを送信する場合
- ・ μ PD753208をスレーブ・デバイスとしてデータを送信する場合

詳細は、5.6.6 (6) エラーの検出または5.6.7 (8) エラーの検出を参照してください。

5.6.4 動作停止モード

動作停止モードは、シリアル転送を行わないときに使用するモードです。消費電力を低減することができます。

このモードでは、シフト・レジスタはシフト動作を行わないので、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。

リセット信号入力時は、動作停止モードになります。P02/SO/SB0端子，P03/SI/SB1端子は，入力ポートに固定されます。また，P01/ $\overline{\text{SCK}}$ は，シリアル動作モード・レジスタの設定により，入力ポートとして使用することができます。

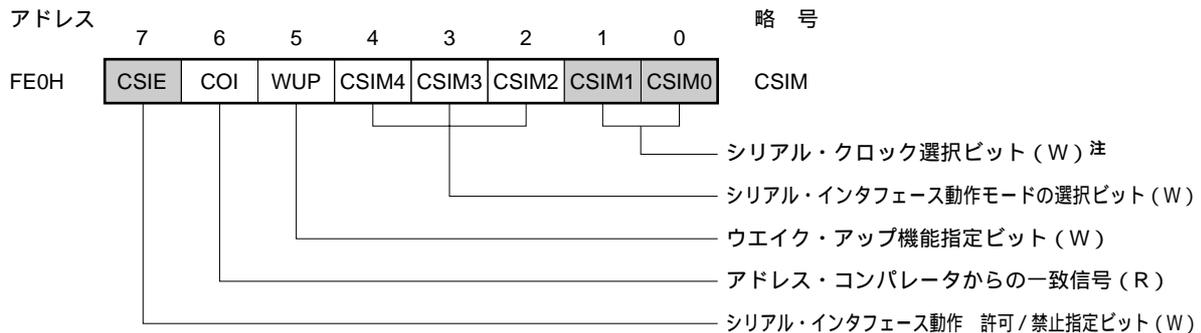
〔レジスタの設定〕

動作停止モードの設定は，シリアル動作モード・レジスタ（CSIM）で行います（CSIMのフォーマットについては，5.6.3（1）シリアル動作モード・レジスタ（CSIM）を参照）。

CSIMの操作は，8ビット操作命令により行いますが，CSIEはビット操作も可能です。また，ビットの名称を用いて操作することができます。

リセット入力によりCSIMは00Hになります。

は動作停止モードにおける使用ビットを示します。



注 P01/ $\overline{\text{SCK}}$ 端子の状態を選択可能

備考 (R) 読み出しのみ可能

(W) 書き込みのみ可能

シリアル・インタフェース動作 許可/禁止指定ビット (W)

		シフト・レジスタの動作	シリアル・クロック・カウンタ	IRQCSIフラグ	SO/SB0, SI/SB1端子
CSIE	0	シフト動作禁止	クリア	保持	ポート0機能専用

シリアル・クロック選択ビット (W)

P01/SCK端子は、CSIM0、CSIM1の設定により以下のような状態になります。

CSIM1	CSIM0	P01/SCK端子の状態
0	0	ハイ・インピーダンス
0	1	ハイ・レベル
1	0	
1	1	

シリアル転送中にCSIEをクリアするときは次の手順で行ってください。

割り込み許可フラグ (IECSI) をクリアして割り込み禁止状態にする。

CSIEをクリアする。

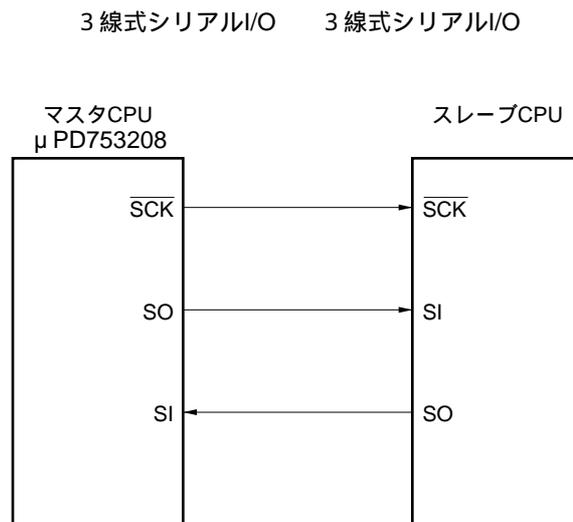
割り込み要求フラグ (IRQCSI) をクリアする。

5.6.5 3線式シリアルI/Oモードの動作

3線式シリアルI/Oモードは、他の75XLシリーズや75Xシリーズ、78Kシリーズ、各種周辺I/Oデバイスなどで用いられている方式と接続可能なモードです。

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本のラインで通信を行います。

図5 - 59 3線式シリアルI/Oのシステム構成例



備考 μPD753208をスレーブCPUとして使用することもできます。

(1) レジスタの設定

3線式シリアルI/Oモードを使用するときは、次の2つのレジスタの設定により行います。

シリアル動作モード・レジスタ (CSIM)

シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

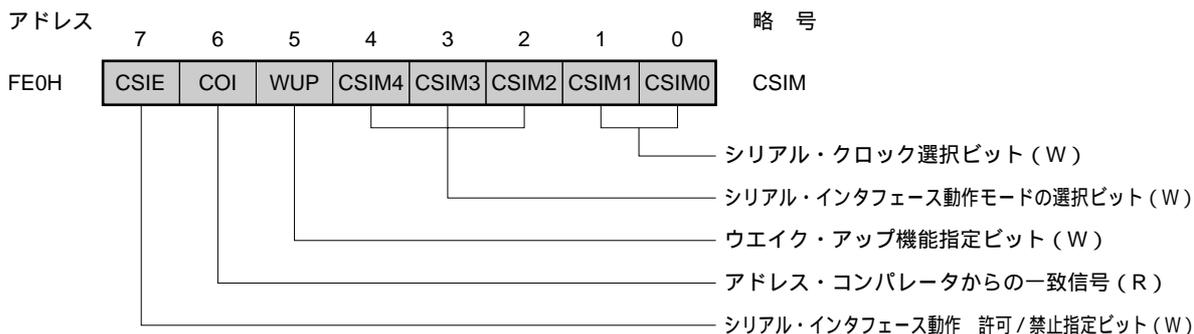
(a) シリアル動作モード・レジスタ (CSIM)

3線式シリアルI/Oモードを使用する場合は、CSIMを下の図のように設定します (CSIMのフォーマットについては5.6.3 (1) シリアル動作モード・レジスタ (CSIM)を参照)。

操作は、8ビット操作命令により行います。ビット7, 6, 5についてはビット操作も可能です。

リセット入力によりCSIMは00Hになります。

■は3線式シリアルI/Oモードにおける使用ビットを示します。



備考 (R) 読み出しのみ可能
(W) 書き込みのみ可能

シリアル・インタフェース動作 許可/禁止指定ビット (W)

	シフト・レジスタの動作	シリアル・クロック・カウンタ	IRQCSIフラグ	SO/SB0, SI/SB1端子
CSIE	1 シフト動作可能	カウンタ動作	セット可能	各モードでの機能とポート0 兼用

アドレス・コンパレータからの信号 (R)

COI ^注	クリアされる条件 (COI = 0)	セットされる条件 (COI = 1)
	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致しなかったとき	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致したとき

注 COIの読み出しは、シリアル転送開始前、完了後にのみ有効です。転送中には不定の値が読み出されます。また、8ビット操作命令によって書き込まれたCOIのデータは無視されます。

ウェイク・アップ機能指定ビット (W)

WUP	0	シリアル転送終了時に毎回IRQCSIをセットする。
-----	---	---------------------------

シリアル・インタフェース動作モードの選択ビット (W)

CSIM4	CSIM3	CSIM2	シフト・レジスタのビット順	SO端子機能	SI端子機能
x	0	0	SIO ₇₋₀ XA (MSB先頭で転送)	SO (CMOS出力)	SI (CMOS入力)
		1	SIO ₀₋₇ XA (LSB先頭で転送)		

備考 x : don't care

シリアル・クロック選択ビット (W)

CSIM1	CSIM0	シリアル・クロック	SCK端子モード
0	0	SCK端子への外部からの入力クロック	入力
0	1	タイマ/イベント・カウンタ出力 (TO)	出力
1	0	$f_x/2^4$ (262 kHz) 注	
1	1	$f_x/2^3$ (524 kHz) 注	

注 () 内は $f_x = 4.19$ MHz動作時

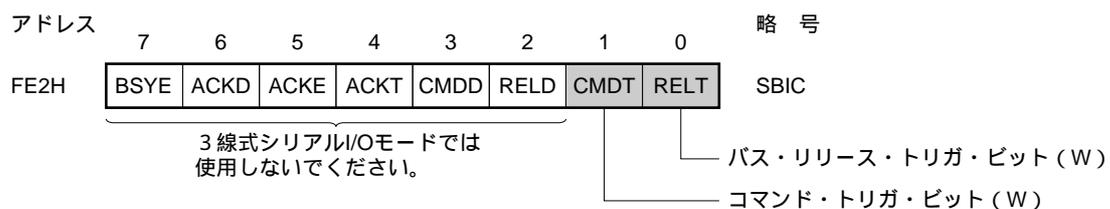
(b) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

3線式シリアル/Oモードを使用する場合は、SBICを下の図のように設定します (SBICのフォーマットについては5.6.3(2)シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)を参照)。

操作はビット操作命令により行います。

リセット入力によりSBICは00Hになります。

 は3線式シリアル/Oモードにおける使用ビットを示します。



備考 (W) 書き込みのみ可能

コマンド・トリガ・ビット

CMDT	コマンド信号 (CMD) のトリガ出力制御ビットです。セット (CMDT = 1) することによりSOラッチがクリア (0) され、その後、CMDTビットは自動的にクリア (0) されます。
------	---

バス・リリース・トリガ・ビット (W)

RELT	バス・リリース信号 (REL) のトリガ出力制御ビットです。セット (RELT = 1) することによりSOラッチがセット (1) され、その後、RELTビットは自動的にクリア (0) されます。
------	--

注意 3線式シリアル/I/OモードではCMDT, RELT以外は使用しないでください。

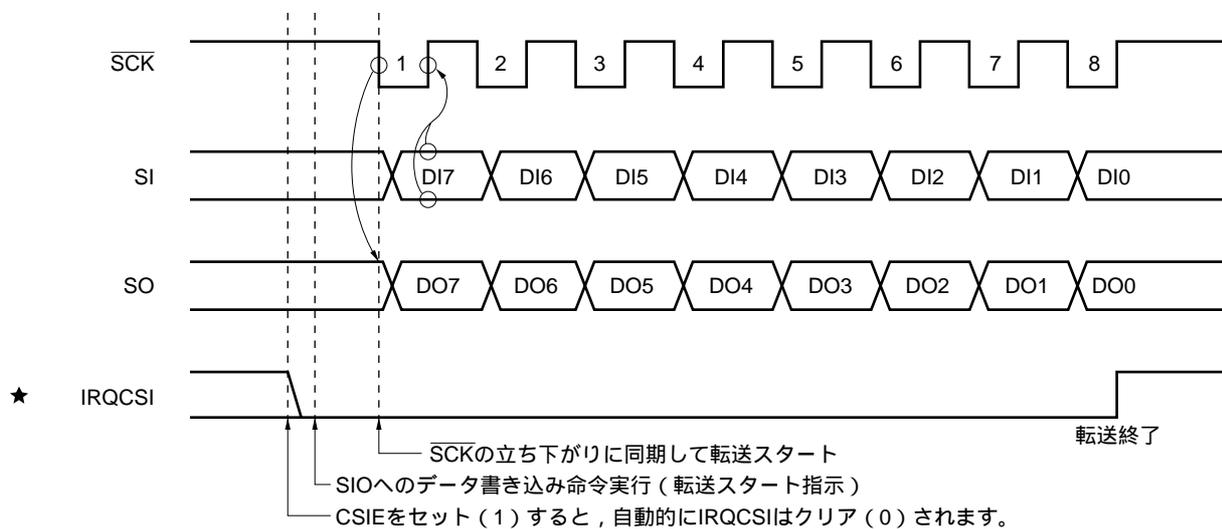
(2) 通信動作

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を行います。

シフト・レジスタのシフト動作は、シリアル・クロック (\overline{SCK}) 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ラッチに保持され、SO端子から出力されます。また、 \overline{SCK} の立ち上がりで、SI端子に入力された受信データがシフト・レジスタにラッチされます。

8ビット転送終了によりシフト・レジスタの動作は自動的に停止し、割り込み要求フラグ (IRQCSI) がセットされます。

図5 - 60 3線式シリアル/I/Oモードのタイミング



SO端子はCMOS出力となり，SOラッチの状態を出力しますので，RELTビット，CMDTビットのセットによって，SO端子出力状態を操作することができます。

ただし，シリアル転送中にはこの操作を行わないでください。

$\overline{\text{SCK}}$ 端子は，出力モード（内部システム・クロックのモード）時にはP01ラッチを操作すれば，出力状態を制御できます（5.6.8 $\overline{\text{SCK}}$ 端子出力の操作参照）。

（3）シリアル・クロックの選択

シリアル・クロックの選択は，シリアル動作モード・レジスタ（CSIM）のビット0，1の設定により行います。次の4種類のクロックを選択できます。

表5 - 9 シリアル・クロックの選択と応用（3線式シリアルI/Oモード時）

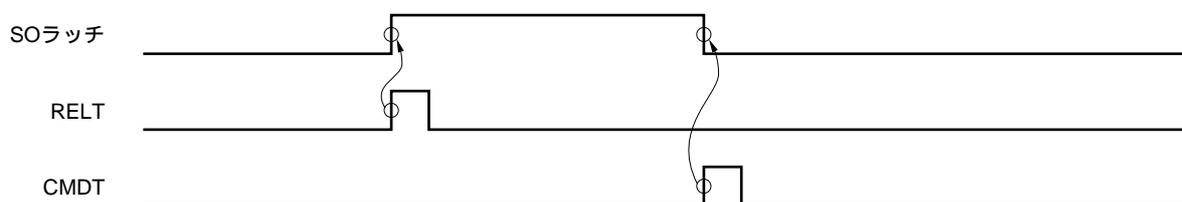
モード・レジスタ		シリアル・クロック		シフト・レジスタのR/W，およびシリアル転送スタートの可能なタイミング	応 用
CSIM 1	CSIM 0	ソース	シリアル・クロックのマスク		
0	0	外部 $\overline{\text{SCK}}$	8ビット・データ転送終了により，自動的にマスクする。	動作可能モード時（CSIE = 1） 8ビット・シリアル転送後で，シリアル・クロックがマスクされた状態のとき $\overline{\text{SCK}}$ がハイ・レベルのとき	スレーブCPU
0	1	TOUT F/F			半二重調歩同期転送 （ソフト制御）
1	0	$f_x/2^4$			中速シリアル転送
1	1	$f_x/2^3$			高速シリアル転送

★

（4）各種信号

図5 - 61にRELT，CMDTの動作を示します。

図5 - 61 RELT，CMDTの動作



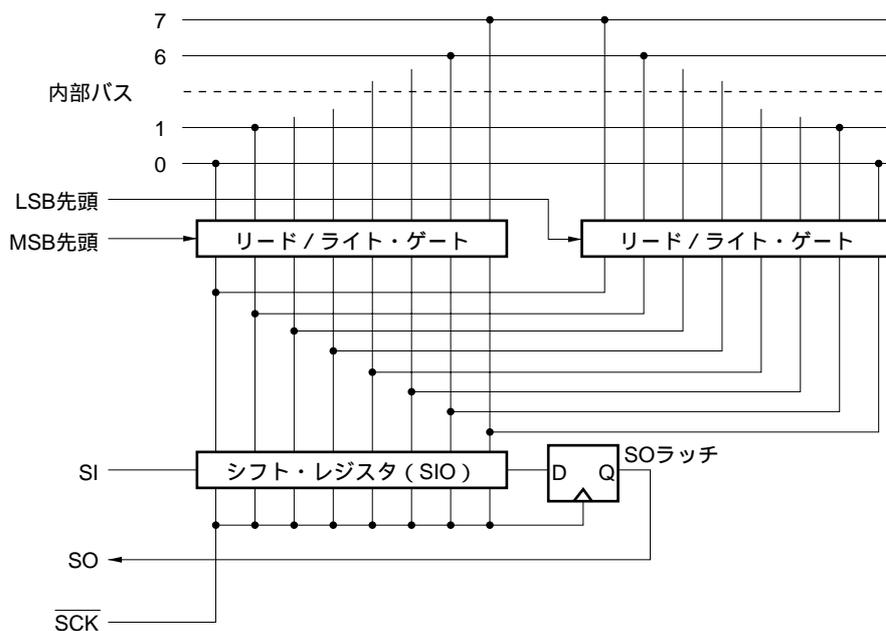
(5) MSB/LSB 先頭の切り替え

3線式シリアルI/Oモードは、転送がMSB先頭か、LSBの先頭かを選択できる機能を持っています。

図5-62にシフト・レジスタ（SIO）、および内部バスの構成を示します。図5-62に示すようにMSB/LSBを反転して読み出し/書き込みを行うことができます。

MSB/LSB先頭切り替えはシリアル動作モード・レジスタ（CSIM）のビット2により指定できます。

図5-62 転送ビットの切り替え回路



先頭ビットの切り替えは、シフト・レジスタ（SIO）へのデータ書き込みのビット順を切り替えることによって実現させています。SIOのシフト順は常に同じです。

したがって、MSB/LSBの先頭ビットの切り替えは、シフト・レジスタにデータを書き込む前に行ってください。

(6) 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、シフト・レジスタ(SIO)に転送データをセットすることで開始します。

シリアル・インタフェース動作 許可/禁止ビット(CSIE) = 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 $\overline{\text{SCK}}$ がハイ・レベルの状態

注意 シフト・レジスタにデータを書き込んだあと、CSIE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(IRQCSI)をセットします。

例 HLレジスタで指定されたRAMのデータをSIOに転送すると同時に、SIOのデータをアキュムレータに取り込み、シリアル転送をスタートする。

```
MOV  XA, @HL    ; RAMから送信データを取り出す
SEL  MB15       ; またはCLR1 MBE
XCH  XA, SIO    ; 送信データと受信データを交換し、転送をスタートする。
```

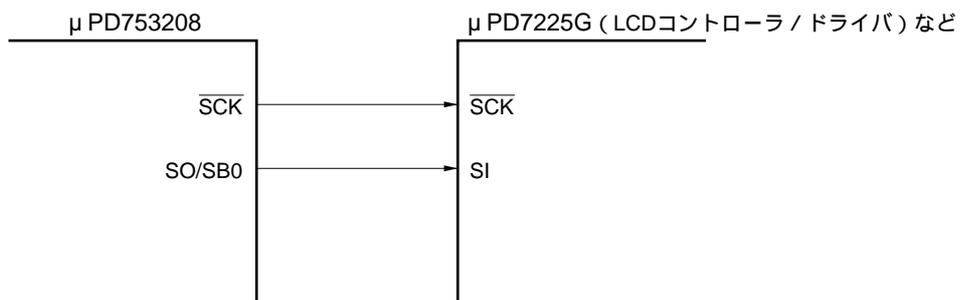
(7) 3線式シリアルI/Oモードの応用

例1 . 転送クロック262 kHz (4.19 MHz動作時) で、データをMSB先頭で転送する (マスタ動作)。

プログラム例

```
CLR1  MBE
MOV   XA, #10000010B
MOV   CSIM, XA           ; 転送モードの設定
MOV   XA, TDATA          ; TDATAは転送データの格納アドレス
MOV   SIO, XA           ; 転送データのセット, 転送スタート
```

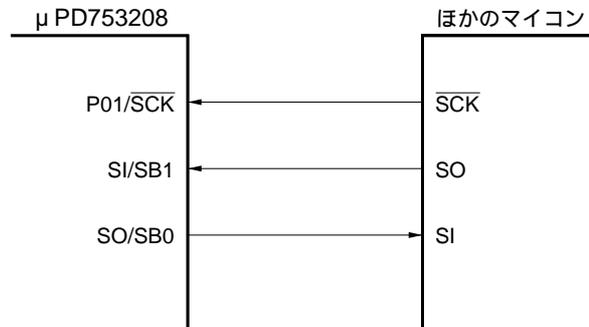
注意 2回目以降はSIOへのデータのセット (MOV SIO, XAまたはXCH XA, SIO) によって転送スタートできます。



この応用の場合、μ PD753208のSI/SB1端子は入力として使用できます。

例2．外部クロックでLSB先頭のデータを送受信する（スレーブ動作）。

（この場合は、シフト・レジスタのリード/ライトをMSBとLSBを反転して行う機能を活用します。）



プログラム例

メイン・ルーチン

```

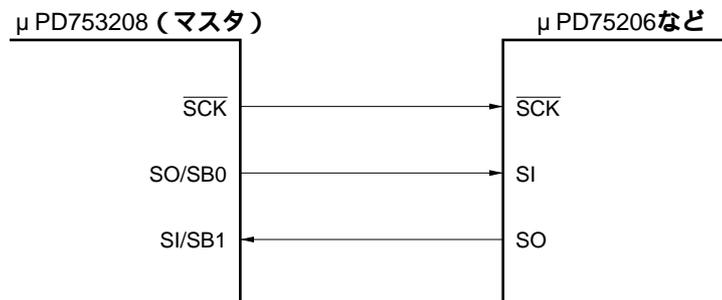
CLR1  MBE
MOV   XA, #84H
MOV   CSIM, XA    ; シリアル動作停止, MSB/LSB反転モード, 外部クロック
MOV   XA, TDATA
MOV   SIO, XA     ; 転送データ・セット, 転送スタート
EI    IECSI
EI
  
```

割り込みルーチン (MBE = 0)

```

MOV   XA, TDATA
XCH  XA, SIO     ; 受信データ 送信データ, 転送スタート
MOV  RDATA, XA  ; 受信データ・セーブ
RETI
  
```

例3 . 転送クロックで524 kHz (4.19 MHz動作時) を使用し、高速でデータを送受信する。



プログラム例 マスタ側

```

CLR1      MBE
MOV       XA, #10000011B
MOV       CSIM, XA      ; 転送モードの設定
MOV       XA, TDATA
MOV       SIO, XA      ; 転送データをセット, 転送スタート
          ⋮
          ⋮
LOOP : SKTCLR  IRQCSI      ; IRQCSIのテスト
BR        LOOP
MOV       XA, SIO      ; 受信データ取り込み
  
```

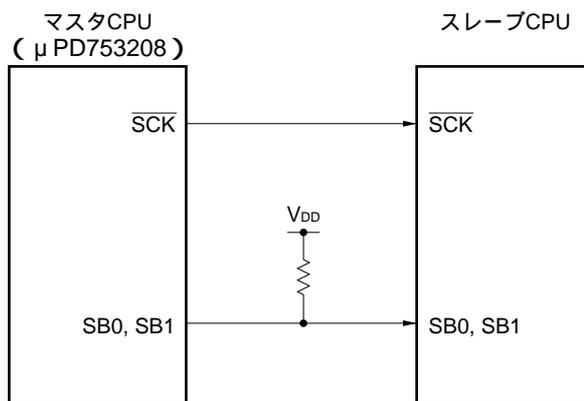
5.6.6 2線式シリアルI/Oモードの動作

2線式シリアルI/Oモードは、プログラムにより任意の通信フォーマットに対応可能なモードです。

基本的にはシリアル・クロック ($\overline{\text{SCK}}$) とシリアル・データ入力/出力 (SB0またはSB1) の2本のラインで通信を行います。

図5 - 63 2線式シリアルI/Oのシステム構成例

2線式シリアルI/O 2線式シリアルI/O



備考 μPD753208をスレーブCPUとして使用することもできます。

(1) レジスタの設定

2線式シリアルI/Oモードを使用するときは、次の2つのレジスタの設定により行います。

シリアル動作モード・レジスタ (CSIM)

シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

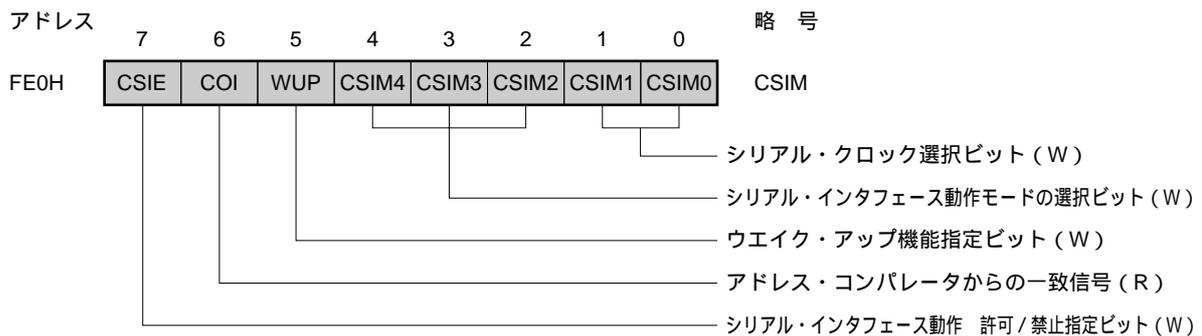
(a) シリアル動作モード・レジスタ (CSIM)

2線式シリアルI/Oモードを使用する場合は、CSIMを下の図のように設定します (CSIMのフォーマットについては5.6.3 (1) シリアル動作モード・レジスタ (CSIM) を参照)。

操作は、8ビット操作命令により行います。ビット7、6、5についてはビット操作も可能です。

リセット入力によりCSIMは00Hになります。

■は2線式シリアルI/Oモードにおける使用ビットを示します。



備考 (R) 読み出しのみ可能

(W) 書き込みのみ可能

シリアル・インタフェース動作 許可/禁止指定ビット (W)

	シフト・レジスタの動作	シリアル・クロック・カウンタ	IRQCSIフラグ	SO/SB0, SI/SB1端子	
CSIE	1	シフト動作可能	カウント動作	セット可能	各モードでの機能とポート0 兼用

アドレス・コンパレータからの信号 (R)

COI ^注	クリアされる条件 (COI = 0)	セットされる条件 (COI = 1)
	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致しなかったとき	スレーブ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致したとき

注 COIの読み出しは、シリアル転送開始前、完了後にのみ有効です。転送中には不定の値が読み出されます。

また、8ビット操作命令によって書き込まれたCOIのデータは無視されます。

ウエイク・アップ機能指定ビット (W)

WUP	0	シリアル転送終了時に毎回IRQCSIをセットする。
-----	---	---------------------------

シリアル・インタフェース動作モードの選択ビット (W)

CSIM4	CSIM3	CSIM2	シフト・レジスタのビット順	SO端子機能	SI端子機能
0	1	1	SIO ₇₋₀ XA (MSB先頭で転送)	SB0 (N-chオープン・ドレイン) 入出力	P03入力 (CMOS入力)
1				P02入力 (CMOS入力)	SB1 (N-chオープン・ドレイン) 入出力

シリアル・クロック選択ビット (W)

CSIM1	CSIM0	シリアル・クロック	SCK端子モード
0	0	SCK端子への外部からの入力クロック	入力
0	1	タイマ/イベント・カウンタ出力 (TO)	出力
1	0	$f_x/2^6$ (65.5 kHz) 注	
1	1		

注 () 内は $f_x = 4.19$ MHz動作時

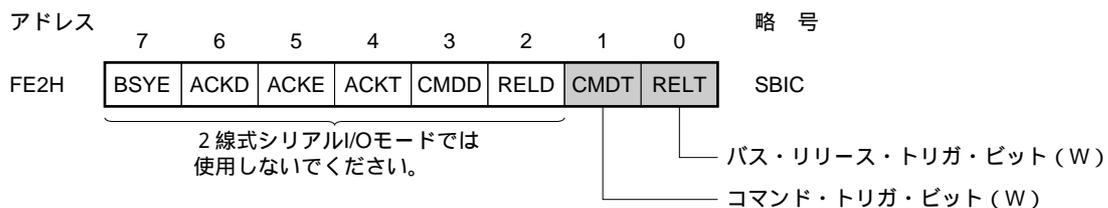
(b) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

2線式シリアル/I/Oモードを使用する場合は、SBICを下の図のように設定します (SBICのフォーマットについては、5.6.3 (2) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) を参照)。

操作はビット操作命令により行います。

リセット入力によりSBICは00Hになります。

 は2線式シリアル/I/Oモードにおける使用ビットを示します。



備考 (W) 書き込みのみ可能

コマンド・トリガ・ビット

CMDT	コマンド信号 (CMD) のトリガ出力制御ビットです。セット (CMDT = 1) することによりSOラッチがクリア (0) され、その後、CMDTビットは自動的にクリア (0) されます。
------	---

バス・リリース・トリガ・ビット (W)

RELT	バス・リリース信号 (REL) のトリガ出力制御ビットです。セット (RELT = 1) することによりSOラッチがセット (1) され、その後、RELTビットは自動的にクリア (0) されます。
------	--

注意 2線式シリアル/I/OモードではCMDT, RELT以外は使用しないでください。

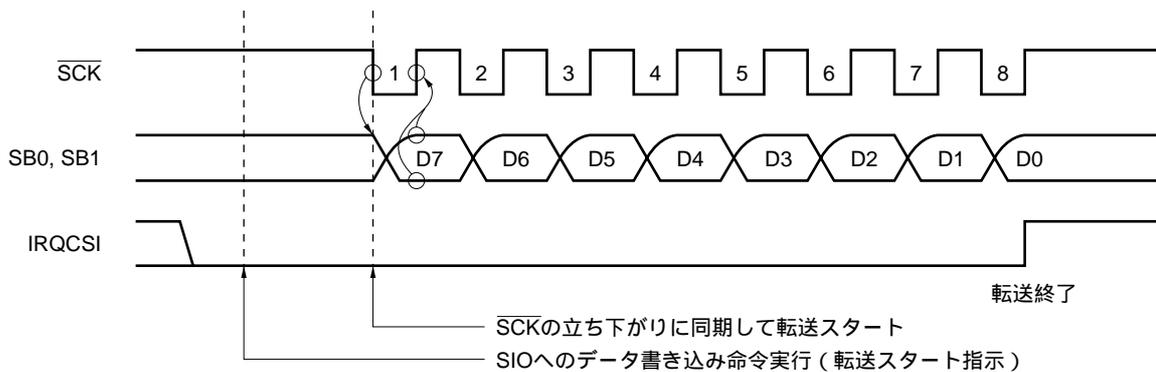
(2) 通信動作

2線式シリアル/I/Oモードでは、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を行います。

シフト・レジスタのシフト動作は、シリアル・クロック (\overline{SCK}) 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ラッチに保持され、SB0/P02 (またはSB1/P03) 端子からMSBを先頭にして出力されます。また、 \overline{SCK} の立ち上がりで、SB0端子 (またはSB1) から入力された受信データがシフト・レジスタにラッチされます。

8ビット転送終了によりシフト・レジスタの動作は自動的に停止し、割り込み要求フラグ (IRQCSI) がセットされます。

図5 - 64 2線式シリアル/I/Oモードのタイミング



SB0 (またはSB1) 端子のシリアル・データ・バスに指定された端子はN-chオープン・ドレイン入出力となりますので、外部でプルアップする必要があります。また、データの受信時にはN-chトランジスタをオフさせる必要があるため、SIOにはあらかじめFFHを書き込んでください。

SB0 (またはSB1) 端子はSOラッチの状態を出力しますので、RELTビット, CMDTビットのセットによって、SB0 (またはSB1) 端子の出力状態を操作することができます。

ただし、シリアル転送中にはこの操作を行わないでください。

\overline{SCK} 端子は、出力モード (内部システム・クロックのモード) 時にはP01出力ラッチを操作すれば、出力状態を制御できます (5.6.8 \overline{SCK} 端子出力の操作参照)。

(3) シリアル・クロックの選択

シリアル・クロックの選択は、シリアル動作モード・レジスタ (CSIM) のビット0, 1 の設定により行います。次の3種類のクロックを選択できます。

表5 - 10 シリアル・クロックの選択と応用 (2線式シリアル/I/Oモード時)

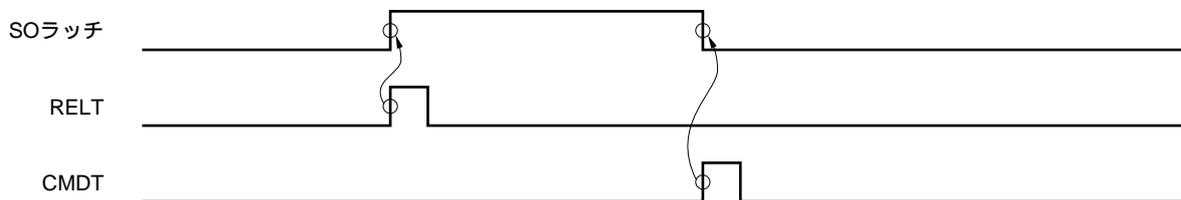
モード・レジスタ		シリアル・クロック		シフト・レジスタのR/W, およびシリアル転送 スタートの可能なタイミング	応 用
CSIM 1	CSIM 0	ソース	シリアル・ク ロックのマスク		
0	0	外部 $\overline{\text{SCK}}$	8ビット・ データ転送	動作可能モード時 (CSIE = 1) 8ビット・シリアル転送後で、シリアル・ク ロックがマスクされた状態のとき $\overline{\text{SCK}}$ がハイ・レベルのとき	スレーブCPU
0	1	TOUT F/F	終了により、 自動的にマ スクする		任意速シリアル転送
1	0	fx/2 ⁶	スクする		低速シリアル転送
1	1				

★

(4) 各種信号

図5 - 65にRELT, CMDTの動作を示します。

図5 - 65 RELT, CMDTの動作



(5) 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、シフト・レジスタ(SIO)に転送データをセットすることで開始します。

シリアル・インタフェース動作 許可/禁止ビット(CSIE) = 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 $\overline{\text{SCK}}$ がハイ・レベルの状態

注意1．シフト・レジスタにデータを書き込んだあと、CSIEを“1”にしても、転送はスタートしません。

2．データ受信時にはN-chトランジスタをオフする必要がありますので、SIOにはあらかじめFFHを書き込んでおいてください。

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(IRQCSI)をセットします。

(6) エラーの検出

2線式シリアルI/Oモードでは、送信中のシリアル・バスSB0またはSB1の状態が送信しているデバイスのシフト・レジスタSIOにも取り込まれるため、次の方法によって送信エラーの検出をすることができます。

(a) 送信開始前と送信終了後のSIOのデータを比較する方法

この場合、2つのデータが異なっていれば送信エラーが発生したと判断します。

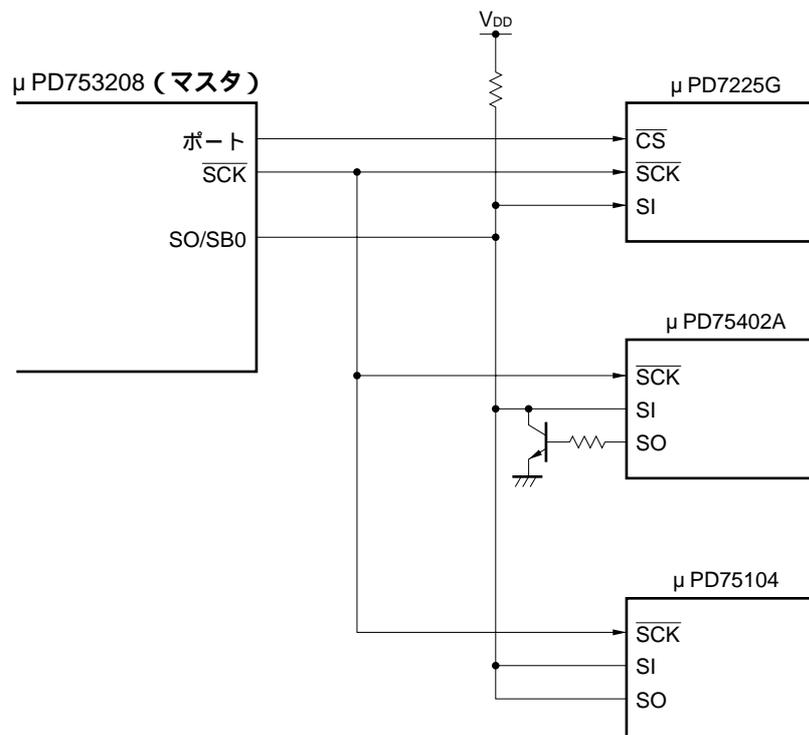
(b) スレーブ・アドレス・レジスタ(SVA)を利用する方法

送信データをSIOとSVAにもセットし、送信を行います。送信終了後に、シリアル動作モード・レジスタ(CSIM)のCOIビット(アドレス・コンパレータからの一致信号)をテストし、“1”ならば正常な送信、“0”ならば送信エラーと判断します。

(7) 2線式シリアルI/Oモードの応用

シリアル・バスを構成し、複数のデバイスを接続する。

例 μ PD753208をマスタとし、スレーブに μ PD75104、 μ PD75402A、 μ PD7225Gを接続してシステムを構成する。



μ PD75104はSI端子とSO端子を接続し、シリアル・データを出力するとき以外は、シリアル動作モード・レジスタを操作し、出力バッファをオフしてバスを開放します。

μ PD75402AはSO端子をハイ・インピーダンスにできないため、図のようにトランジスタを接続し、オープン・コレクタ出力となるようにします。そしてデータを入力するタイミングでは、あらかじめソフト・レジスタに00Hを書き込んでトランジスタをオフさせます。

各マイコンがいつデータを出力するかは、あらかじめ、取り決めしておきます。

シリアル・クロックはマスタ・マイコンである μ PD753208が出力し、ほかのスレーブ・マイコンはいずれも外部クロックで動作します。

5.6.7 SBIモードの動作

SBI (シリアル・バス・インタフェース) は、NECシリアル・バス・フォーマット準拠の高速シリアル・インタフェース方式です。

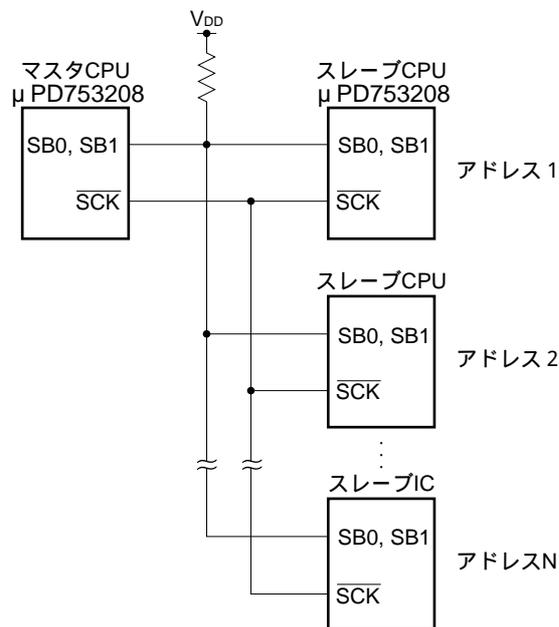
SBIは、シングル・マスタの高速シリアル・バスで、2本の信号線で複数のデバイスとの通信を行うことができるように、クロック同期式のシリアルI/O方式に、バス構成のための機能が追加されたフォーマットになっています。そのため複数のマイコンや周辺ICでシリアル・バスを構成する場合に、使用するポート数や基板上的配線数を削減することができます。

また、マスタはスレーブに対してシリアル・データ・バス上に、シリアル通信の対象デバイス選択のための“アドレス”、対象デバイスに対して指令を与える“コマンド”、および実際の“データ”を出力できます。スレーブは、受信したデータをハードウェアにより、“アドレス”、“コマンド”、“データ”に判別できます。この機能により、応用プログラムのシリアル・インタフェースの制御部分を簡単にできます。

SBI機能は、75XLシリーズ、75Xシリーズや、8、16ビット・シングルチップ・マイクロコンピュータの78Kシリーズなどの数種のデバイスに内蔵されています。

SBIに準拠するシリアル・インタフェースを持つCPUや、周辺ICを使用した場合のシリアル・バス構成例を図5-66に示します。

図5-66 SBIのシステム構成例



注意1 . SBIでは、シリアル・データ・バス端子SB0 (またはSB1) が、オープン・ドレイン出力になっているため、シリアル・データ・バス・ラインはワイアード・オア状態になります。また、シリアル・データ・バス・ラインには、プルアップ抵抗が必要です。

2 . マスタ/スレーブの交換処理を行う場合は、シリアル・クロック・ライン ($\overline{\text{SCK}}$) の入力/出力の切り替えがマスタ、スレーブ間で非同期に行われるため、 $\overline{\text{SCK}}$ にもプルアップ抵抗が必要となります。

(1) SBIの機能

従来のシリアルI/O方式では、データ転送機能しか持っていないために、複数のデバイスを接続してシリアル・バスを構成した場合に、チップ・セレクト信号やコマンド/データの区別、ビジー状態の判断などのため多くのポートや配線が必要となります。また、これらの制御をソフトウェアで行おうとすると、ソフトウェアの負担が大きくなってしまいます。

SBIでは、シリアル・クロック $\overline{\text{SCK}}$ と、シリアル・データ・バスSB0またはSB1の2本の信号線でシリアル・バスを構成することができます。そのため、マイコンのポート数の削減や、基板内の配線や引き回しの減少に有効となります。

SBIの機能について次に示します。

(a) アドレス/コマンド/データの判断機能

シリアル・データを、アドレス、コマンド、およびデータの3種類に区別します。

(b) アドレスによるチップ・セレクト機能

マスタは、アドレスの送信により、チップ・セレクト（選択）を行います。

(c) ウェイク・アップ機能

スレーブは、ウェイク・アップ機能（ソフトウェアで設定/解除が可能）により、アドレス受信の判断（チップ・セレクトの判断）を容易に行うことができます。

ウェイク・アップ機能を設定した場合、一致アドレス受信時に割り込み（IRQCSI）が発生します。そのため、複数のデバイスと通信を行う場合も、選択されたスレーブ以外のCPUはシリアル通信に関係なく動作することができます。

(d) アクノリッジ信号 ($\overline{\text{ACK}}$) 制御機能

シリアル・データの受信確認のための、アクノリッジ信号を制御します。

(e) ビジー信号 ($\overline{\text{BUSY}}$) 制御機能

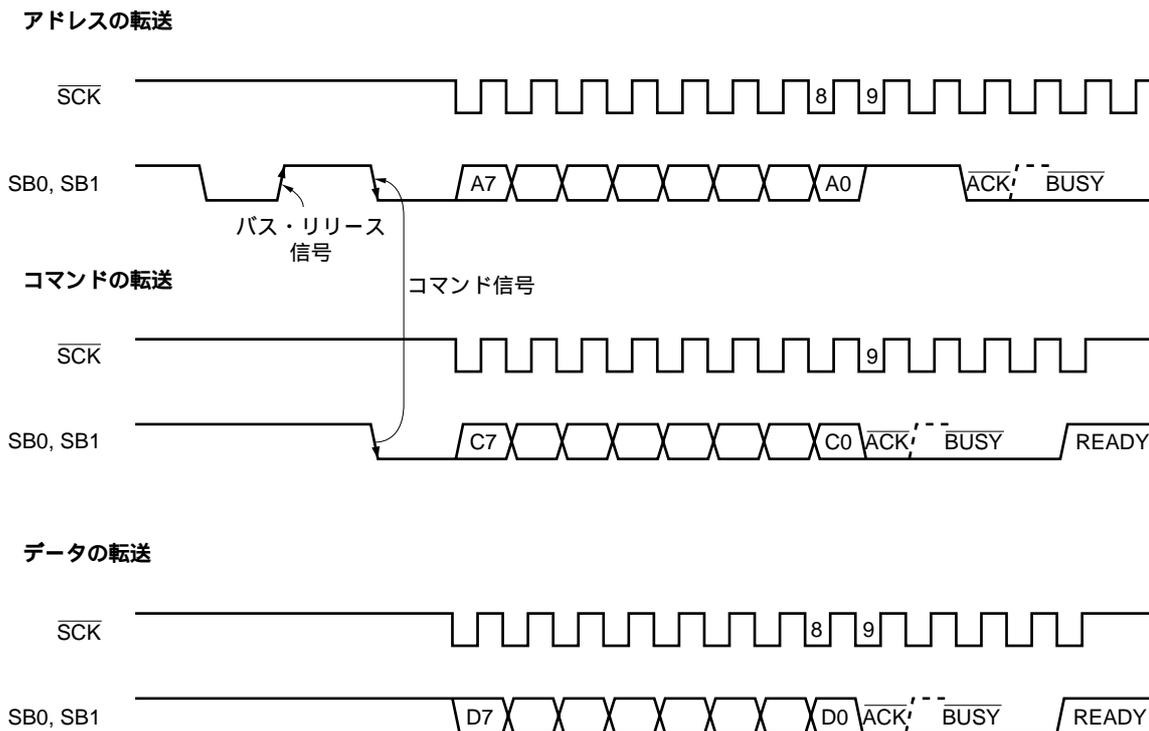
スレーブのビジー状態を知らせるための、ビジー信号を制御します。

(2) SBIの定義

SBIのシリアル・データのフォーマットおよび、使用する信号の意味について説明します。

SBIで転送されるシリアル・データは「アドレス」、「コマンド」、「データ」の3種類に区別されます。

図5 - 67 SBI転送のタイミング



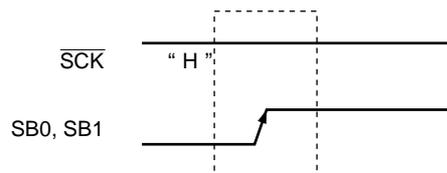
バス・リリース信号およびコマンド信号はマスタが出力します。また、 \overline{BUSY} はスレーブが出力します。 \overline{ACK} はマスタ、スレーブのどちらでも出力できます（通常、8ビット・データの受信側が出力します）。

シリアル・クロックは、8ビット・データの転送開始から、 \overline{BUSY} が解除されるまで、マスタが出力し続けます。

(a) バス・リリース信号 (REL)

バス・リリース信号は、 $\overline{\text{SCK}}$ ラインがハイ・レベルのとき（シリアル・クロックが出力されていない場合）に、SB0またはSB1ラインがロウ・レベルからハイ・レベルに変化した信号です。この信号は、マスタが出力します。

図5 - 68 バス・リリース信号



バス・リリース信号は、これからマスタがスレーブに対してアドレスを送信することを示すものです。スレーブは、バス・リリース信号を検出するハードウェアを内蔵しています。

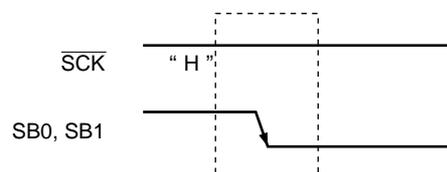
★

注意 $\overline{\text{SCK}}$ ラインがハイ・レベルのときに、SB0またはSB1ラインがロウ・レベルからハイ・レベルに変化すると、バス・リリース信号と認識されます。したがって、基板容量などの影響でバスの変化タイミングにずれが生じると、データを送信しているにもかかわらず、バス・リリース信号と判断されてしまうことがあります。配線の引き回しには十分注意してください。

(b) コマンド信号 (CMD)

コマンド信号は、 $\overline{\text{SCK}}$ ラインが、ハイ・レベルのとき（シリアル・クロックが出力されていない場合）に、SB0またはSB1ラインがハイ・レベルからロウ・レベルに変化した信号です。この信号は、マスタが出力します。

図5 - 69 コマンド信号



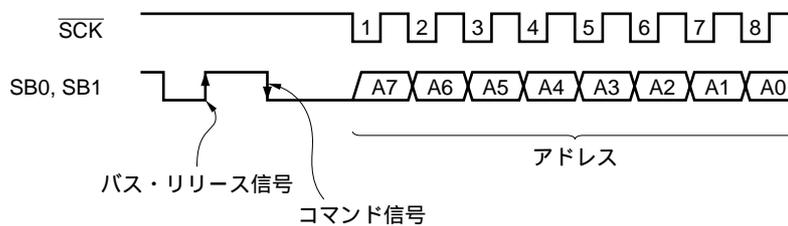
スレーブは、コマンド信号を検出するハードウェアを内蔵しています。

- ★ **注意** $\overline{\text{SCK}}$ ラインがハイ・レベルのときに、SB0またはSB1ラインがハイ・レベルからロウ・レベルに変化すると、コマンド信号と認識されます。したがって、基板容量などの影響でバスの変化タイミングにずれが生じると、データを送信しているにもかかわらず、コマンド信号と判断されてしまうことがあります。配線の引き直しには十分注意してください。

(c) アドレス

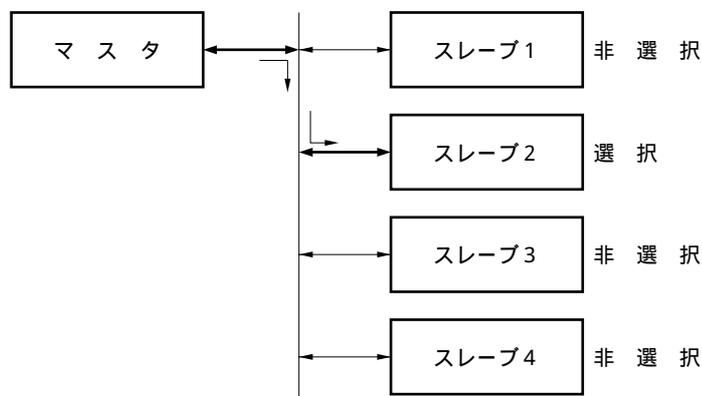
アドレスは、マスタがバス・ラインに接続されているスレーブに対して、特定のスレーブを選択するために出力する8ビット・データです。

図5 - 70 アドレス



バス・リリース信号、コマンド信号に続く8ビット・データはアドレスと定義されています。スレーブでは、ハードウェアでこの条件を検出し、8ビット・データが自分の指定番号（スレーブ・アドレス）と一致しているかをハードウェアでチェックします。このとき、8ビット・データと、スレーブ・アドレスが一致すると、そのスレーブが選択されたことになり、以後、マスタから切り離し指示があるまで、マスタとの通信を行います。

図5 - 71 アドレスによるスレーブの選択



(d) コマンド, データ

アドレスの送信により選択したスレーブに対して、マスタはコマンドの送信や、データの送受信を行います。

図5 - 72 コマンド

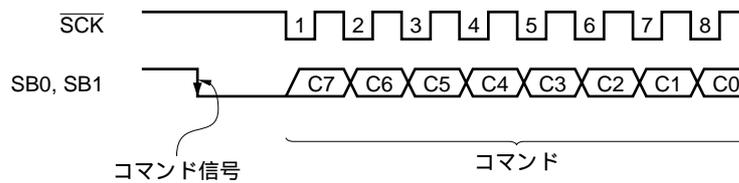
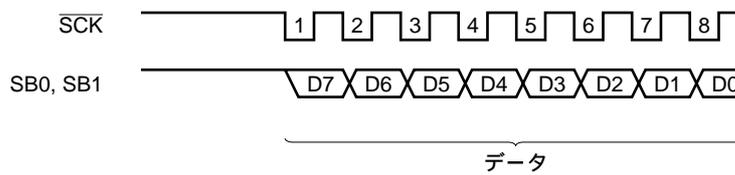


図5 - 73 データ



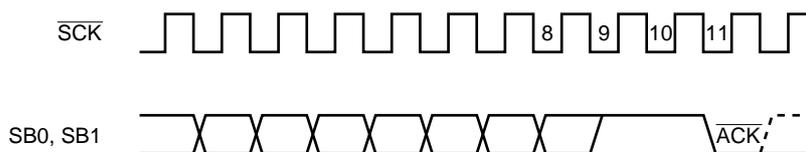
コマンド信号の次の8ビット・データはコマンドと定義されています。コマンド信号なしの8ビット・データはデータと定義されています。コマンド、データの使用方法は、通信の仕様によって任意に決定することができます。

(e) アクノリッジ信号 ($\overline{\text{ACK}}$)

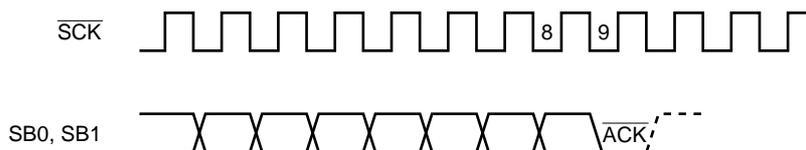
アクノリッジ信号は、送信側と受信側の間における、シリアル・データ受信の確認のための信号です。

図5 - 74 アクノリッジ信号

〔11クロック目の $\overline{\text{SCK}}$ に同期して出力した場合〕



〔9クロック目の $\overline{\text{SCK}}$ に同期して出力した場合〕



アクノリッジ信号は、8ビット・データ転送後の $\overline{\text{SCK}}$ の立ち下がりに同期したワンショット・パルスで、その位置は任意で何クロック目の $\overline{\text{SCK}}$ に同期させてもかまいません。

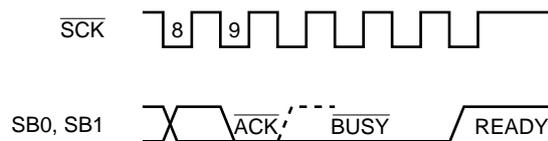
送信側は、8ビット・データ送信後、受信側がアクノリッジ信号を返したかをチェックします。データ送信後、一定時間、アクノリッジ信号が返らない場合は、受信が正しく行われなかったものと判断することができます。

(f) ビジィ信号 ($\overline{\text{BUSY}}$) , レディ信号 (READY)

ビジィ信号は、スレーブがデータの送受信のための準備中であることをマスタに知らせるための信号です。

レディ信号は、スレーブがデータの送受信が可能であることをマスタに知らせるための信号です。

図5 - 75 ビジィ信号, レディ信号



SBIでは、スレーブが、SB0 (またはSB1) ラインをロウ・レベルにすることにより、マスタにビジィ状態を知らせます。

ビジィ信号は、マスタ、またはスレーブの出力したアクノリッジ信号にひき続いて出力させます。ビジィ信号は、 $\overline{\text{SCK}}$ の立ち下がりに同期して、設定/解除を行います。マスタは、ビジィ信号が解除されると自動的にシリアル・クロック $\overline{\text{SCK}}$ の出力を終了します。

マスタは、ビジィ信号が解除され、レディ信号の状態になると次の転送を開始することができます。

(3) レジスタの設定

SBIモードを使用するときは、次の2つのレジスタの設定により行います。

シリアル動作モード・レジスタ (CSIM)

シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

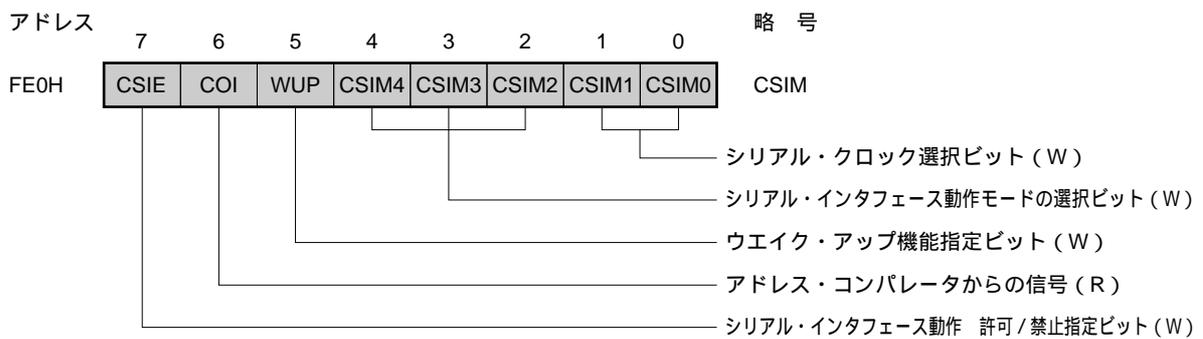
(a) シリアル動作モード・レジスタ (CSIM)

SBIモードを使用する場合は、CSIMを下の図のように設定します (CSIMのフォーマットについては、5.6.3 (1) シリアル動作モード・レジスタ (CSIM) を参照)。

操作は、8ビット操作命令により行います。ビット7, 6, 5についてはビット操作も可能です。

リセット入力によりCSIMは00Hになります。

■はSBIモードにおける使用ビットを示します。



備考 (R) 読み出しのみ可能

(W) 書き込みのみ可能

シリアル・インタフェース動作 許可/禁止指定ビット (W)

	シフト・レジスタの動作	シリアル・クロック・カウンタ	IRQCSIフラグ	SO/SB0, SI/SB1端子
CSIE	1 シフト動作範囲	カウンタ動作	セット可能	各モードでの機能とポート0 兼用

アドレス・コンパレータからの信号 (R)

COI ^注	クリアされる条件 (COI = 0)	セットされる条件 (COI = 1)
	スレープ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致しなかったとき	スレープ・アドレス・レジスタ (SVA) とシフト・レジスタのデータが一致したとき

注 COIの読み出しはシリアル転送開始前、完了後にのみ有効です。転送中には不定の値が読み出されます。

また、8ビット操作命令によって書き込まれたCOIのデータは無視されます。

ウェイク・アップ機能指定ビット (W)

WUP	0	SBIモードのマスク状態でのシリアル転送終了時に毎回IRQCSIをセットする。
	1	SBIモードのスレープ時のみに使用。バス・リリース後に受信したアドレスがスレープ・アドレス・レジスタのデータと一致したとき (ウェイク・アップ状態) のみIRQCSIをセットする。SB0またはSB1はハイ・インピーダンス。

注意 $\overline{\text{BUSY}}$ 信号出力中にWUP = 1にすると、 $\overline{\text{BUSY}}$ が解除されません。SBIでは、 $\overline{\text{BUSY}}$ の解除指示後、次のシリアル・クロック (SCK) の立ち上がりまで $\overline{\text{BUSY}}$ 信号が出力されます。WUP = 1とするときは、必ず $\overline{\text{BUSY}}$ 信号を解除したのちに、SB0 (またはSB1) 端子がハイ・レベルになったことを確認してから、WUP = 1としてください。

シリアル・インタフェース動作モードの選択ビット (W)

CSIM4	CSIM3	CSIM2	シフト・レジスタのビット順	SO端子機能	SI端子機能
0	1	0	SIO ₇₋₀ XA (MSB先頭で転送)	SB0 (N-chオープン・ドレイン) 入出力	P03 (CMOS入力)
1				P02 (CMOS入力)	SB1 (N-chオープン・ドレイン) 入出力

シリアル・クロック選択ビット (W)

CSIM1	CSIM0	シリアル・クロック	$\overline{\text{SCK}}$ 端子モード
0	0	$\overline{\text{SCK}}$ 端子への外部からの入力クロック	入力
0	1	タイマ/イベント・カウンタ出力 (TO)	出力
1	0	$f_x/2^4$ (262 kHz) 注	
1	1	$f_x/2^3$ (524 kHz) 注	

注 ()内は $f_x = 4.19$ MHz動作時

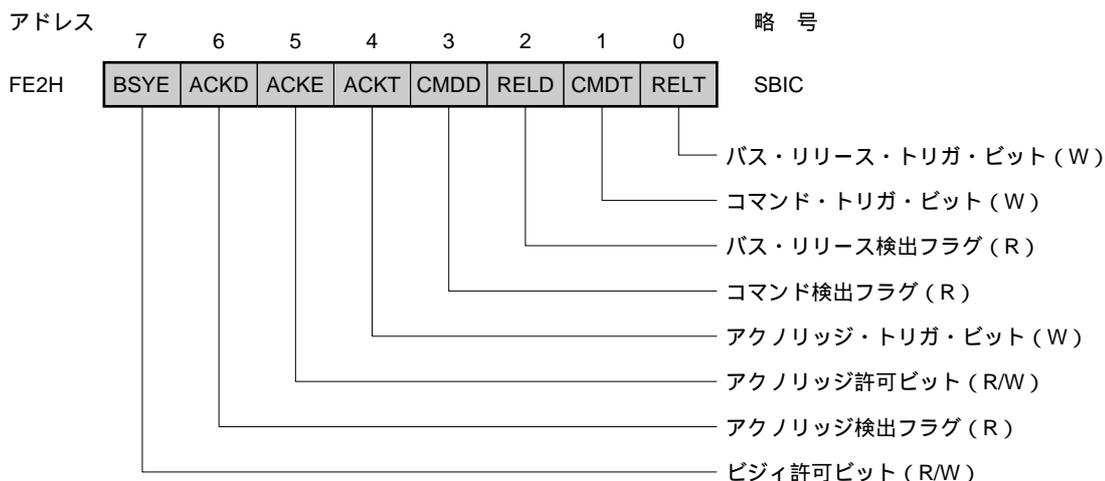
(b) シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)

SBIモードを使用する場合は、SBICを下の図のように設定します (SBICのフォーマットについては5.6.3(2)シリアル・バス・インタフェース・コントロール・レジスタ (SBIC)を参照)。

操作は、ビット操作命令により行います。

リセット入力によりSBICは00Hになります。

■はSBIモードにおける使用ビットを示します。



- 備考 (R) 読み出しのみ可能
 (W) 書き込みのみ可能
 (R/W) 読み出し / 書き込みともに可能

ビジー許可ビット (R/W)

BSYE	0	ビジー信号の自動出力禁止 クリア命令実行直後の、 \overline{SCK} の立ち下がりに同期して、ビジー信号の出力を停止
	1	アクノリッジ信号に続き \overline{SCK} の立ち下がりに同期して、ビジー信号を出力

アクノリッジ検出フラグ (R)

ACKD	クリアされる条件 (ACKD = 0)	セットされる条件 (ACKD = 1)
	転送スタート時 リセット入力時	アクノリッジ信号 (\overline{ACK}) 検出時 (\overline{SCK} の立ち上がりに同期します)

アクノリッジ許可ビット (R/W)

ACKE	0	アクノリッジ信号の自動出力を禁止します (ACKTによる出力は可能)	
	1	転送終了前にセットした場合	\overline{SCK} の9クロック目に同期して \overline{ACK} を出力します。
転送終了後にセットした場合		セット命令実行直後の \overline{SCK} に同期して \overline{ACK} を出力します。	

ACKトリガ・ビット (W)

ACKT	転送終了後にセットすると、次のSCKに同期してACKが出力されます。ACK信号出力後、自動的にクリア (0) されます。
------	--

注意 1 . シリアル転送終了前、および転送中にセット (1) しないでください。

2 . ACKTは、ソフトウェアでクリアすることはできません。

3 . ACKTをセットするときは、ACKE = 0 にしてください。

コマンド検出フラグ (R)

CMDD	クリアされる条件 (CMDD = 0)	セットされる条件 (CMDD = 1)
	転送スタート命令実行時 バス・リリース信号 (REL) 検出時 リセット入力時 CSIE = 0 (図5 - 56参照)	コマンド信号 (CMD) 検出時

バス・リリース検出フラグ (R)

RELD	クリアされる条件 (RELD = 0)	セットされる条件 (RELD = 1)
	転送スタート命令実行時 リセット入力時 CSIE = 0 (図5 - 56参照) アドレス受信時にSVAとSIOが一致しなかったとき	バス・リリース信号 (REL) 検出時

コマンド・トリガ・ビット (W)

CMDT	コマンド信号 (CMD) のトリガ出力制御ビットです。セット (CMDT=1) することによりSOラッチがクリア (0) され、その後、CMDTビットは自動的にクリア (0) されます。
------	---

注意 SB0 (またはSB1) のセットはシリアル転送中に行わないでください。必ず転送開始前、または転送終了後に行ってください。

バス・リリース・トリガ・ビット (W)

RELT	バス・リリース信号 (REL) のトリガ出力制御ビットです。セット (RELT = 1) することによりSOラッチがセット (1) され、その後、RELTビットは自動的にクリア (0) されます。
------	--

注意 SB0 (またはSB1) のセットはシリアル転送中に行わないでください。必ず転送開始前、または転送終了後に行ってください。

(4) シリアル・クロックの選択

シリアル・クロックの選択は、シリアル動作モード・レジスタ (CSIM) のビット0, 1の設定により行います。次の4種類のクロックを選択できます。

表5 - 11 シリアル・クロックの選択と応用 (SBIモード時)

モード・レジスタ		シリアル・クロック		シフト・レジスタのR/W, およびシリアル転送 スタートの可能なタイミング	応 用
CSIM 1	CSIM 0	ソース	シリアル・クロック のマスク		
0	0	外部 $\overline{\text{SCK}}$	8ビット・データ転送終了により, 自動的にマスクする。	動作可能モード時 (CSIE = 1) 8ビット・シリアル転送後で, シリアル・クロックがマスクされた状態のとき $\overline{\text{SCK}}$ がハイ・レベルのとき	スレーブCPU
0	1	TOUT F/F			任意速シリアル転送
1	0	$f_x/2^4$			中速シリアル転送
1	1	$f_x/2^3$			高速シリアル転送

内部システム・クロックを選択した場合, $\overline{\text{SCK}}$ は, 内部に対して8発で停止しますが, 外部に対しては, スレーブがレディ状態になるまでカウントします。

(5) 各種信号

SBIにおける各種信号と、SBIC上のフラグの動作について図5-76から図5-81に示します。また、SBIの各種信号一覧を表5-12に示します。

図5-76 RELT, CMDT, RELD, CMDD (マスタ) の動作

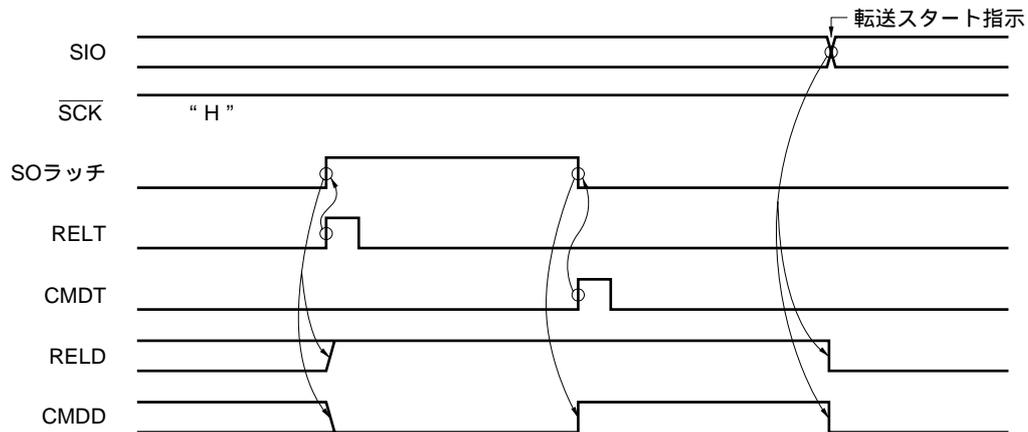


図5-77 RELT, CMDT, RELD, CMDD (スレーブ) の動作

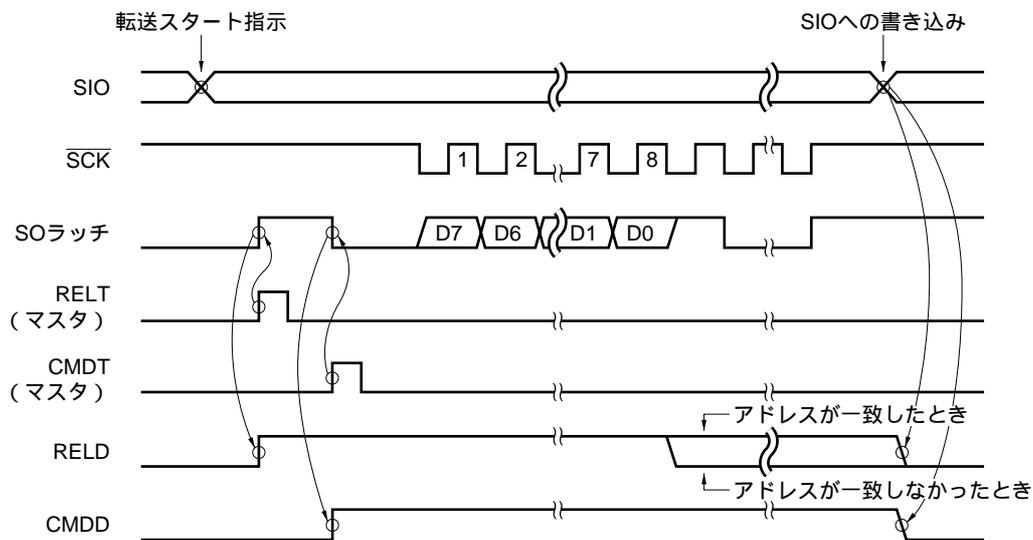
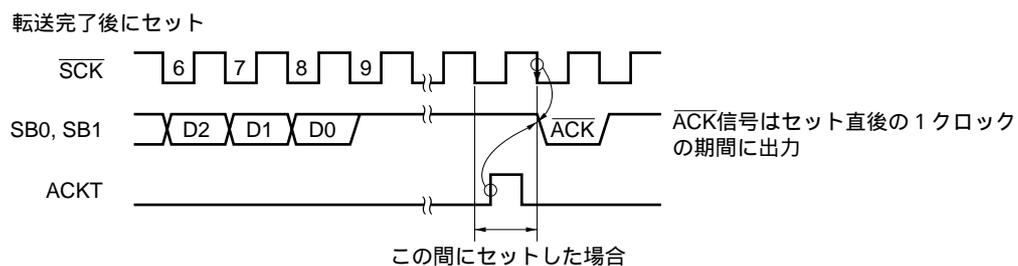


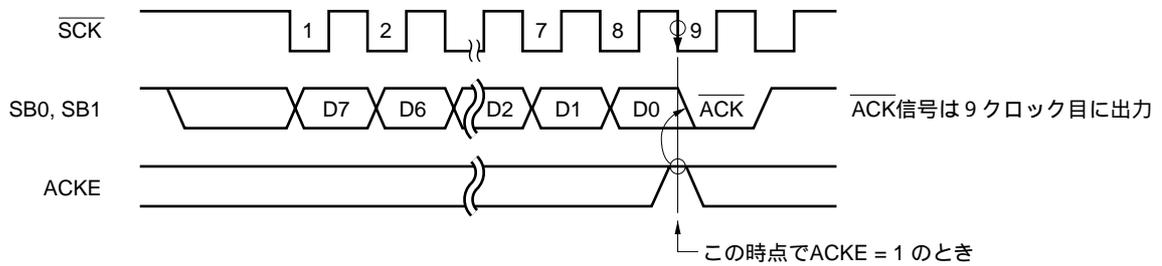
図5-78 ACKTの動作



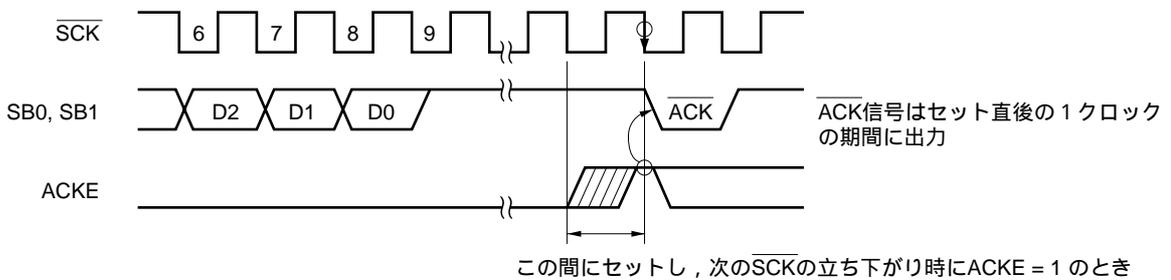
注意 ACKTは転送終了前にはセットしないでください。

図5-79 ACKEの動作

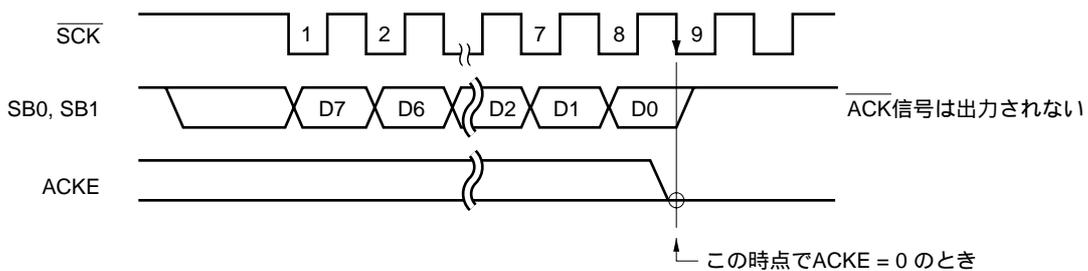
(a) 転送完了時にACK = 1 の場合



(b) 転送完了後にセットした場合



(c) 転送完了時にACK = 0 の場合



(d) ACKE = 1 の期間が短い場合

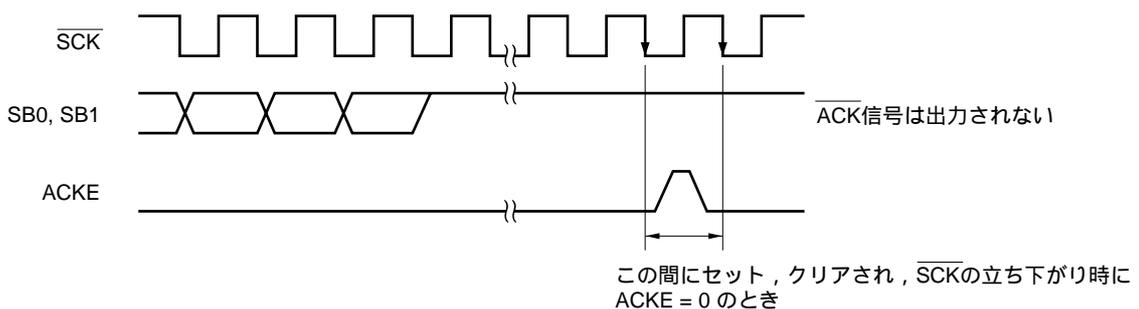


図5 - 80 ACKDの動作

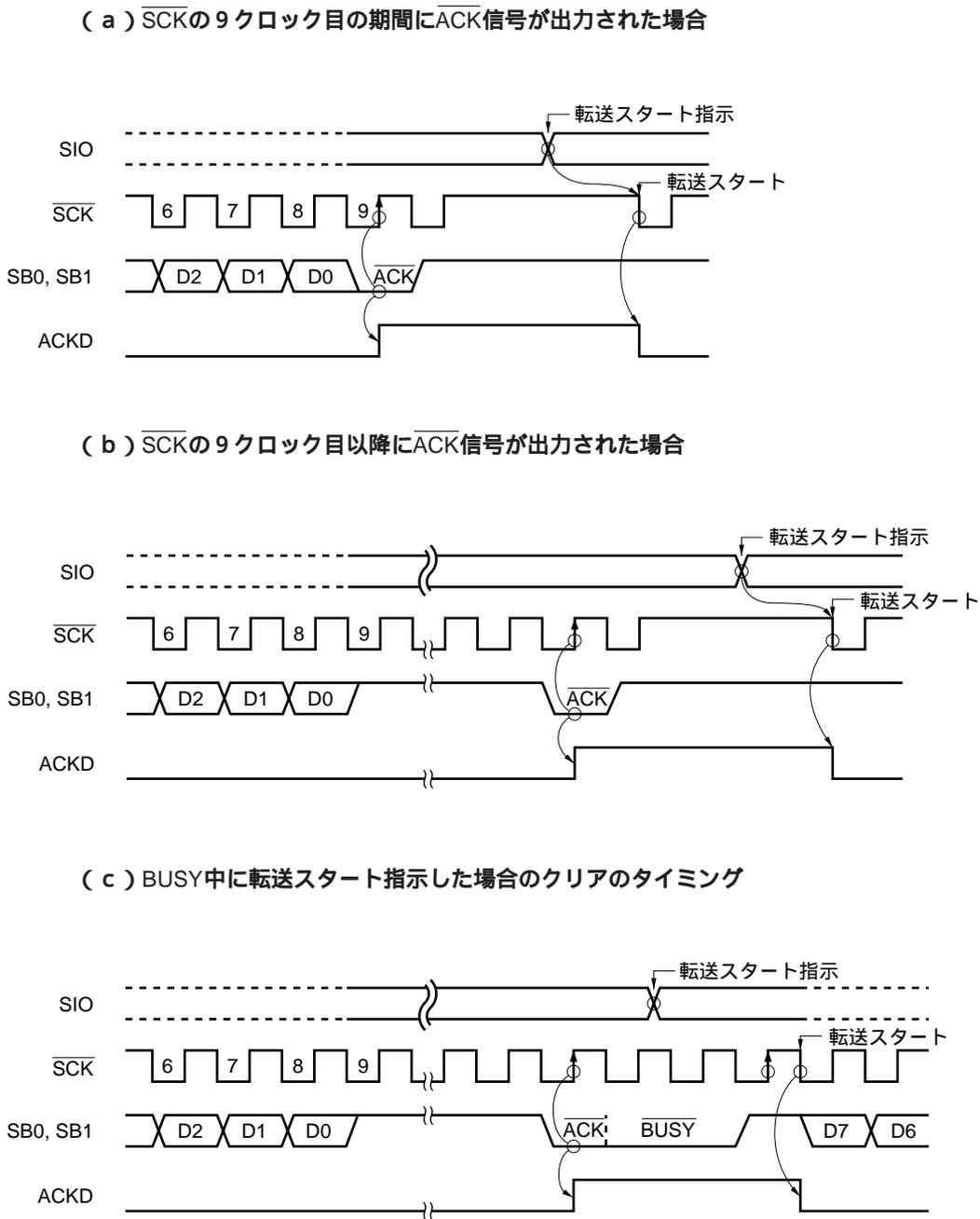


図5 - 81 BSYEの動作

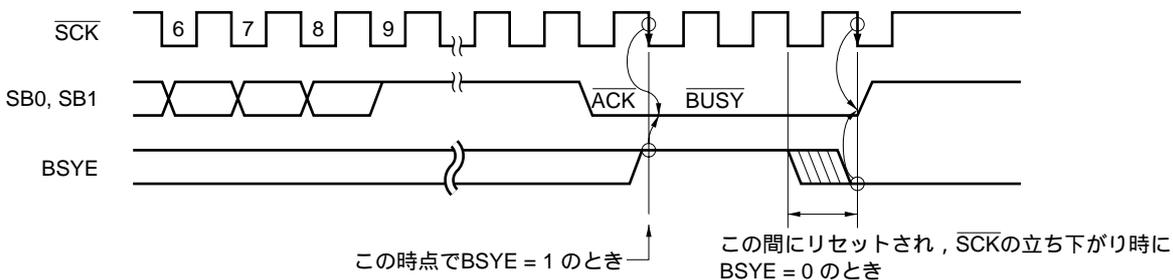


表 5 - 12 SBIモードにおける各種の信号 (1/2)

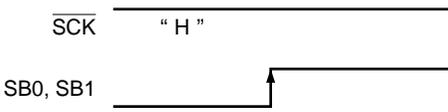
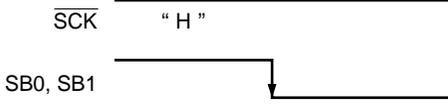
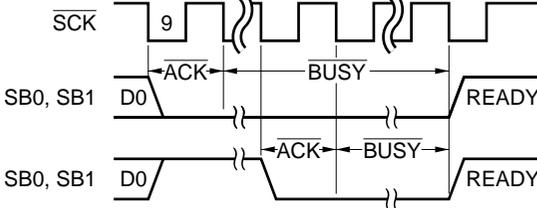
信号名称	出力するデバイス	定義	タイミング・チャート	出力される条件	フラグへの影響	信号の意味
バス・リリース信号 (REL)	マスタ	$\overline{SCK} = 1$ のときの、 SB0, SB1の立ち上がりエッジ		・ RELTのセット	・ RELDをセット ・ CMDDをクリア	続いてCMD信号を出力し、送信データがアドレスであることを示す。
コマンド信号 (CMD)	マスタ	$\overline{SCK} = 1$ のときの、 SB0, SB1の立ち下がりエッジ		・ CMDTのセット	・ CMDDをセット	i) REL信号出力後送信データはアドレス。 ii) REL信号出力なし送信データはコマンド。
アクリッジ信号 (ACK)	マスタ/ スレーブ	シリアル受信完了後、 \overline{SCK} の1クロックの期間SB0, SB1に出力されるロウ・レベルの信号	[同期ビジィ出力]	ACKE = 1 ACKTのセット	・ ACKDをセット	受信完了。
ビジィ信号 (BUSY)	スレーブ	[同期ビジィ信号] アクリッジ信号に続いてSB0, SB1に出力されるロウ・レベルの信号		・ BSYE = 1	-	処理中のため、シリアル受信不可能状態。
レディ信号 (READY)	スレーブ	シリアル転送開始前、完了後SB0, SB1に出力されるハイ・レベルの信号		BSYE = 0 SIOへのデータ書き込み命令実行 (転送開始指示)	-	シリアル受信可能状態。

表 5 - 12 SBIモードにおける各種の信号 (2/2)

信号名称	出力するデバイス	定義	タイミング・チャート	出力される条件	フラグへの影響	信号の意味
シリアル・クロック (\overline{SCK})	マスタ	アドレス/コマンド/データ、 \overline{ACK} 信号、同期 \overline{BUSY} 信号などの出力のための同期クロック。 最初の8個でアドレス/コマンド/データを転送する。		CSIE = 1のときの、SIOへのデータ書き込み命令実行 (シリアル転送のスタート指示) ^{注2}	IRQCSIをセット (9クロック目の立ち上がり) ^{注1}	シリアル・データ・バスへの信号出力のタイミング
アドレス (A7-A0)	マスタ	REL信号、CMD信号出力後に、 \overline{SCK} に同期して転送される8ビット・データ。				シリアル・バス上のスレーブ・デバイスのアドレス値
コマンド (C7-C0)	マスタ	REL信号は出力されず、CMD信号のみ出力されたあと、 \overline{SCK} に同期して転送される8ビット・データ。				スレーブ・デバイスへの指示・メッセージ
データ (D7-D0)	マスタ/ スレーブ	REL信号、CMD信号ともに出力されず、 \overline{SCK} に同期して転送される8ビット・データ。				スレーブ、またはマスタ・デバイスが処理する数値

注1．WUP=0のとき、常に9クロック目の \overline{SCK} の立ち上がりでIRQCSIをセットする。

WUP=1のとき、アドレスを受信し、そのアドレスがスレーブ・アドレス・レジスタ(SVA)の値と一致したときのみ、 \overline{SCK} の9クロック目の立ち上がりでIRQCSIをセットする。

2． \overline{BUSY} 状態のときは、READY状態になったあと、転送スタートする。

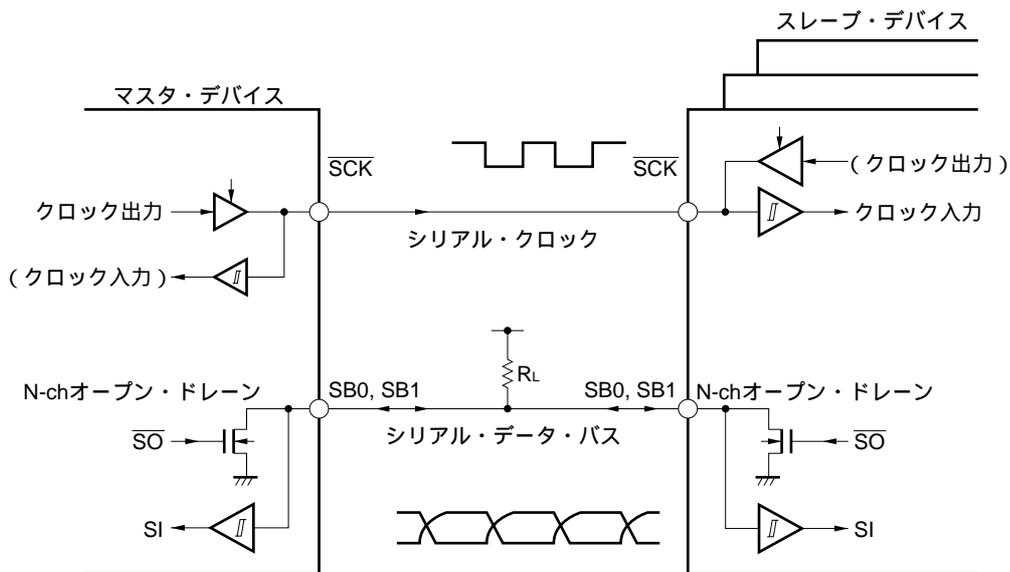
(6) 端子構成

シリアル・クロック端子 ($\overline{\text{SCK}}$) と、シリアル・データ・バス端子 (SB0またはSB1) の構成は、次のようになっています。

- (a) $\overline{\text{SCK}}$ シリアル・クロックを入出力するための端子
 - マスタ CMOS, プッシュプル出力
 - スレーブ ... シュミット入力
- (b) SB0, SB1 シリアル・データの入出力兼用端子
 - マスタ, スレーブともに出力はN-chオープン・ドレイン, 入力は, シュミット入力

シリアル・データ・バス・ラインは、出力がN-chオープン・ドレインのため、外部にプルアップ抵抗が必要となります。

図5 - 82 端子構成図



注意 データ受信時にはN-chトランジスタをオフする必要がありますので、SIOにはあらかじめFFHを書き込んでおいてください。転送中は常にオフさせることができます。ただし、ウェイク・アップ機能指定ビット (WUP) = 1 の場合は、N-chトランジスタは常にオフとなりますので、受信前に、SIOにFFHを書き込む必要はありません。

(7) アドレスの一致検出方法

SBIモードでは、マスタのアドレス通信より、特定のスレーブ・デバイスを選択し、通信がスタートされます。

アドレス一致の検出は、ハードウェアで行います。スレーブ・アドレス・レジスタ (SVA) を備え、ウエイク・アップ状態 (WUP = 1) では、マスタから送信されたアドレスとSVAに設定された値が一致したときのみ、IRQCSIをセットします。

注意 1 . スレーブの選択, 非選択状態の検出は, バス・リリース (RELD = 1 の状態) のあとに受信したスレーブ・アドレスの一致検出により行います。

この一致検出は、通常、WUP = 1 の状態で発生するアドレスの一致割り込み (IRQCSI) を使用します。したがって、スレーブ・アドレスによる選択, 非選択の検出は、WUP = 1 の状態で行ってください。

2 . WUP = 0 で、割り込みを使用せずに選択, 非選択を検出する場合には、アドレスの一致検出による方法を使用せず、あらかじめプログラムで設定したコマンドの送受信で行ってください。

(8) エラーの検出

SBIモードでは、送信中のシリアル・バスSB0またはSB1の状態が送信しているデバイスのシフト・レジスタSIOにも取り込まれるため、次の方法によって送信エラーの検出をすることができます。

(a) 送信開始前と送信終了後のSIOのデータを比較する方法

この場合、2つのデータが異なっていれば送信エラーが発生したと判断します。

(b) スレーブ・アドレス・レジスタ (SVA) を利用する方法

送信データをSIOとSVAにもセットし、送信を行います。送信終了後に、シリアル動作モード・レジスタ (CSIM) のCOIビット (アドレス・コンパレータからの一致信号) をテストし、“1” ならば正常な送信、“0” ならば送信エラーと判断します。

(9) 通信動作

SBIモードでは、マスタがシリアル・バス上に「アドレス」を出力することで複数のデバイスのうち、通信対象となるスレーブ・デバイスを通常1つ選択します。

通信対象デバイスを決定したのちに、マスタ・デバイスとスレーブ・デバイスの間で、コマンド、データの送受信を行い、シリアル通信を実現します。

各データ通信のタイミング・チャートを図5 - 83から図5 - 86に示します。

SBIモードでは、シリアル・クロック (\overline{SCK}) の立ち下がりに同期してシフト・レジスタのシフト動作が行われ、送信データがSOラッチに、SB0/P02端子または、SB1/P03端子からMSBを先頭にして出力されます。また、 \overline{SCK} の立ち上がりでSB0 (またはSB1) 端子に入力された受信データがシフト・レジスタにラッチされます。

図5 - 83 マスタ・デバイスからスレーブ・デバイス (WUP = 1) へのアドレス送信動作

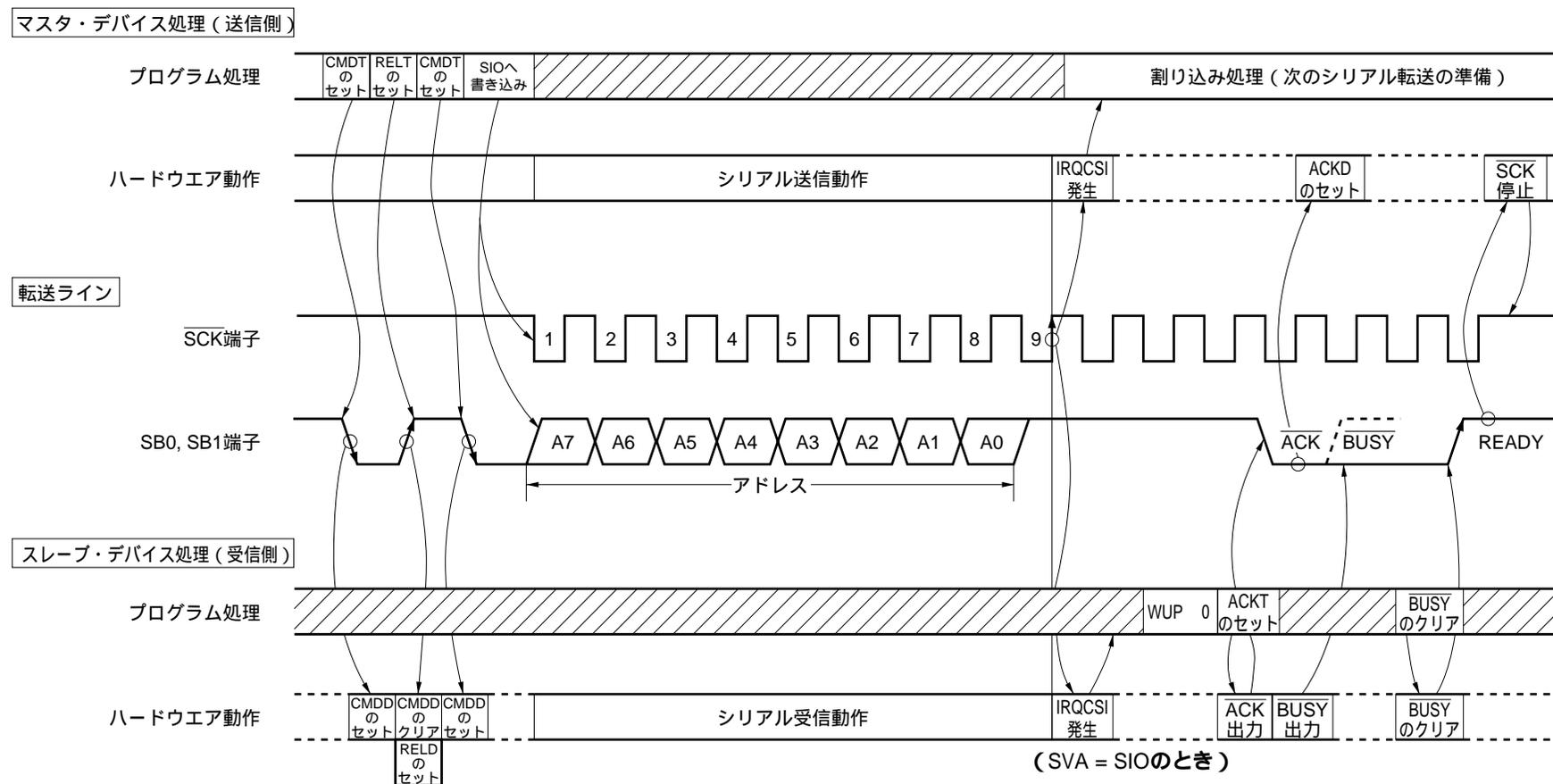


図5 - 84 マスタ・デバイスからスレーブ・デバイスへのコマンド送信動作

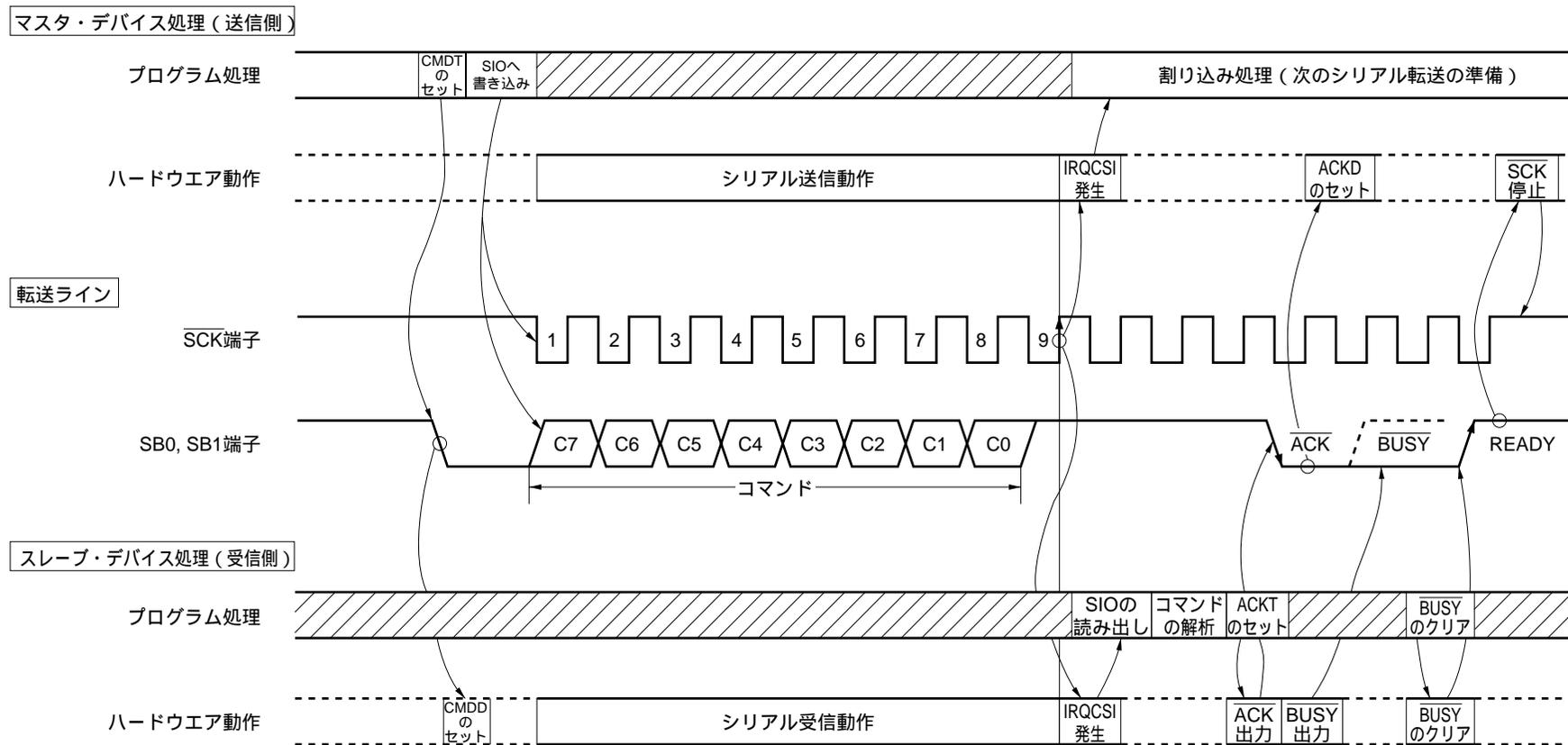


図5 - 85 マスタ・デバイスからスレーブ・デバイスへのデータ送信動作

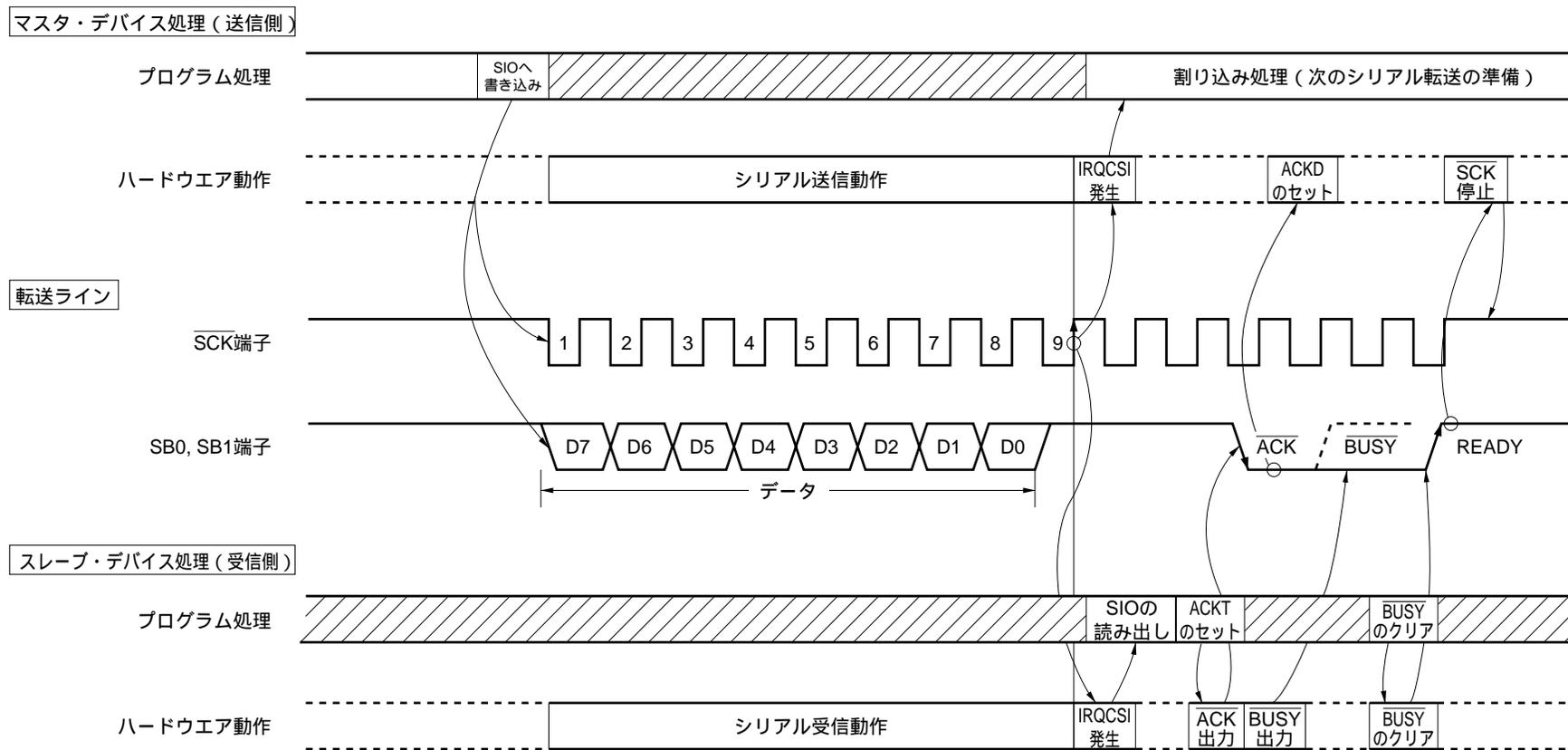
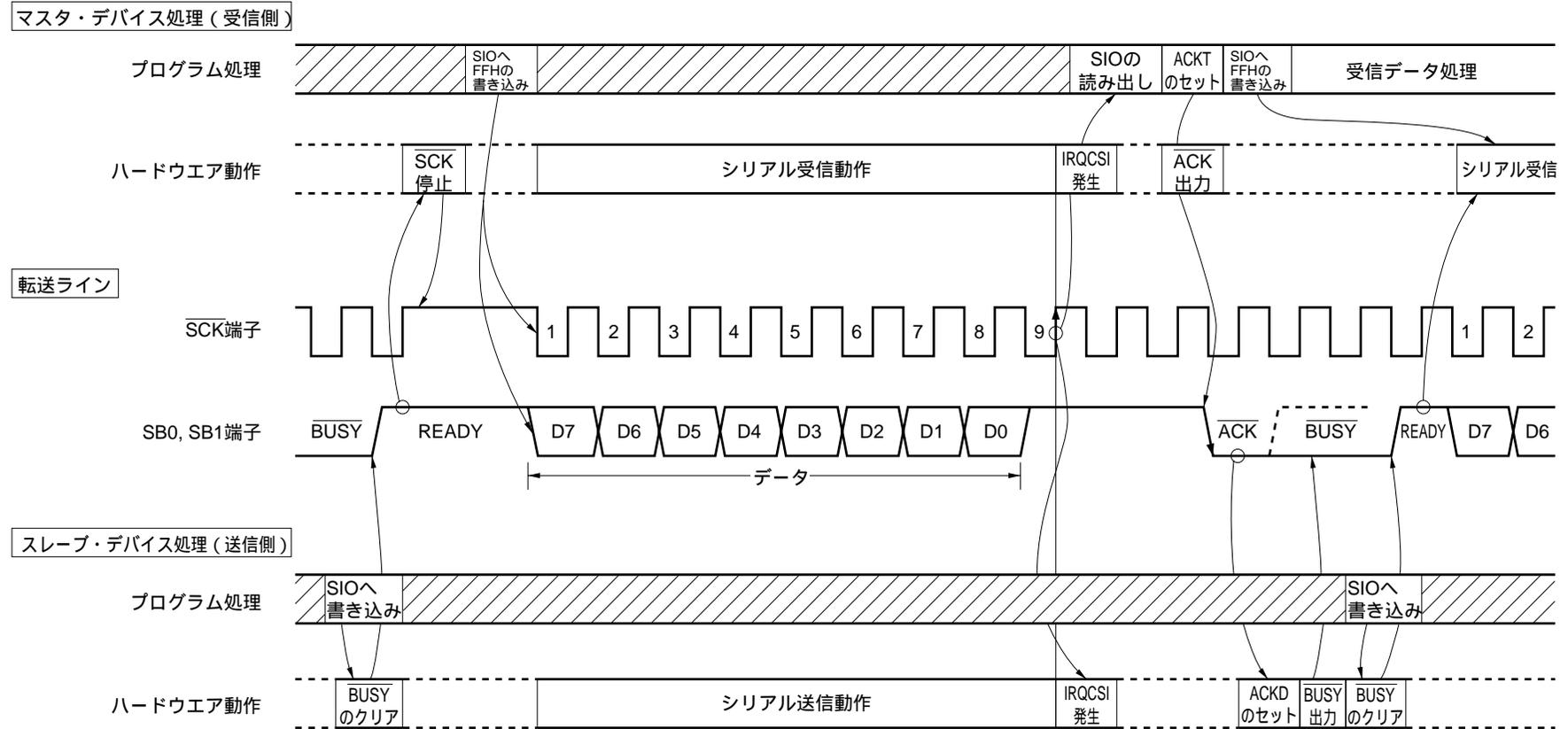


図5 - 86 スレーブ・デバイスからマスタ・デバイスへのデータ送信動作



(10) 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、シフト・レジスタ (SIO) に転送データをセットすることで開始します。

シリアル・インタフェース動作 許可/禁止ビット (CSIE) = 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 $\overline{\text{SCK}}$ がハイ・レベルの状態

注意 1 . シフト・レジスタにデータを書き込んだあと、CSIEを“ 1 ”にしてもスタートはしません。

2 . データ受信時にはN-chトランジスタをオフする必要がありますので、SIOにはあらかじめFFHを書き込んでおいてください。

ただし、ウエイク・アップ機能指定ビット (WUP) = 1 の場合は、N-chトランジスタは常にオフとなりますので、受信前に、SIOにFFHを書き込む必要はありません。

3 . スレープがビジィ状態のときに、SIOにデータを書き込んだ場合、そのデータは失われません。

ビジィ状態が解除されて、SB0 (またはSB1) 入力が高レベル (レディ) 状態になったときに転送がスタートします。

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 (IRQCSI) をセットします。

例 HLレジスタで指定されたRAMのデータをSIOに転送すると同時に、SIOのデータをアキュムレータに取り込み、シリアル転送をスタートする。

```
MOV  XA,    @HL    ; RAMから送信データを取り出す
SEL  MB15    ; またはCLR1 MBE
XCH  XA,    SIO    ; 送信データと受信データを交換し、転送をスタートする。
```

(11) SBIモードの注意事項

(a) スレーブの選択、非選択状態の検出は、バス・リリース (RELD = 1 の状態) のあとに受信したスレーブ・アドレスの一致検出により行います。

この一致検出は、通常、WUP = 1 の状態で発生するアドレスの一致割り込み (IRQCSI) を使用します。したがって、スレーブ・アドレスによる選択、非選択の検出は、WUP = 1 の状態で行ってください。

(b) WUP = 0 で、割り込みを使用せずに選択、非選択を検出する場合には、アドレスの一致検出による方法を使用せず、あらかじめプログラムで設定したコマンドの送受信で行ってください。

(c) $\overline{\text{BUSY}}$ 信号出力中に WUP = 1 とすると、 $\overline{\text{BUSY}}$ が解除されません。SBI では、 $\overline{\text{BUSY}}$ の解除指示後、次のシリアル・クロック ($\overline{\text{SCK}}$) の立ち下がりまで $\overline{\text{BUSY}}$ 信号が出力されます。WUP = 1 とするときは、必ず $\overline{\text{BUSY}}$ を解除したのちに、SB0 (または SB1) 端子がハイ・レベルになったことを確認してから WUP = 1 としてください。

(12) SBIモードの応用

この項では、SBIモードでシリアル・データ通信を行う応用例を紹介します。この応用例では、 μ PD753208をシリアル・バス上のマスタCPU、スレーブCPUのどちらでも動作できるようになっています。

また、コマンドにより、マスタの変更もできるようになっています。

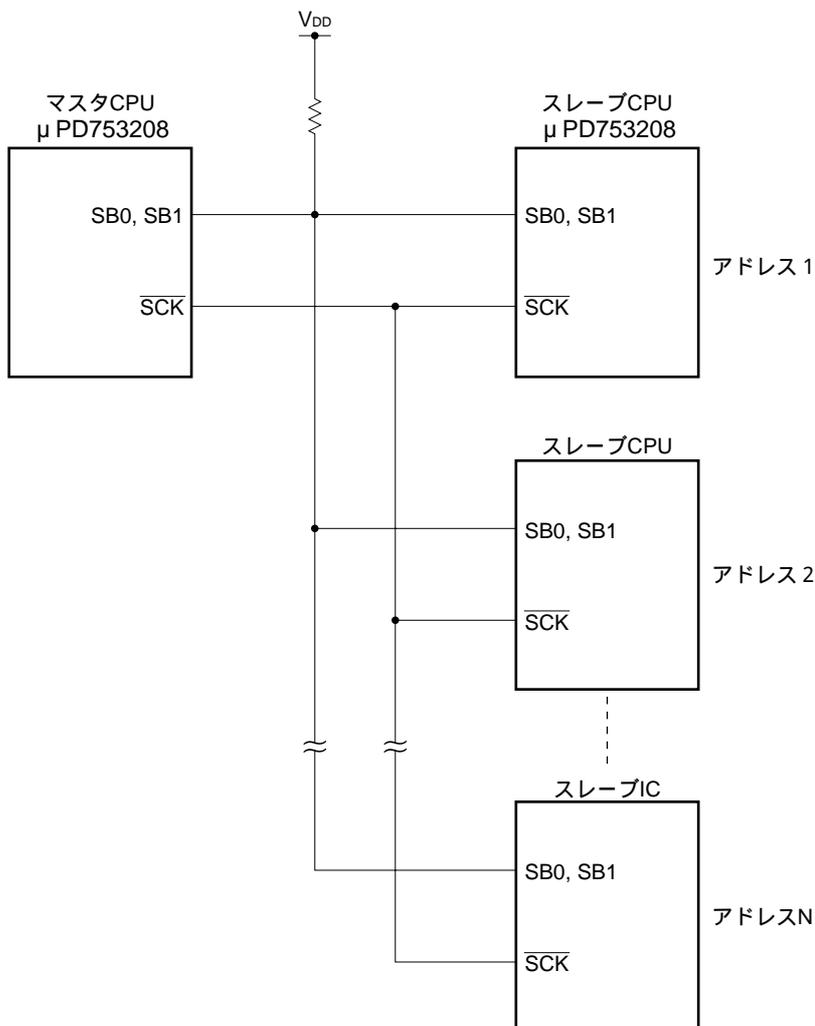
(a) シリアル・バス構成

この項に記載されている応用例でのシリアル・バスの構成は、 μ PD753208がシリアル・バス内のデバイスの1つとしてバス・ラインに接続されているものとします。

μ PD753208の使用する端子は、シリアル・データ・バスSB0（またはSB1）とシリアル・クロック $\overline{\text{SCK}}$ 2本です。

図5 - 87にシリアル・バスの構成例を示します。

図5 - 87 シリアル・バス構成例



(b) コマンド説明

コマンドの種類

この応用例では次に示すコマンドを設定します。

READコマンド	: スレーブからマスタへのデータ転送を行います。
WRITEコマンド	: マスタからスレーブへのデータ転送を行います。
ENDコマンド	: WRITEコマンドの完了をスレーブに示します。
STOPコマンド	: WRITEコマンドの中断をスレーブに示します。
STATUSコマンド	: スレーブ側の状態を読み出します。
RESETコマンド	: 現在選択されているスレーブを非選択にします。
CHGMSTコマンド	: スレーブ側にマスタの権利を受け渡します。

通信手順

マスタ，スレーブ間の通信手順は次のように行います。

マスタが通信しようとするスレーブのアドレスを送信して，スレーブの選択を行う（チップ・セレクト）ことにより通信を開始します。

アドレスを受信したスレーブは $\overline{\text{ACK}}$ を返してマスタとの通信を行います（非選択状態より選択状態になります）。

の処理で選択されたスレーブとマスタとの間で，コマンドおよびデータの転送より通信を行います。

ただし，コマンド，データの転送はマスタ，スレーブの1対1の転送なので，ほかのスレーブは非選択状態である必要があります。

通信は，スレーブが非選択状態になることにより終了します。スレーブが非選択状態になるのは次の場合です。

RESETコマンドをマスタより送信すると，選択されていたスレーブは非選択状態になります。

CHGMSTコマンドによりマスタに変更した場合，マスタからスレーブになったデバイスは非選択状態になります。

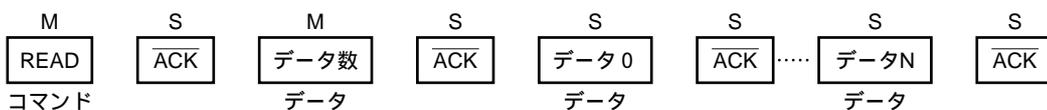
コマンドのフォーマット

各コマンドの転送フォーマットを以下に示します。

READコマンド

スレーブからの読み出しを行うコマンドです。読み出すデータ数は1-256バイトの可変とし、マスタからパラメータとしてデータ数を指定します。データ数として00Hが指定された場合は、256バイトのデータ転送が指定されたことになります。

図5 - 88 READコマンドの転送フォーマット



備考 M : マスタが出力
S : スレーブが出力

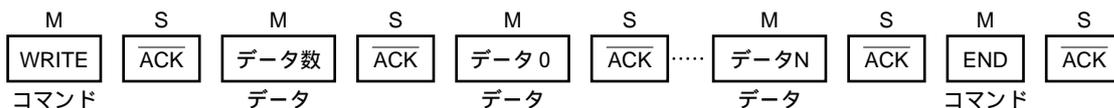
スレーブはデータ数を受信後、送信可能なデータがそのデータ数以上あれば、 $\overline{\text{ACK}}$ を返します。データが不足すれば、 $\overline{\text{ACK}}$ を返さずにエラーとなります。

マスタはデータ受信時には1バイト受信することに $\overline{\text{ACK}}$ をスレーブに送ります。

WRITEコマンド, ENDコマンド, STOPコマンド

スレーブへのデータ書き込みを行うコマンドです。書き込みデータ数は、1-256バイトの可変とし、マスタがパラメータとしてデータ数を指定します。データ数として00Hが指定された場合は、256バイトのデータ転送が指定されます。

図5 - 89 WRITE, ENDコマンドの転送フォーマット



備考 M : マスタが出力
S : スレーブが出力

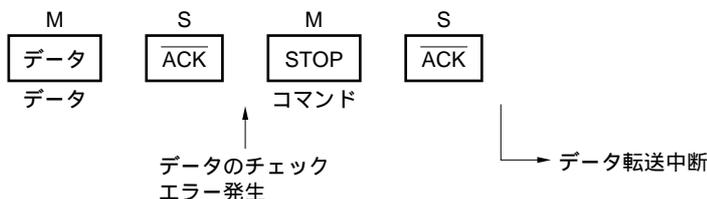
スレーブはデータ数を受信後、受信データを格納するエリアがそのデータ数以上あれば、ACKを返します。格納するエリアが不足すれば、ACKを返さずエラーとなります。

マスタは、すべてのデータの転送が終了するとENDコマンドを送信します。ENDコマンドにより、すべてのデータが正しく転送されたことをスレーブに知らせます。

また、スレーブは、すべてのデータ受信前でもENDコマンドを受けます。この場合ENDコマンドを受信する直前のデータまでを有効にします。

マスタはデータ送信時には、送信前と送信後のSIOの内容を比較することにより、データが正しくバス上に出力されたことを確かめます。送信前と送信後のSIOの内容が変化していたならば、STOPコマンドを送信してデータ転送を中断します。

図5 - 90 STOPコマンドの転送フォーマット



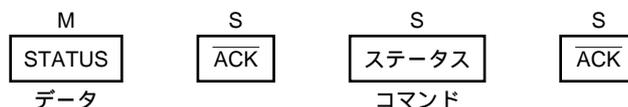
備考 M : マスタが出力
S : スレーブが出力

スレーブはSTOPコマンドを受信すると、その直前に受信した1バイトのデータを無効とします。

STATUSコマンド

現在選択されているスレーブ状態（ステータス）を読み出すためのコマンドです。

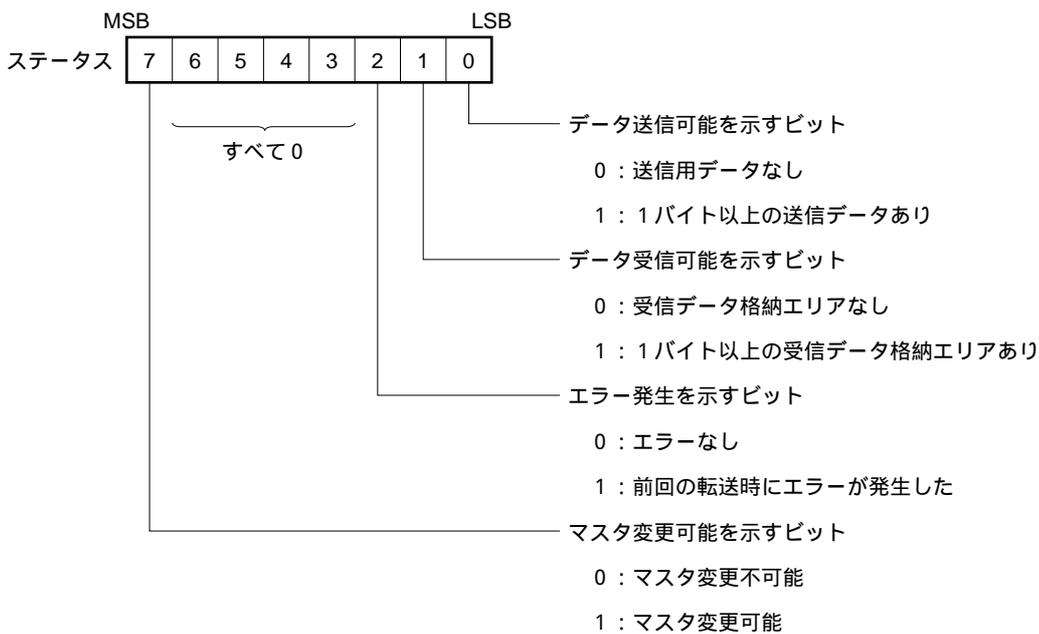
図5 - 91 STATUSコマンドの転送フォーマット



備考 M : マスタが出力
S : スレーブが出力

スレーブが返すステータスのフォーマットは次のようになります。

図5 - 92 STATUSコマンドのステータス・フォーマット



マスタはステータスのデータを受信すると、 $\overline{\text{ACK}}$ をスレーブへ返します。

RESETコマンド

現在選択されているスレーブを非選択状態にするためのコマンドです。RESETコマンドを送信することにより、すべてのスレーブを非選択状態とすることができます。

図5 - 93 RESETコマンドの転送フォーマット



備考 M : マスタが出力
S : スレーブが出力

CHGMSTコマンド

現在選択されているスレーブにマスタの権利を渡すコマンドです。

図5 - 94 CHGMSTコマンドの転送フォーマット



備考 M : マスタが出力
S : スレーブが出力

スレーブはCHGMSTコマンドを受信すると、マスタの権利を受け取れるかを判断し、マスタにデータを返します。データは、次のとおりです。

FFH : マスタ変更可能
00H : マスタ変更不可能

スレーブはデータの転送時に、転送前と転送後のSIOの内容を比較し、一致しなければ $\overline{\text{ACK}}$ を返さずエラーとなります。

マスタは、データを受信すると $\overline{\text{ACK}}$ をスレーブに返し、受信データがFFHであれば、以後スレーブとして動作します。スレーブは、FFHのデータを送信し、マスタから $\overline{\text{ACK}}$ が返ってきたあとにマスタとして動作します。

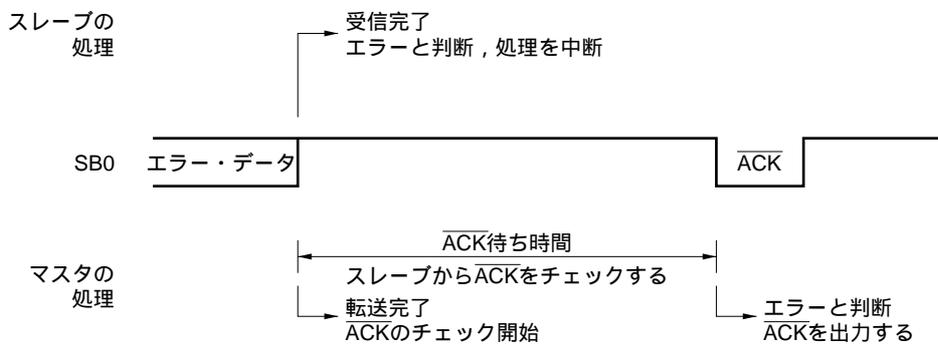
エラーの発生

通信に対してのエラーの発生時の動作について説明します。

スレーブは、マスタに対して $\overline{\text{ACK}}$ を返さないことにより、エラーの発生を知らせます。スレーブが受信をする場合のみエラーが発生すると、ステータスのエラー発生を示すビットをセットし、実行中のコマンドの処理はすべてキャンセルします。

マスタは、1バイトの送信の完了後に、スレーブから $\overline{\text{ACK}}$ をチェックします。送信の完了後、一定時間の間にスレーブから $\overline{\text{ACK}}$ が返されなければ、エラーが発生したものと判断し、マスタが $\overline{\text{ACK}}$ 信号（ダミーとして）を出力します。

図5 - 95 エラー発生時のマスタ、スレーブの動作



エラーには次に示すものがあります。

スレーブ側で発生するエラー

- コマンドの転送フォーマットが違っている場合
- 未定義なコマンドを受信した場合
- READコマンド時に、転送するデータ数が不足する場合
- WRITEコマンド時に、データを格納するエリアが不足する場合
- READ, STATUS, CHGMSTコマンドのデータ送信時に、データが変化する場合

以上、 $\overline{\text{ACK}}$ が発生すると $\overline{\text{ACK}}$ を返しません。

マスタ側で発生するエラー

WRITEコマンドのデータ送信時にデータが変化した場合、STOPコマンドをスレーブに送信します。

5.6.8 SCK端子出力の操作

$\overline{\text{SCK}}$ /P01端子には、出力ラッチが内蔵されているため、通常のシリアル・クロック以外に、ソフトウェア操作によりスタティック出力も可能です。

また、P01出力ラッチの操作により、 $\overline{\text{SCK}}$ の数をソフトウェアで任意に設定することができます（SO/SB0/P02, SI/SB1/P03端子の制御はSBICのRELT, CMDTビットによって行います）。

次に、 $\overline{\text{SCK}}$ /P01端子出力の操作方法を示します。

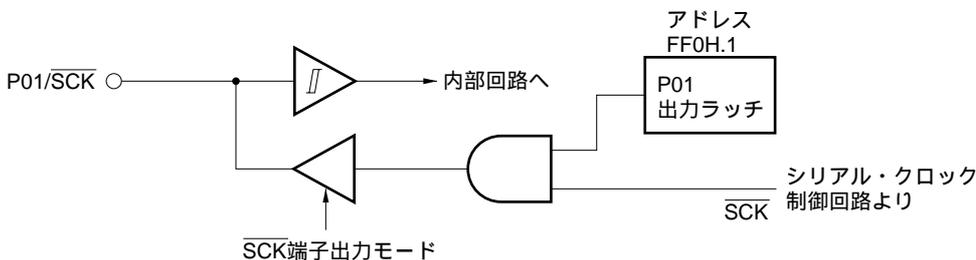
シリアル動作モード・レジスタ（CSIM）を設定します（ $\overline{\text{SCK}}$ 端子：出力モード）。シリアル転送停止中では、シリアル・クロック制御回路からの $\overline{\text{SCK}}$ が1となっています。

P01出力ラッチを、ビット操作命令により操作します。

例 ソフトウェアにより $\overline{\text{SCK}}$ /P01端子に1クロック出力する。

```
SEL    MB15                ; または CLR1 MBE
MOV    XA, #10000011B     ;  $\overline{\text{SCK}}$  (fx/23) , 出力モード
MOV    CSIM, XA
CLR1   OFF0H. 1          ;  $\overline{\text{SCK}}$ /P01 0
SET1   OFF0H. 1          ;  $\overline{\text{SCK}}$ /P01 1
```

図5 - 96 $\overline{\text{SCK}}$ /P01端子の構成



P01出力ラッチは、FF0H番地のビット1にマッピングされています。 $\overline{\text{RESET}}$ 信号発生によりP01出力ラッチは“1”にセットされます。

注意1 通常のシリアル転送中は、P01出力ラッチは1にセットしておく必要があります。

2 . P01出力ラッチのアドレスは下記のように「PORT0.1」では指定できません。オペランドにはアドレス（OFF0H.1）を直接記述するかSCKPで指定してください。ただし、その命令実行時には、あらかじめMBE = 0または（MBE = 1, MBS = 15）としておく必要があります。

★

使用不可

CLR1 PORT0.1
SET1 PORT0.1

使用可

CLR1 OFF0H.1
SET1 OFF0H.1
CLR1 SCKP
SET1 SCKP

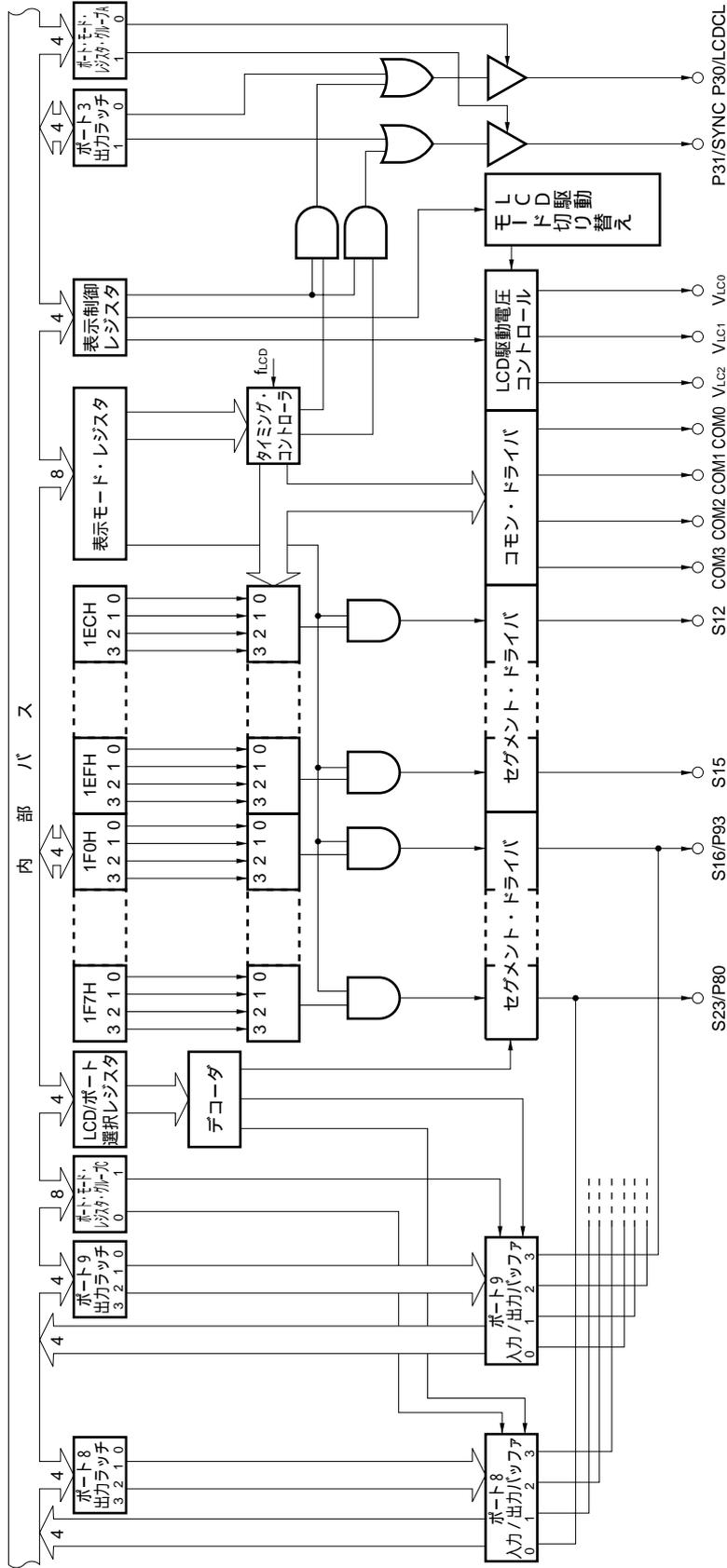
5.7 LCDコントローラ/ドライバ

5.7.1 LCDコントローラ/ドライバの構成

μPD753208は、表示データ・メモリのデータに従い、セグメント信号とコモン信号を発生する表示コントローラと、LCDパネルを直接駆動可能なセグメント・ドライバ、コモン・ドライバを内蔵しています。

LCDコントローラ/ドライバの構成は図5 - 97のようになっています。

図 5 - 97 LCDコントローラ/ドライバのブロック図



5.7.2 LCDコントローラ/ドライバの機能

μPD753208に内蔵されているLCDコントローラ/ドライバの機能は次のようなものです。

- (a) DMA動作により自動的に表示データ・メモリを読み出してセグメント信号とコモン信号を発生。
- (b) 5種類の表示モードが選択可能。
 - スタティック
 - 1/2デューティ (2 時分割) , 1/2バイアス
 - 1/3デューティ (3 時分割) , 1/2バイアス
 - 1/3デューティ (3 時分割) , 1/3バイアス
 - 1/4デューティ (4 時分割) , 1/3バイアス
- (c) 各表示モードにおいて、4種類のフレーム周波数が選択可能。
- (d) セグメント信号出力は最大12本 (S12-S23) , コモン出力は4本 (COM0-COM3) 。
- (e) セグメント信号出力 (S16-S23) は、入出力ポート (PORT8, PORT9) に切り替え可能。
- (f) LCD駆動用電源供給用の分割抵抗内蔵可能 (マスク・オプション) 。
 - ・各種バイアス法, LCD駆動電圧に対応可能。
 - ・表示オフ時には, 分割抵抗へ流れる電流をカット。
- (g) 表示に使用しない表示データ・メモリは通常のデータ・メモリとして使用可能。

各表示モードにおける表示可能な最大画素数を表5 - 13に示します。

表5 - 13 最大表示画素

バイアス法	時分割	使用COM信号	最大画素
-	スタティック	COM0 (COM1, 2, 3)	12 (セグメント12×コモン1) 注1
1/2	2	COM0, 1	24 (セグメント12×コモン2) 注2
	3	COM0, 1, 2	36 (セグメント12×コモン3) 注3
1/3	3		
	4	COM0, 1, 2, 3	48 (セグメント12×コモン4) 注4

注1 . $\frac{5}{8}$. 形のLCDパネルで8セグメント信号/桁のもの1桁。

2 . $\frac{5}{8}$. 形のLCDパネルで4セグメント信号/桁のもの3桁。

3 . $\frac{5}{8}$. 形のLCDパネルで3セグメント信号/桁のもの4桁。

4 . $\frac{5}{8}$. 形のLCDパネルで2セグメント信号/桁のもの6桁。

5.7.3 表示モード・レジスタ (LCDM)

表示モード・レジスタ (LCDM) は、表示モード, LCDクロック, フレーム周波数, および表示出力のオン, オフ制御を指定する8ビット・レジスタです。

LCDMは、8ビット・メモリ操作命令により設定します。またビット3 (LCDM3) のみビット操作命令によりセット, クリア可能です。

$\overline{\text{RESET}}$ 信号発生により全ビット“0”にクリアされます。

図5 - 98 表示モード・レジスタのフォーマット

アドレス	7	6	5	4	3	2	1	0	略号
F8CH	0	0	LCDM5	LCDM4	LCDM3	LCDM2	LCDM1	LCDM0	LCDM

LCDクロックの選択

LCDM5	LCDM4	LCDCL ^注
0	0	$f_w/2^9$ (64 Hz)
0	1	$f_w/2^8$ (128 Hz)
1	0	$f_w/2^7$ (256 Hz)
1	1	$f_w/2^6$ (512 Hz)

注 LCDCLは時計用タイマが動作しているときのみ供給されます。LCDコントローラを用いるときは時計モード・レジスタWMのビット2を“1”にセットしてください。

表示モードの選択

LCDM3	LCDM2	LCDM1	LCDM0	時分割数	バイアス法
0	x	x	x	表示オフ ^注	
1	0	0	0	4	1/3
1	0	0	1	3	1/3
1	0	1	0	2	1/2
1	0	1	1	3	1/2
1	1	0	0	スタティック	
上記以外				設定禁止	

注 全セグメント信号は非選択レベル。

フレーム周波数 (Hz)

LCDCL	$f_w/2^9$ (64 Hz)	$f_w/2^8$ (128 Hz)	$f_w/2^7$ (256 Hz)	$f_w/2^6$ (512 Hz)
表示デューティ				
スタティック	64	128	256	512
1/2	32	64	128	256
1/3	21	43	85	171
1/4	16	32	64	128

$f_w = 32.768$ kHzの場合

f_w : 時計用タイマへの入力クロック ($f_x/128$)

5.7.4 表示制御レジスタ (LCDC)

表示制御レジスタは、次に示すようなLCD駆動の制御を行います。

- ・ コモンおよびセグメント出力の許可 / 禁止。
- ・ LCD駆動用電源供給用の分割抵抗へ流れる電流のカット。
- ・ 外部セグメント信号拡張用コントローラ / ドライバへの同期クロック (LCDCL) , 同期信号 (SYNC) の出力許可 / 禁止。
- ・ 電源電圧によるLCD駆動モードの切り替え (通常モード / 低電圧モード)。
 通常モード.....低消費電流。
 低電圧モード.....低電圧に対応。

注意 $V_{DD} = 2.2V$ でLCDを駆動するときは、必ず低電圧モードにしてください。

LCDCは4ビット・メモリ操作命令により設定します。

\overline{RESET} 信号発生により表示制御レジスタはすべて“0”にクリアされます。

図5 - 99 表示制御レジスタのフォーマット

アドレス	3	2	1	0	略号
F8EH	0	LCDC2	VAC0	LCDC0	LCDC

LCDC, SYNC信号の出力の許可 / 禁止指定ビット

LCDC2	0	LCDC, SYNC信号の出力を禁止。
LCDC2	1	LCDC, SYNC信号の出力を許可。

注意 LCDCL, SYNC信号の出力は将来のシステム拡張に備えた信号です。

現在は信号出力を禁止してください。

LCD駆動モード切り替えビット

VAC0	0	通常モード (2.2V $V_{DD} = 5.5V$)
VAC0	1	低電圧モード (1.8V $V_{DD} = 5.5V$)

LCDC0とLCDM3による表示出力の状態

LCDC0	0	1	
LCDM3	x	0	1
COM0-COM3	“L” を出力 (表示オフ)	表示モードに対応したコモン信号を出力	表示モードに対応したコモン信号を出力
S12-S15	“L” を出力 (表示オフ)	表示モードに対応したセグメント信号を出力 (非選択レベル出力, 表示オフ)	表示モードに対応したセグメント信号を出力 (表示オン)
S16-S23 のセグメント指定端子			
S16-S23 のポート指定端子	入出力ポート (入力ポートか出力ポートかは, ポート・モード・レジスタ・グループC (PMGC) の指定によります)		
分割抵抗への電源供給 (BIAS端子出力)	オフ (ハイ・インピーダンス) ^注	オン (ハイ・レベル) ^注	オン (ハイ・レベル) ^注

注 ()内は分割抵抗を内蔵しない場合

5.7.5 LCD/ポート選択レジスタ (LPS)

LCD/ポート選択レジスタ (LPS) は、セグメント信号出力 (S16-S23) を入出力ポートに切り替えるレジスタです。

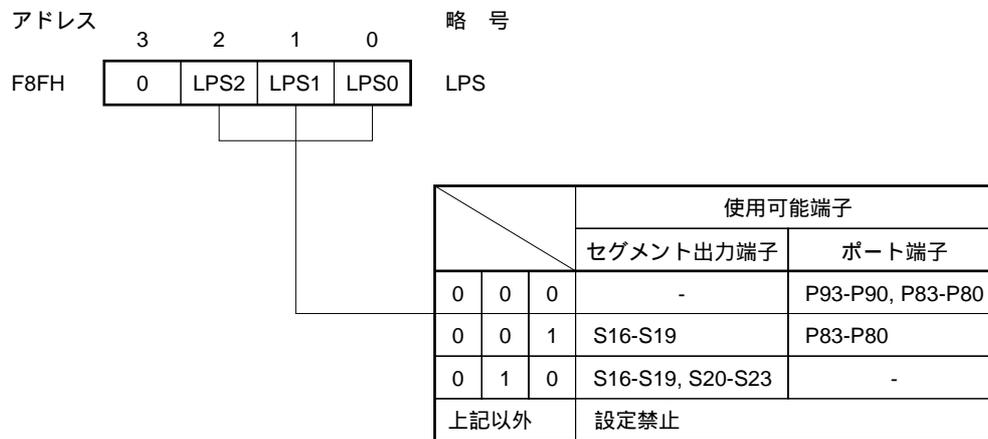
セグメント信号出力のうち、S16-S19はPORT9、S20-S23はPORT8と兼用になっており、4本単位で切り替えられます。

また、LPSの値を“000”にセットすることにより、S16-S23は入出力ポート (PORT9, PORT8) に切り替えることができます。

LPSは4ビット・メモリ操作命令により設定します。

RESET信号発生によりLPSはすべて“0”にクリアされます。

図5 - 100 LCD/ポート選択レジスタのフォーマット



注意1 . リセット時、LPSはすべて“0”にクリアされるため、LCDを使用することはできません。したがって、LCDを使用する場合には、必ずLPSの値を001か010にセットしてください。

2 . LPSのビット3には必ず“0”をセットしてください。

3 . S16/P93-S19/P90, S20/P83-S23/P80端子をセグメント信号出力として使用する場合は、ソフトウェアにより内蔵プルアップ抵抗の接続を指定しないでください。

LPSによりセグメント信号出力に指定したポートは、ポート・モード・レジスタ・グループCにより入力モードを指定しても、フローティングにはなりません。したがって、ソフトウェアにより内蔵プルアップ抵抗の接続を指定する必要はありません。

セグメント信号出力に指定したポートに対して入力命令を実行した場合は、出力ラッチの内容が入力されます。

5.7.6 表示データ・メモリ

表示データ・メモリは、1ECH-1F7Hにマッピングされています。

表示データ・メモリは、CPUの動作とは無関係にLCDコントローラ/ドライバがDMA動作によって読み出すメモリ・エリアです。LCDコントローラは表示データ・メモリのデータに従ってセグメント信号を制御します。

LCD表示やポートとして使用しない領域は、通常のデータ・メモリとして使用できます。

表示データ・メモリは1ビットまたは4ビット単位で操作します。8ビット単位の操作はできません。

表示データ・メモリの各ビットとセグメント出力の対応を図5-102に示します。

図5-101 データ・メモリ・マップ

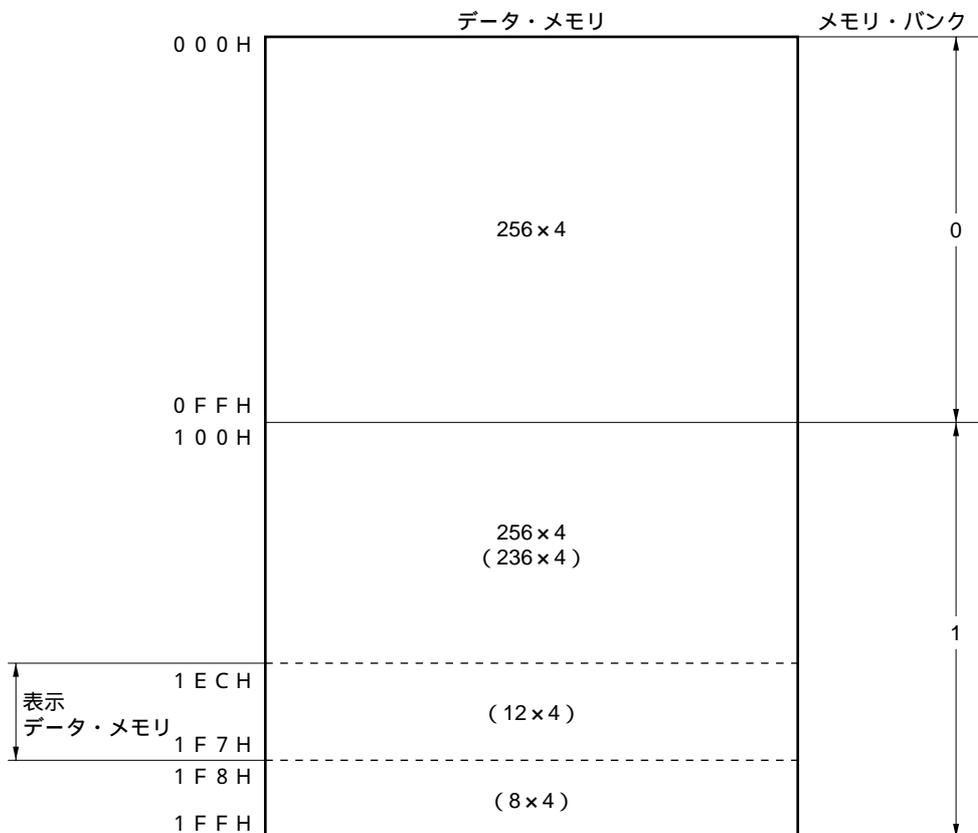
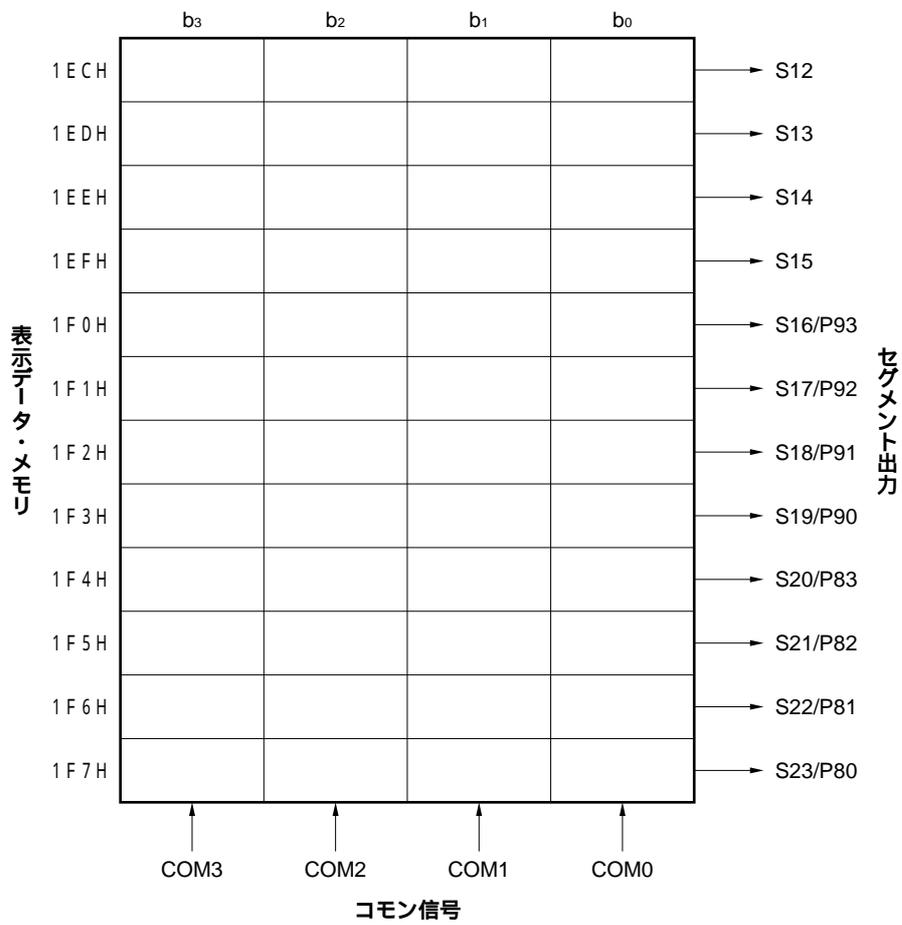


図5 - 102 表示データ・メモリとコモン・セグメント対応



5.7.7 コモン信号とセグメント信号

LCDパネルの各画素は、それに対応するコモン信号とセグメント信号の電位差が一定電圧（LCD駆動電圧 V_{LCD} ）以上になると点灯します。 V_{LCD} 以下の電位差または電位差が0Vになると消灯します。

LCDパネルは、コモン信号とセグメント信号にDC電圧が加えられると劣化するため、AC電圧によって駆動されます。

(1) コモン信号

コモン信号、設定される時分割数に応じて表5-14に示す順で選択タイミングとなり、それらを一周期として繰り返し動作を行います。スタティック・モードの場合はCOM0-COM3は同一信号が出力されます。なお、2時分割の場合のCOM2、COM3端子および3時分割の場合のCOM3端子はオープンにして使用してください。

表5-14 COM信号

COM信号 時分割数	COM0	COM1	COM2	COM3
スタティック				
2			オープン	オープン
3				オープン
4				

(2) セグメント信号

セグメント信号は、データ・メモリの表示データ・メモリ（1ECH-1F7H）の12ロケーションに対応して12本あり、各ロケーションのビット0はCOM0、ビット1はCOM1、ビット2はCOM2、ビット3はCOM3の選択タイミングに同期して自動的に読み出され、各ビットの内容が1ならば選択電圧に変換され、0ならば非選択電圧に変換されてセグメント端子（S12-S23）より出力されます。

以上のことから表示データ・メモリには使用するLCDパネルの前面電極（セグメント信号に対応）と背面電極（コモン信号に対応）がどのような組み合わせで表示パターンを形成するかを確認のうえ、表示したいパターンに1対1に対応するビット・データを書き込むようにします。

なお、スタティック方式の場合の表示データ・メモリのビット1/2/3、2時分割方式の場合のビット2/3、3時分割方式の場合のビット3はアクセスされませんので、表示以外の目的に使用可能です。

(3) コモン信号とセグメント信号の出力波形

コモン信号とセグメント信号には表5-15～5-17に示すようなレベルの電圧が出力されます。両者がともに選択電圧となったときのみ $+V_{LCD}/-V_{LCD}$ の点灯電圧となり、それ以外の組み合わせでは消灯電圧となります。

表5-15 LCD駆動電圧(スタティック)

セグメント信号Sn		選 択	非 選 択
コモン信号COM0		V_{LC0}/V_{SS}	V_{SS}/V_{LC0}
V_{SS}/V_{LC0}		$+V_{LCD}/-V_{LCD}$	0V/0V

表5-16 LCD駆動電圧(1/2バイアス法)

セグメント信号Sn		選 択	非 選 択
コモン信号COMm		V_{LC0}/V_{SS}	V_{SS}/V_{LC0}
選 択	V_{SS}/V_{LC0}	$+V_{LCD}/-V_{LCD}$	0V/0V
非選択	$V_{LC1} = V_{LC2}$	$+ \frac{1}{2} V_{LCD}/ - \frac{1}{2} V_{LCD}$	$- \frac{1}{2} V_{LCD}/ + \frac{1}{2} V_{LCD}$

表5-17 LCD駆動電圧(1/3バイアス法)

セグメント信号Sn		選 択	非 選 択
コモン信号COMm		V_{LC0}/V_{SS}	V_{LC2}/V_{LC1}
選 択	V_{SS}/V_{LC0}	$+V_{LCD}/-V_{LCD}$	$+ \frac{1}{3} V_{LCD}/ - \frac{1}{3} V_{LCD}$
非選択	V_{LC1}/V_{LC2}	$+ \frac{1}{3} V_{LCD}/ - \frac{1}{3} V_{LCD}$	$+ \frac{1}{3} V_{LCD}/ - \frac{1}{3} V_{LCD}$

図5 - 103～5 - 105にコモン信号波形を，図5 - 106にコモン信号とセグメント信号の電位と位相を示します。

図5 - 103 コモン信号波形 (スタティック)

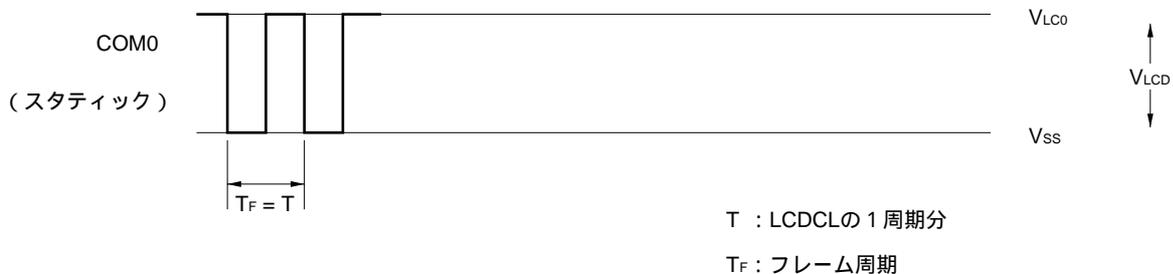


図5 - 104 コモン信号波形 (1/2バイアス法)

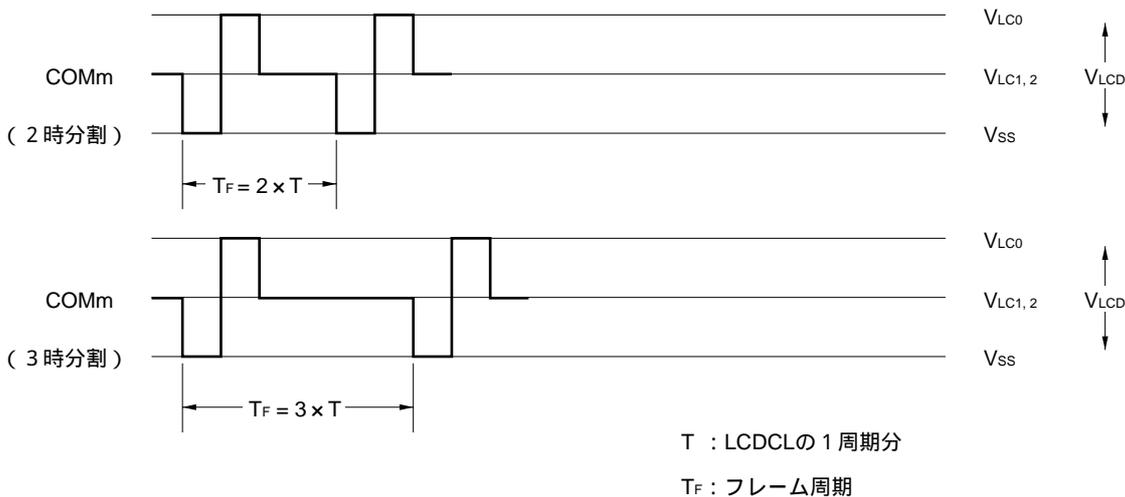


図5 - 105 コモン信号波形 (1/3バイアス法)

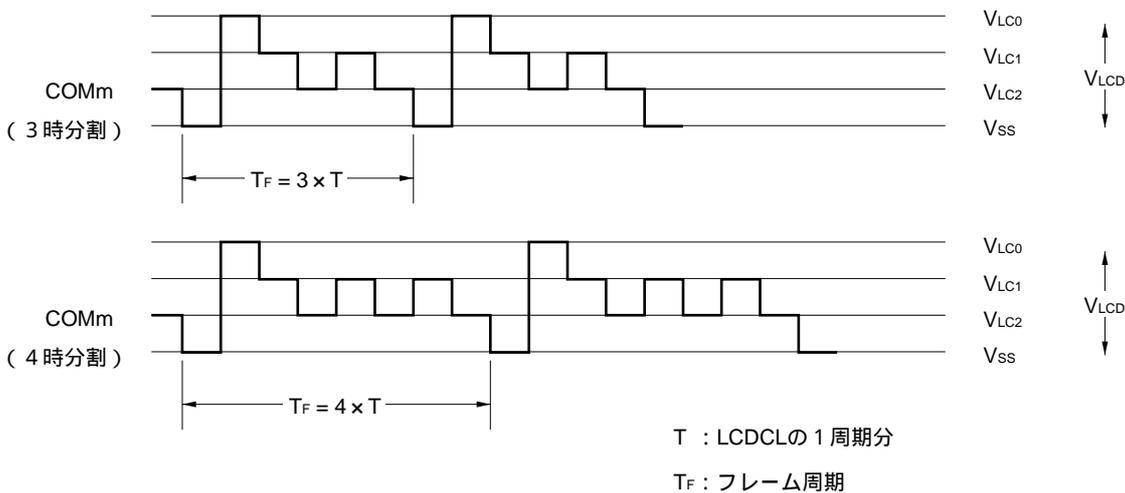
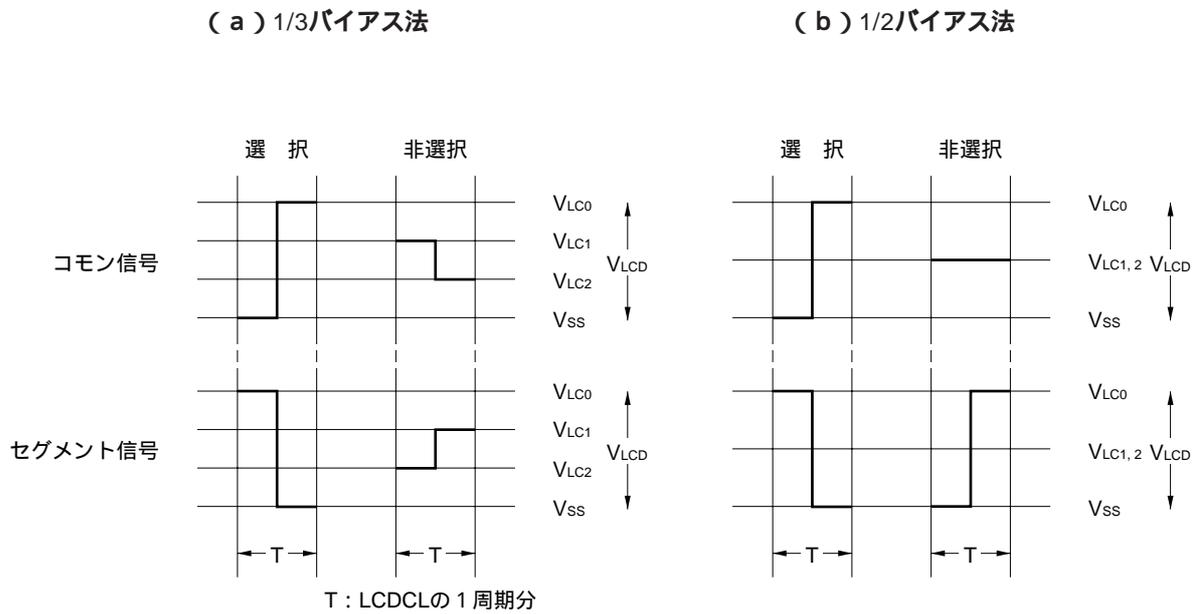
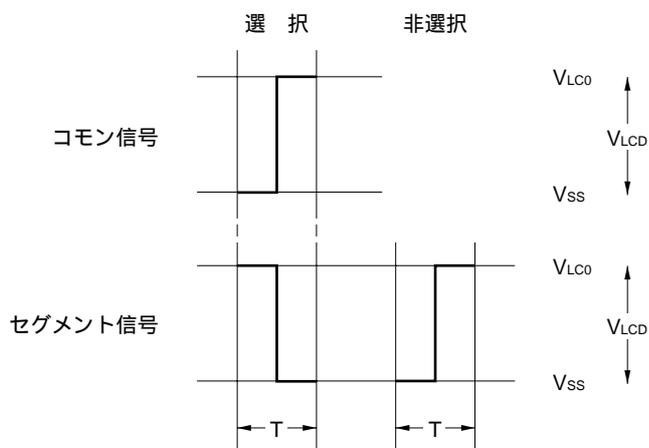


図5 - 106 コモン信号とセグメント信号の電位と位相



(c) スタティック表示モード



5.7.8 LCD駆動用電源 V_{LC0} , V_{LC1} , V_{LC2} の供給

μ PD753208にはLCD駆動用電源供給用に V_{LC0} - V_{LC2} 端子内部に分割抵抗が内蔵可能で、各バイアス法に応じたLCD駆動用電源を外付け分割抵抗なしで供給できます。さらに、各種LCD駆動電圧に対応するためにBIAS端子があり、外部でBIAS端子と V_{LC0} 端子を接続することで対応可能です。

スタティック、1/2および1/3バイアス法に基づく適切なLCD駆動用電源として、次の表に示す値が供給されます。

表5 - 18 LCD駆動用電源の供給値

バイアス法 LCD 駆動用電源	バイアスなし (スタティック ・モード)	1/2	1/3
V_{LC0}	V_{LCD}	V_{LCD}	V_{LCD}
V_{LC1}	$2/3V_{LCD}$	$1/2V_{LCD}$ ^注	$2/3V_{LCD}$
V_{LC2}	$1/3V_{LCD}$		$1/3V_{LCD}$
V_{SS}	0 V	0 V	0 V

注 1/2バイアスのときは V_{LC1} , V_{LC2} 端子を外部で接続する必要があります。

備考 BIAS端子と V_{LC0} 端子が開放時には $V_{LCD} = 3/5V_{DD}$ となります（マスク・オプションで分割抵抗を内蔵しておく必要があります）。

BIAS端子と V_{LC0} 端子が結線時には $V_{LCD} = V_{DD}$ となります。

表5 - 18に従ったLCD駆動用電源の供給例を図5 - 107 (a) , (b) , (c) に示します。

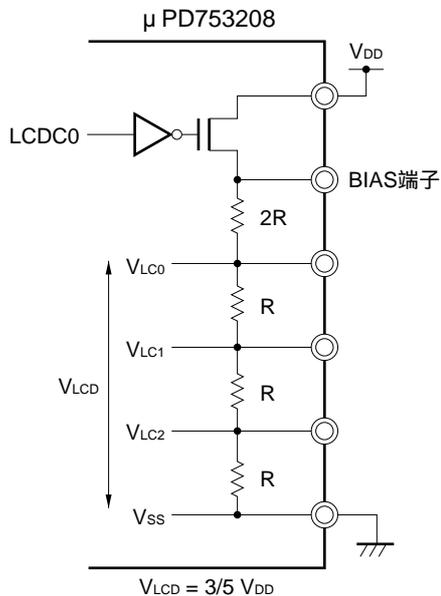
また、表示制御レジスタのビット0 (LCDC0) を “ 0 ” にクリアすることにより分割抵抗を流れる電流をカットすることができます。

このLCD電源のオン、オフ制御は、時計用タイマがシステム・クロックによって動作しているときSTOP命令によってLCDクロック（システム・クロック系を選択している場合）が停止し、LCDにDC電圧が加わってしまうのを防ぐためにも有効です。すなわち、STOP命令を実行する直前に表示制御レジスタのビット0 (LCDC0) を “ 0 ” にクリアし、LCD駆動用電源のすべてを同電位 V_{SS} にすることによって、LCDクロックが停止してもLCDの電極間には電位差が生じないようにすることができます。

図5 - 107 LCD駆動用電源の接続例（分割抵抗内蔵時）

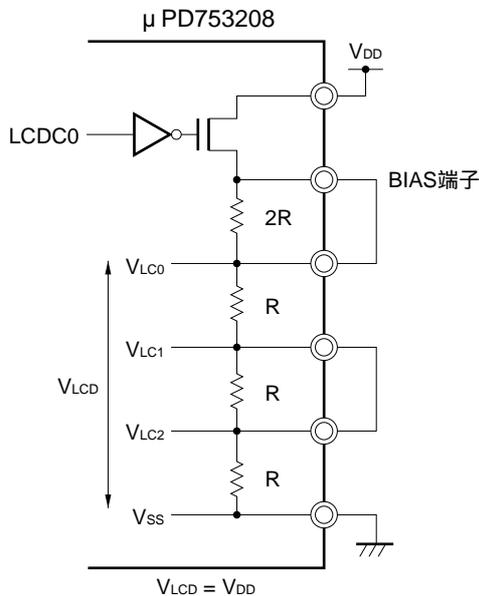
(a) 1/3バイアス法およびスタティック表示モード

($V_{DD} = 5\text{ V}$, $V_{LCD} = 3\text{ V}$ の例)



(b) 1/2バイアス法

($V_{DD} = 5\text{ V}$, $V_{LCD} = 5\text{ V}$ の例)



(c) 1/3バイアス法およびスタティック表示モード

($V_{DD} = 5\text{ V}$, $V_{LCD} = 5\text{ V}$ の例)

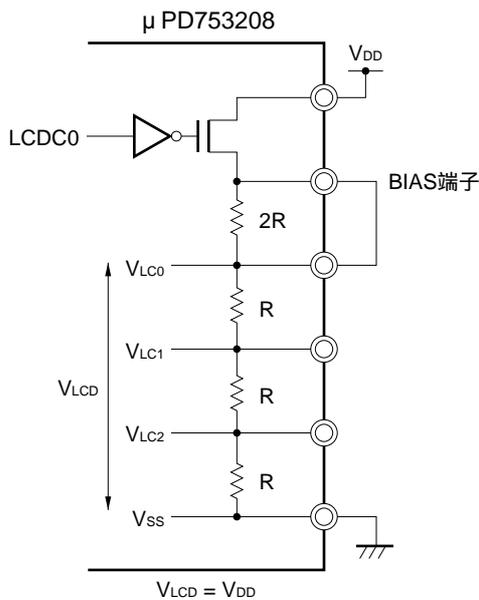
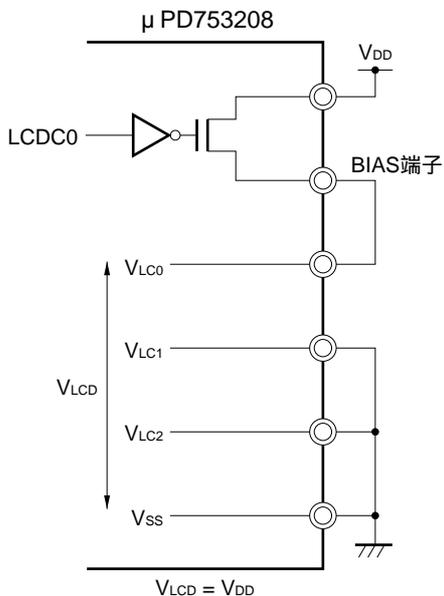


図5 - 108 LCD駆動用電源の接続例（分割抵抗外付け時）

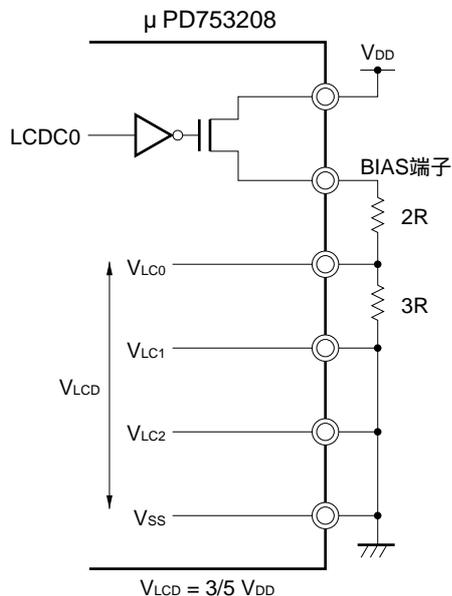
(a) スタティック表示モード^注

($V_{DD} = 5V$, $V_{LCD} = 5V$ の例)



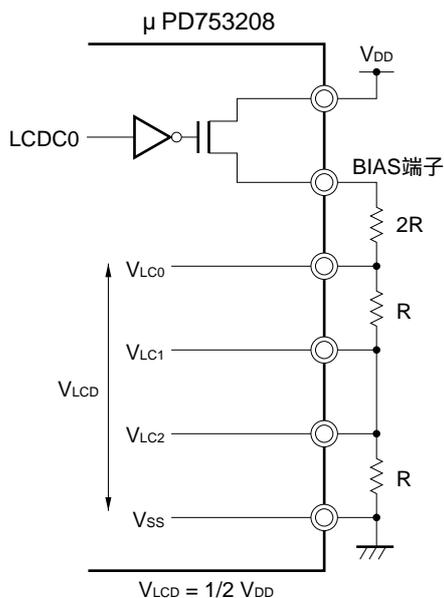
(b) スタティック表示モード

($V_{DD} = 5V$, $V_{LCD} = 3V$ の例)



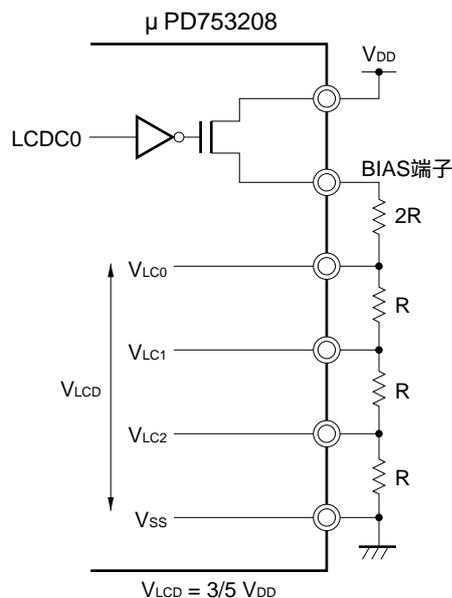
(c) 1/2バイアス法

($V_{DD} = 5V$, $V_{LCD} = 2.5V$ の例)



(d) 1/3バイアス法

($V_{DD} = 5V$, $V_{LCD} = 3V$ の例)



注 LCDC0は常に“1”にセットしておいてください（スタンバイ・モード時を含む）。

5.7.9 表示モード

(1) スタティック表示例

図5 - 110は、図5 - 109の表示パターンを持つスタティック方式の1桁LCDパネルとμPD753208のセグメント信号(S12-S19)およびコモン信号(COM0)との接続を示します。表示例は1で、表示データ・メモリ(1ECH-1F3H番地)の内容はこれに対応しています。

図5 - 110の例では、図5 - 109の表示パターンに従って、COM0のコモン信号のタイミングで表5 - 19に示すような選択、非選択電圧をS12-S19端子に出力する必要があります。

表5 - 19 S12-S19端子の選択、非選択電圧(スタティック表示)

セグメント コモン	S12	S13	S14	S15	S16	S17	S18	S19
COM0	非	選	選	非	非	非	非	非

これにより、S12-S19に対応する表示データ・メモリ(1ECH-1F3H番地)のビット0には、01100000を用意すればよいことが分かります。

S14, S15とCOM0とのLCD駆動波形を図5 - 111に示します。COM0との選択タイミングでS14が選択電圧になるときに、LCD点灯レベルである $+V_{LCD}/-V_{LCD}$ の交流矩形波が発生されることが分かります。

COM1, 2, 3にはCOM0と同じ波形が出力されますので、COM0, 1, 2, 3を接続してドライブ能力を上げることができます。

図5 - 109 スタティック方式LCDの表示パターンと電極結線

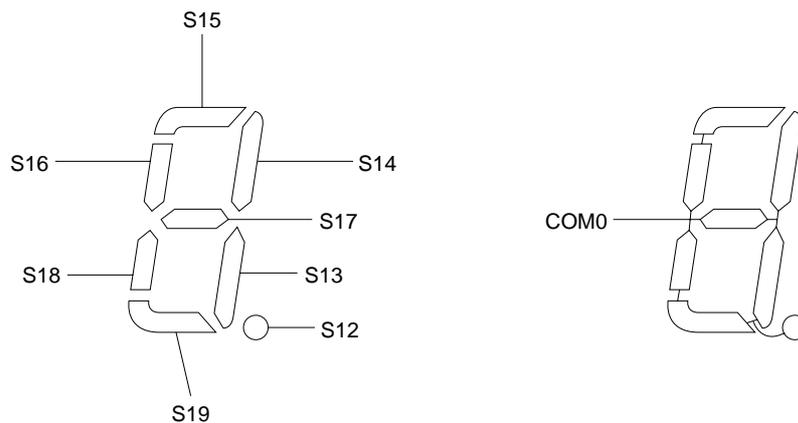


図5 - 110 スタティックLCDパネルの結線例

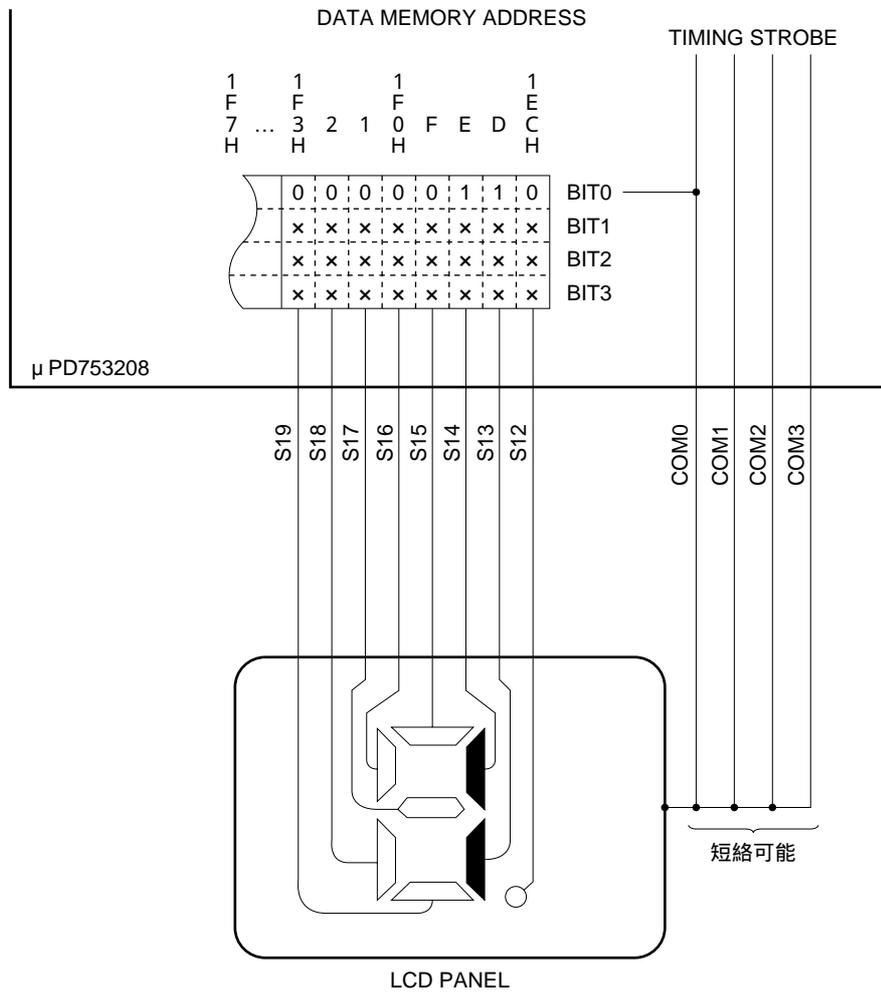
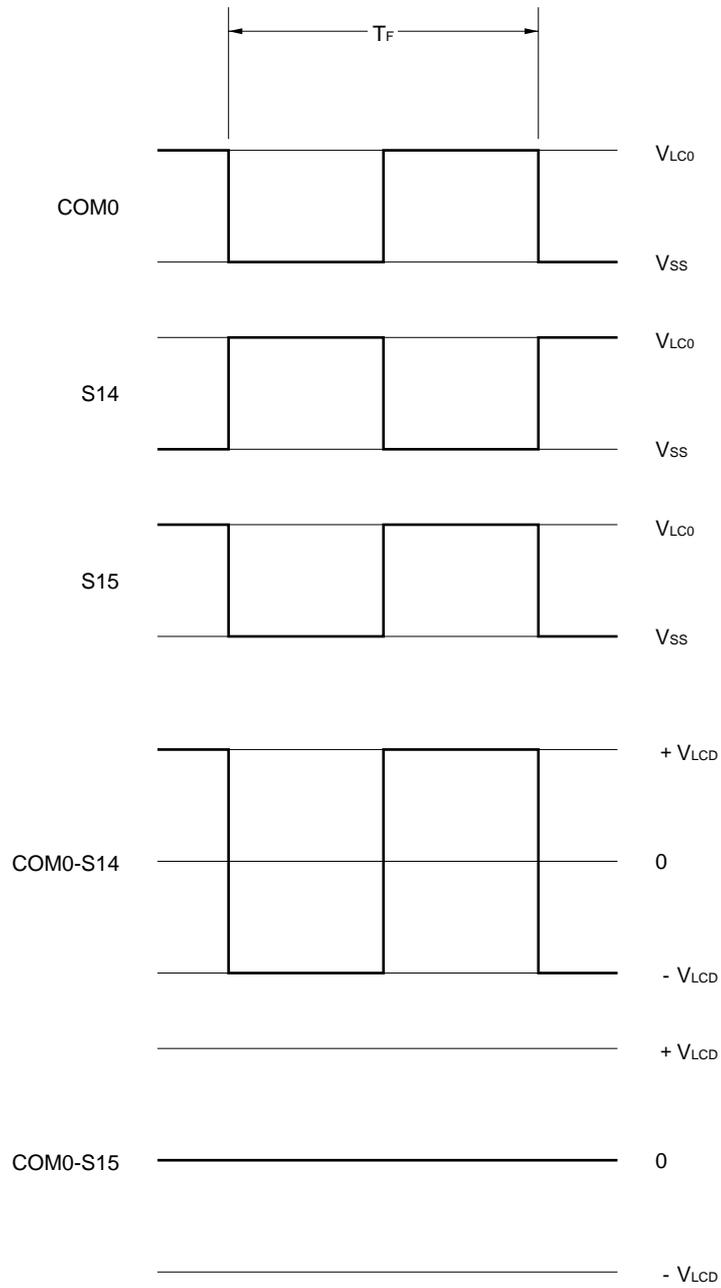


図5 - 111 スタティックLCD駆動波形例



(2) 2時分割表示例

図5 - 113は、図5 - 112の表示パターンを持つ2時分割方式の3桁LCDパネルとμPD753208のセグメント信号(S12-S23)およびコモン信号(COM0, COM1)との接続を示します。表示例は12.3で、表示データ・メモリ(1ECH-1F7H番地)の内容はそれらに対応しています。

ここでは1桁目の3.(3)を例にとって説明します。図5 - 112の表示パターンに従って、COM0, 1の各コモン信号のタイミングで表5 - 20に示すような選択、非選択電圧をS12-S15端子に出力する必要があります。

表5 - 20 S12-S15端子の選択、非選択電圧(2時分割表示例)

セグメント コモン	S12	S13	S14	S15
COM0	選	選	非	非
COM1	非	選	選	選

これにより、たとえばS15に対応する表示データ・メモリ(1EFH番地)には、 $\times \times 10$ を用意すればよいことが分かります。

S15と各コモン信号間のLCD駆動波形例を図5 - 114に示します。COM1の選択タイミングでS15が選択電圧のときに、LCD点灯レベルである $+V_{LCD}/-V_{LCD}$ の交流矩形波が発生されることが分かります。

図5 - 112 2時分割方式LCDの表示パターンと電極結線

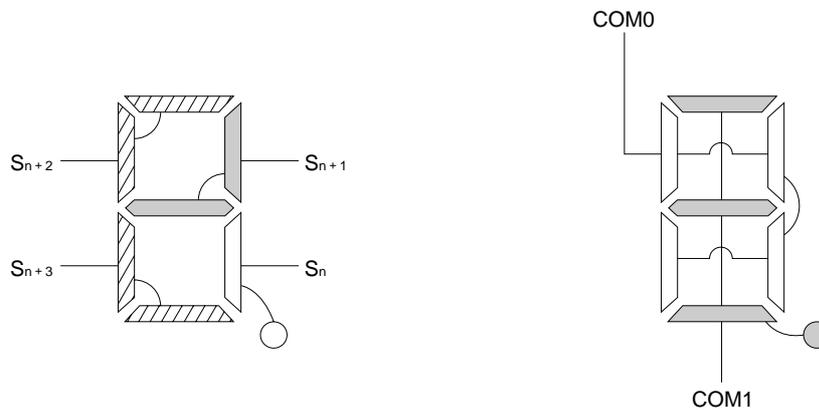
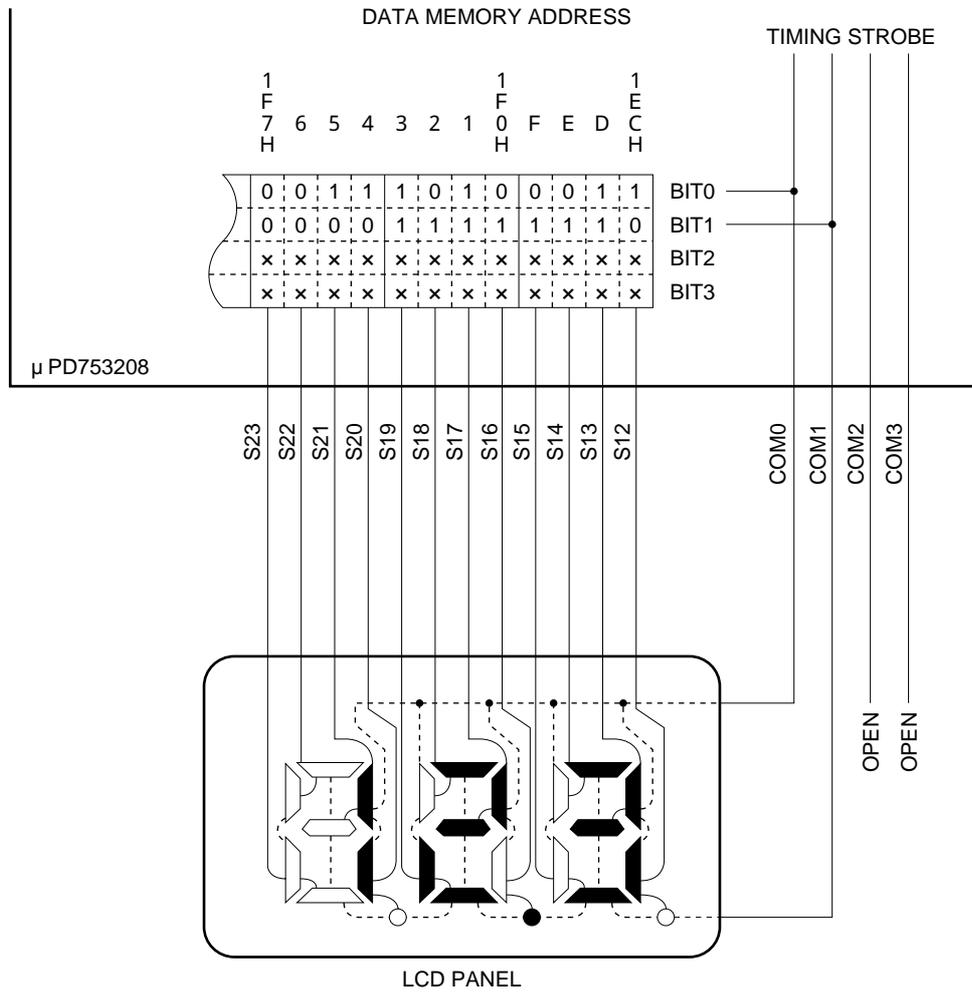
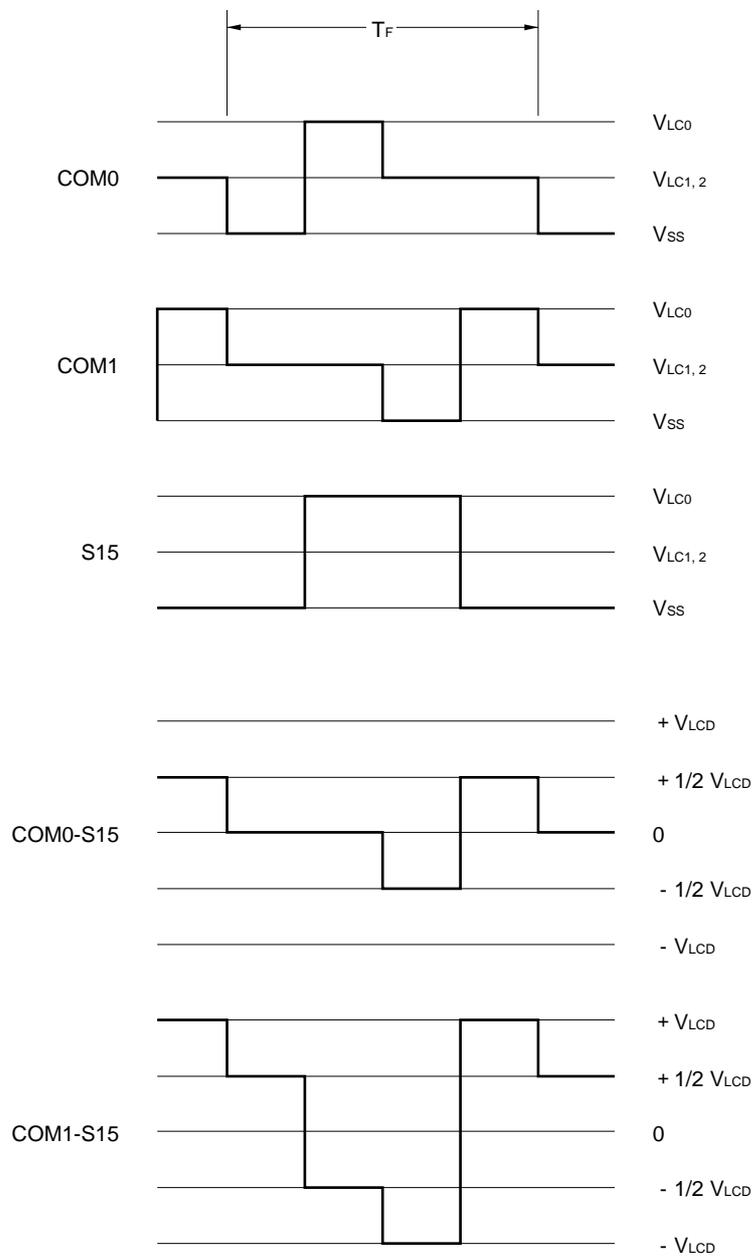


図5 - 113 2時分割LCDパネルの結線例



x : 2時分割表示のため、常に任意のデータをストア可能です。

図5 - 114 2時分割LCD駆動波形例(1/2バイアス法)



(3) 3時分割表示

図5 - 116は、図5 - 115の表示パターンを持つ3時分割方式の4桁LCDパネルとμPD753208のセグメント信号(S12-S23)およびコモン信号(COM0-COM2)との接続を示します。表示例は1236.で、表示データ・メモリ(1ECH-1F7H番地)の内容はこれに対応しています。

ここでは1桁目の6.($\bar{6}$.)を例にとって説明します。図5 - 115の表示パターンに従って、COM0-COM2の各コモン信号のタイミングで表5 - 21に示すような選択、非選択電圧をS12-S14端子に出力する必要があります。

表5 - 21 S12-S14端子の選択、非選択電圧(3時分割表示例)

セグメント コモン	S12	S13	S14
COM0	非	選	選
COM1	選	選	選
COM2	選	選	-

これによりS12に対応する表示データ・メモリ(1ECH番地)には、 $\times 110$ を用意すればよいことが分かります。

S12と各コモン信号間のLCD駆動波形を図5 - 117(1/2バイアス法)、図5 - 118(1/3バイアス法)に示します。COM1の選択タイミングでS12が選択電圧のとき、およびCOM2の選択タイミングでS12が選択電圧のときに、LCD点灯レベルである $+V_{LCD}/-V_{LCD}$ の交流矩形波が発生されることが分かります。

図5 - 115 3時分割方式LCDの表示パターンと電極結線

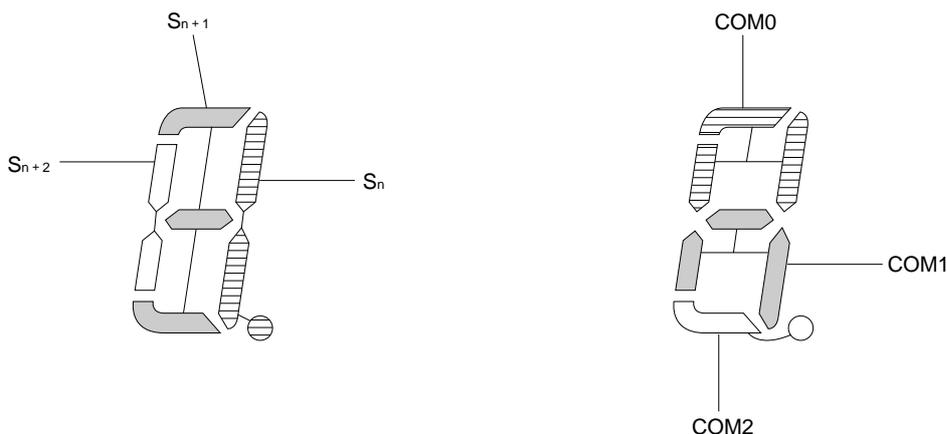
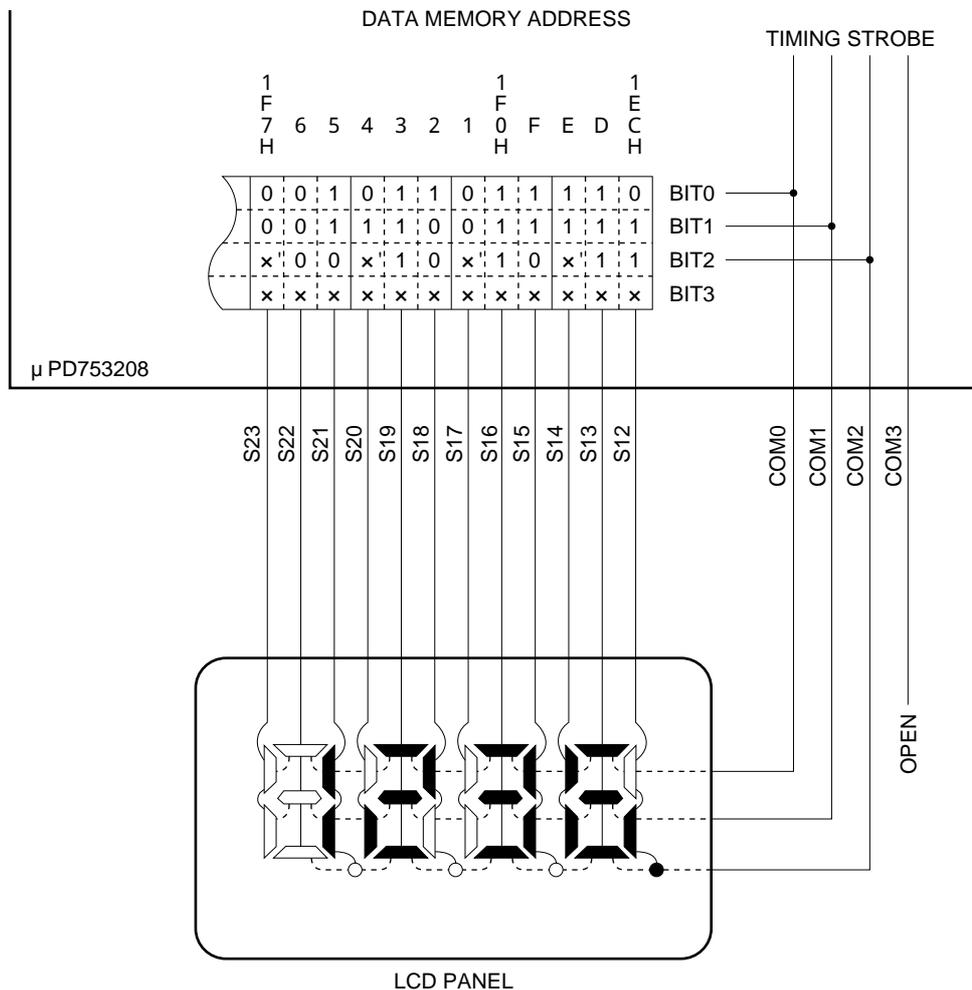


図5 - 116 3時分割LCDパネルの結線例



x' : LCDパネルに対応するセグメントがないため、任意のデータをストア可能です。

x : 3時分割表示のため、常に任意のデータをストア可能です。

図5 - 117 3時分割LCD駆動波形例(1/2バイアス法)

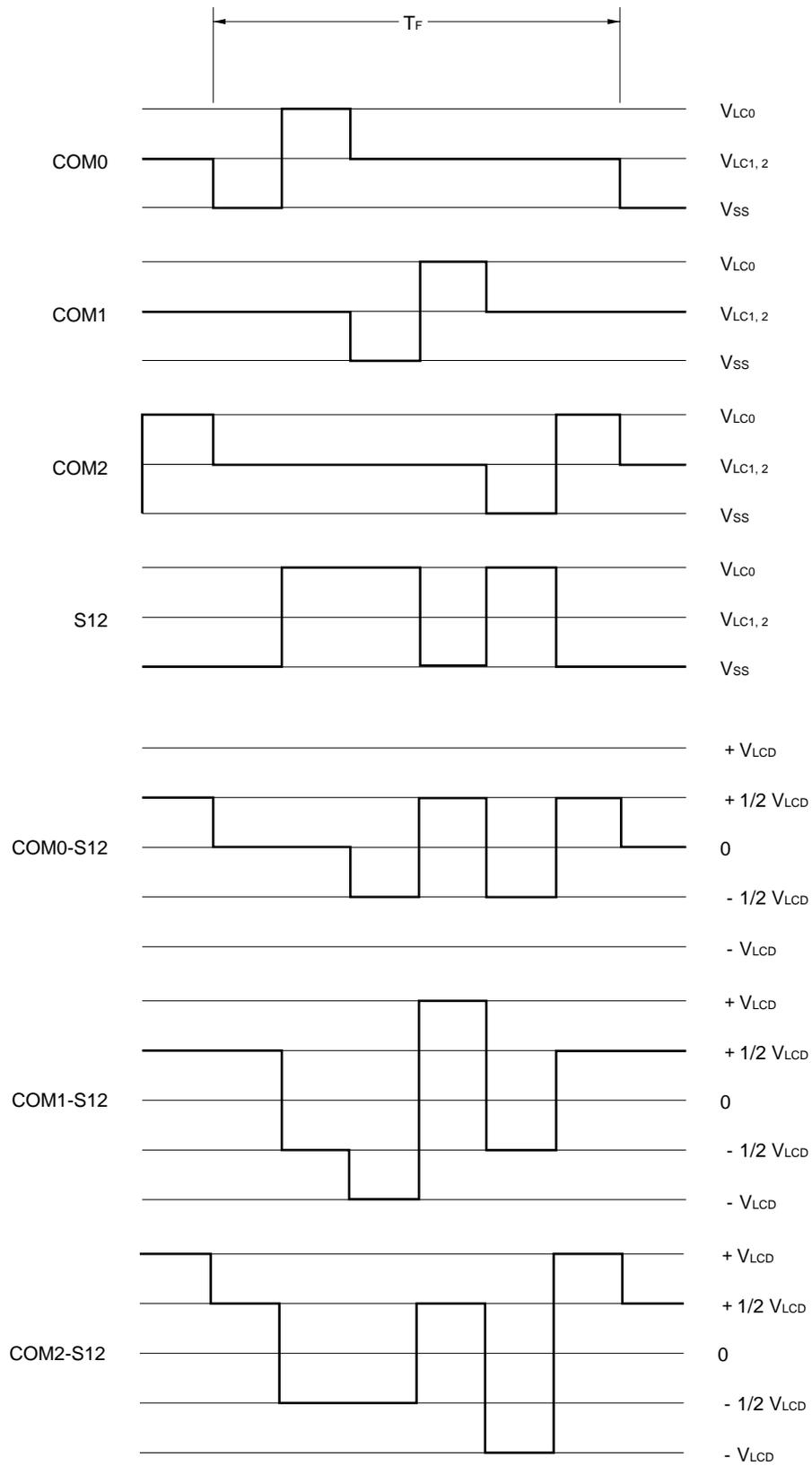
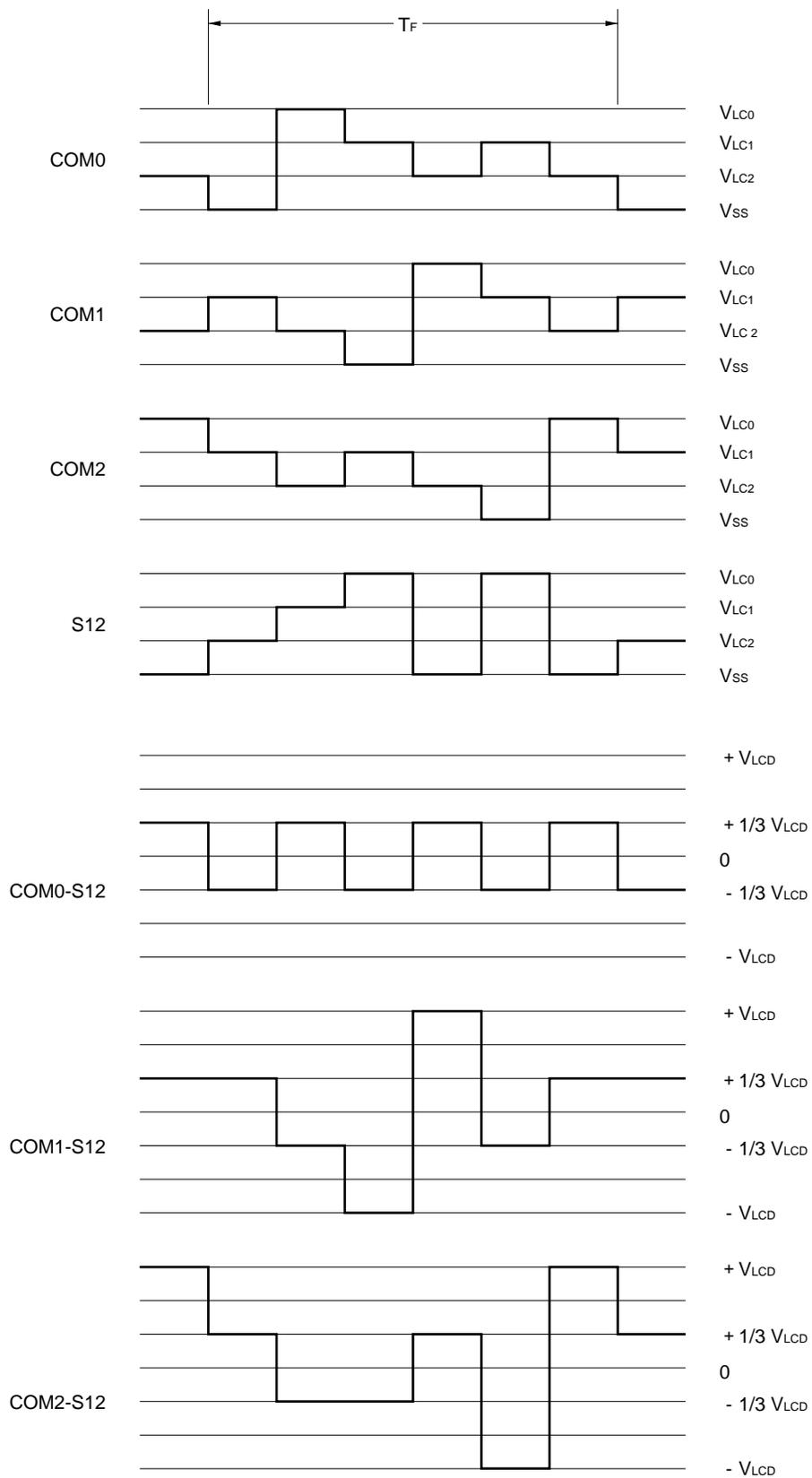


図5 - 118 3時分割LCD駆動波形例(1/3バイアス法)



(4) 4時分割表示例

図5 - 120は、図5 - 119の表示パターンを持つ4時分割方式の6桁LCDパネルとμPD753208のセグメント信号(S12-S23)およびコモン信号(COM0-COM3)との接続を示します。表示例は12345.6で、表示データ・メモリ(1ECH-1F7H番地)の内容はこれに対応しています。

ここでは2桁目の5.(5.)を例にとって説明します。図5 - 119の表示パターンに従って、COM0-COM3の各コモン信号のタイミングで表5 - 22に示すような選択、非選択電圧をS14, S15端子に出力する必要があります。

表5 - 22 S14, S15端子の選択, 非選択電圧(4時分割表示例)

セグメント コモン	S14	S15
COM0	選	選
COM1	非	選
COM2	選	非
COM3	選	選

これによりS14に対応する表示データ・メモリ(1EEH番地)には、1101を用意すればよいことが分かります。

S14とCOM0, COM1信号間とのLCD駆動波形を図5 - 121に示します(図面の都合でCOM2, COM3に対する波形は省略)。COM0の選択タイミングでS14が選択電圧になるときに、LCD点灯レベルである+V_{Lcd}/ - V_{Lcd}の交流矩形波が発生されることが分かります。

図5 - 119 4時分割方式LCDの表示パターンと電極結線

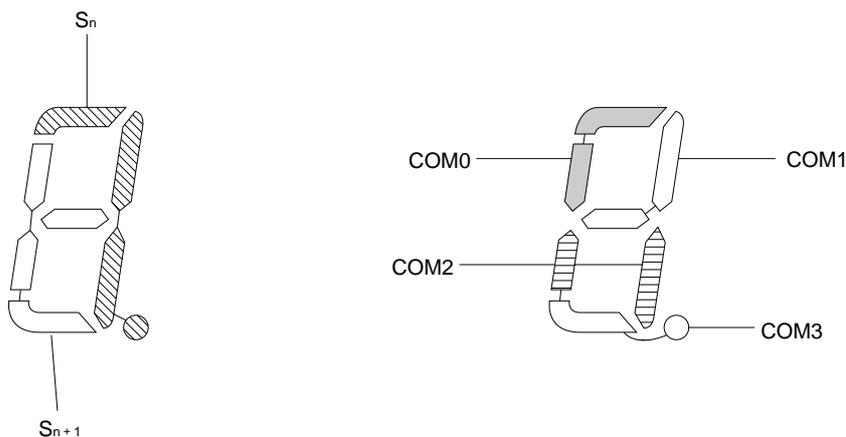


図5 - 120 4時分割LCDパネルの結線例

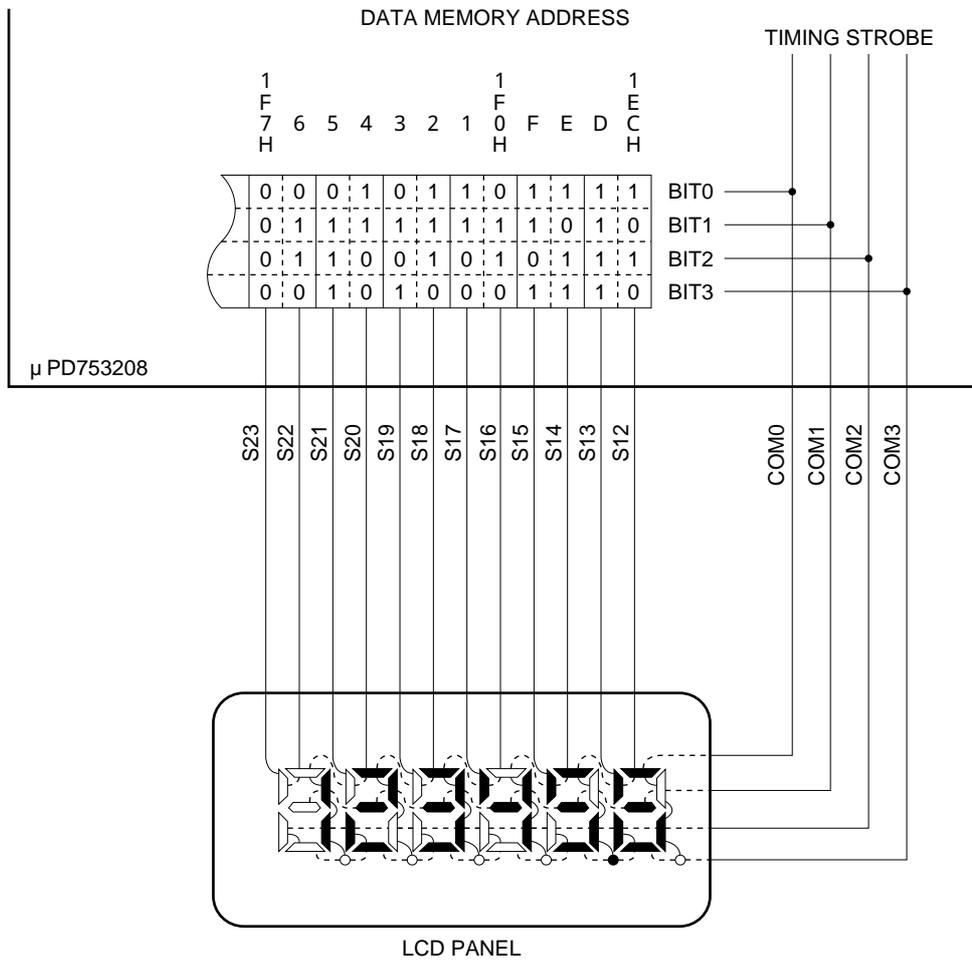
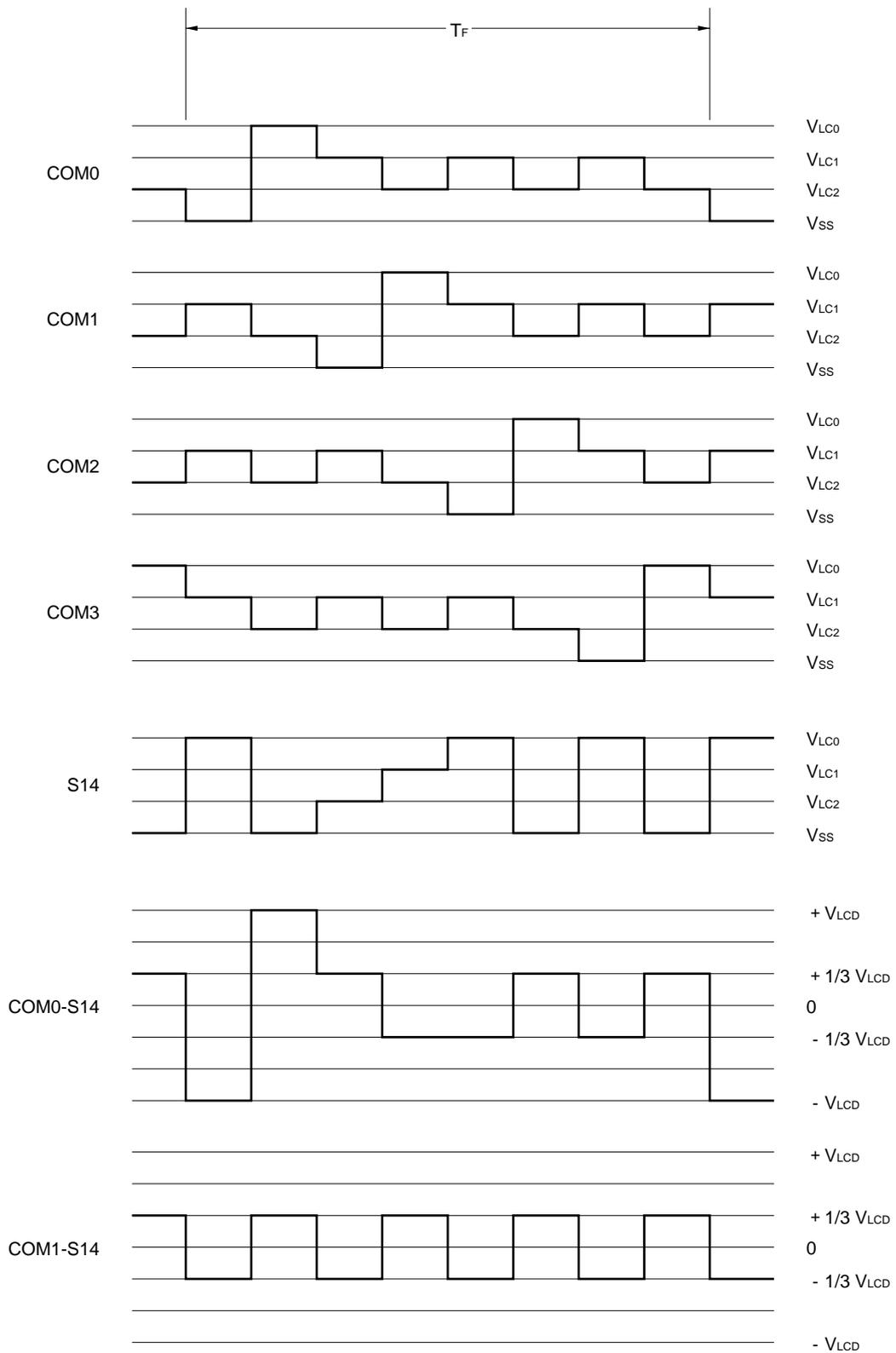


図5 - 121 4時分割LCD駆動波形例(1/3バイアス法)

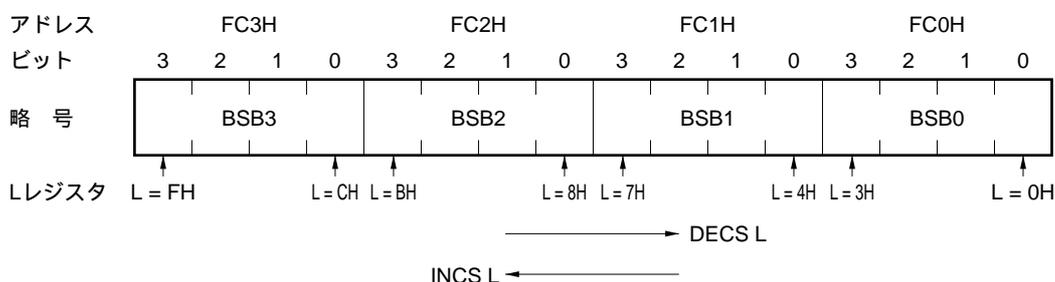


5.8 ビット・シーケンシャル・バッファ.....16ビット

ビット・シーケンシャル・バッファ (BSB) は、ビット操作用の特殊データ・メモリで、特にアドレスおよびビット指定を順次変更してビット操作が容易にできるので、ビット長の長いデータをビット単位で処理するときに便利です。

このデータ・メモリは16ビット構成になっており、ビット操作命令のpmem. @Lアドレッシングが可能で、Lレジスタにより間接ビット指定ができます。この場合、Lレジスタをプログラム・ループ内でインクリメント/デクリメントするだけで順次指定ビットを移動させながら処理を進めることができます。

図5 - 122 ビット・シーケンシャル・バッファのフォーマット



備考1 .pmem. @Lアドレッシングでは、Lレジスタに対応して指定ビットが移動します。

2 .pmem. @Lアドレッシングでは、MBE, MBSの指定にかかわらずに、いつでもBSBを操作できます。

直接アドレッシングでもデータ操作が可能です。1ビット、4ビット、8ビットの直接アドレッシングとpmem. @Lアドレッシングとを合わせて、1ビット・データの連続入力や連続出力などに応用できます。なお、8ビット操作の場合は、BSB0, BSB2を指定して上位、下位8ビットずつを操作します。

例 BUFF1, 2の16ビット・データをポート3のビット0からシリアルに出力する。

```
CLR1  MBE
MOV   XA, BUFF1
MOV   BSB0, XA    ; BSB0, 1をセット
MOV   XA, BUFF2
MOV   BSB2, XA    ; BSB2, 3をセット
MOV   L, #0
LOOP0: SKT   BSB0, @L    ; BSBの指定ビットをテスト
BR    LOOP1
NOP                                ; ダミー (タイミング調整)
SET1  PORT3.0    ; ポート3のビット0をセット
BR    LOOP2
LOOP1: CLR1  PORT3.0    ; ポート3のビット0をクリア
NOP                                ; ダミー (タイミング調整)
NOP
LOOP2: INCS  L          ; L L+1
BR    LOOP0
RET
```

〔メモ〕

第6章 割り込み機能とテスト機能

μPD753208には、7種類のベクタ割り込みソースと、2種類のテスト入力があり多彩な応用が可能です。また、μPD753208の割り込み制御回路には次のような特色があり、非常に高速な割り込み処理が可能となります。

6.1 割り込み制御回路の構成

割り込み制御回路は図6-1のように構成されており、各ハードウェアはデータ・メモリ空間にマッピングされています。

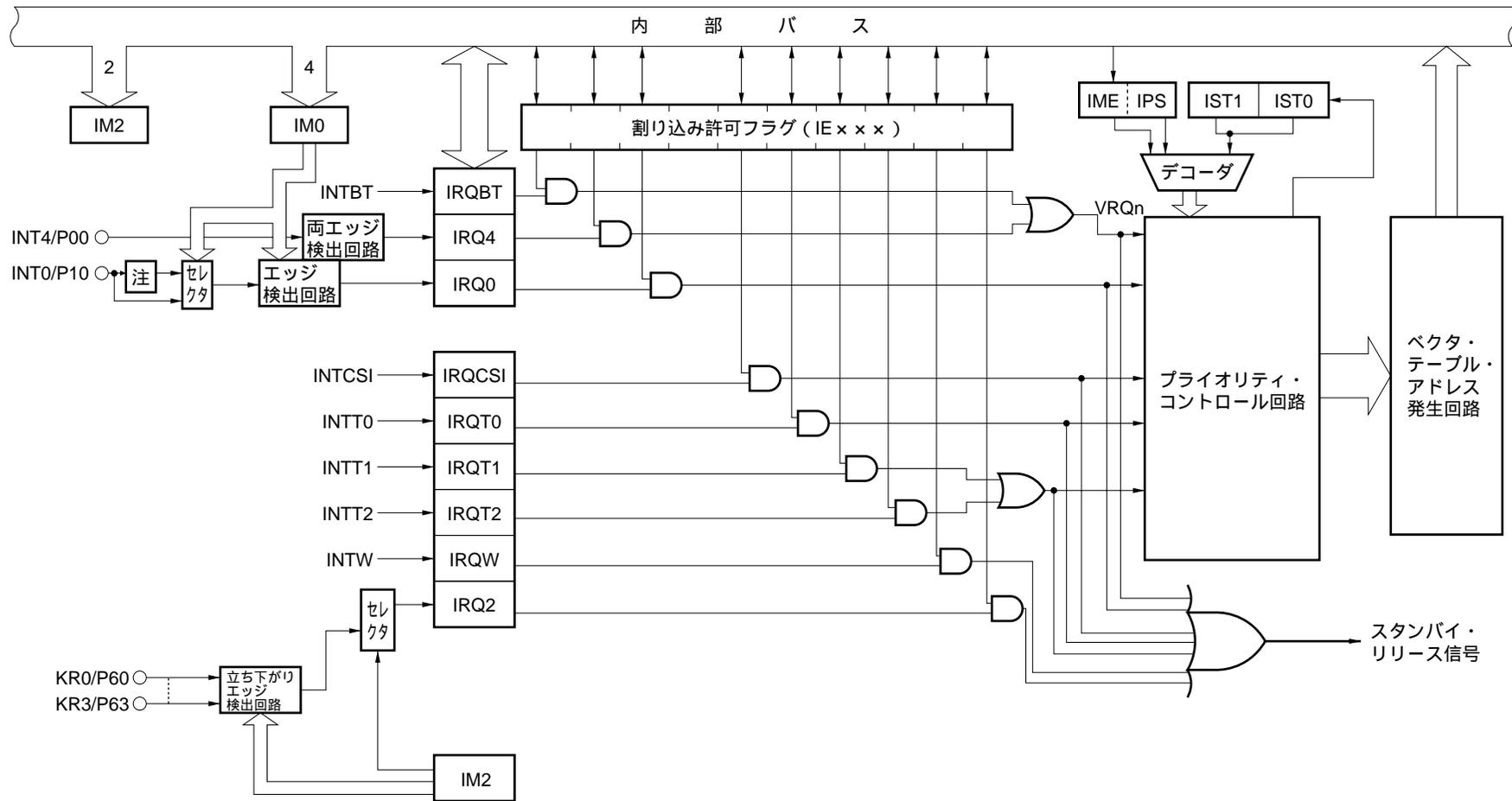
(1) 割り込み機能

- (a) 割り込み許可フラグ (IE × × ×) と、割り込みマスタ許可フラグ (IME) により受け付け可否を制御できるハードウェア制御のベクタ割り込み機能。
- (b) 割り込みスタート・アドレスを任意に設定可能。
- (c) 割り込みプライオリティ選択レジスタ (IPS) により優先順位が指定できる多重割り込み機能。
- (d) 割り込み要求フラグ (IRQ × × ×) のテスト機能 (ソフトウェアで割り込み発生の確認可能)。
- (e) スタンバイ・モードの解除 (割り込み許可フラグにより、解除する割り込みの選択可能)。

(2) テスト機能

- (a) ソフトウェアでテスト要求フラグ (IRQ × × ×) 発生の確認可能。
- (b) スタンバイ・モードの解除 (テスト許可フラグにより、解除するテスト・ソースの選択可能)。

図 6 - 1 割り込み制御回路ブロック図



注 ノイズ除去回路（ノイズ除去回路選択時はスタンバイ・リリース不可）

6.2 割り込みソースの種類とベクタ・テーブル

μPD753208には、次の7種類の割り込みソースがあり、ソフトウェア制御による多重割り込みが可能です。

表6-1 割り込みソースの種類

割り込みソース	内/外	割り込み優先順位 ^注	ベクタ割り込み要求信号 (ベクタ・テーブル・アドレス)
INTBT (ベーシック・インターバル・タイマ/ウォッチドッグ・タイマからの基準時間間隔信号)	内	1	VRQ1 (0002H)
INT4 (立ち上がりおよび立ち下がりエッジ検出の両方が有効)	外		
INT0 (立ち上がり, 立ち下がりエッジ選択)	外	2	VRQ2 (0004H)
INTCSI (シリアル・データ転送終了信号)	内	3	VRQ4 (0008H)
INTT0 (タイマ/イベント・カウンタ0のカウンタ・レジスタとモジュロ・レジスタとの一致信号)	内	4	VRQ5 (000AH)
INTT1 (タイマ・カウンタ1のカウンタ・レジスタとモジュロ・レジスタとの一致信号)	内	5	VRQ6 (000CH)
INTT2 (タイマ・カウンタ2のカウンタ・レジスタとモジュロ・レジスタとの一致信号)	内		

注 割り込み優先順位は、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です。

図6-2 割り込みベクタ・テーブル

0 0 0 2 H	MBE	RBE	INTBT/INT4	スタート・アドレス (上位6ビット)
				" (下位8ビット)
0 0 0 4 H	MBE	RBE	INT0	スタート・アドレス (上位6ビット)
				" (下位8ビット)
0 0 0 6 H	X			
0 0 0 8 H	MBE	RBE	INTCSI	スタート・アドレス (上位6ビット)
				" (下位8ビット)
0 0 0 A H	MBE	RBE	INTT0	スタート・アドレス (上位6ビット)
				" (下位8ビット)
0 0 0 C H	MBE	RBE	INTT1, INTT2	スタート・アドレス (上位6ビット)
				" (下位8ビット)

表6-1の中で、割り込み優先順位と書かれているのは、複数の割り込み要求が同時に発生した場合や、複数の割り込み要求が保留されていた場合に、割り込みが実行される順位を示します。

ベクタ・テーブルには割り込み処理の開始アドレスと、割り込み処理中のMBE, RBEの設定値を書き込みます。ベクタ・テーブルの設定は、アセンブラ疑似命令 (VENTn: n = 1-6) によって行います。

例 INTBT/INT4のベクタ・テーブルの設定

VENT1 MBE = 0, RBE = 0, GOTOBT

- アドレス0002のベクタ・テーブル
- 割り込み処理ルーチンでのMBEの設定
- 割り込み処理ルーチンでのRBEの設定
- 割り込み処理ルーチンの開始アドレスを示すシンボル

★ **注意** VENTn (n = 1-6) 命令のオペランドに記述した内容 (MBE, RBE, 開始アドレス) は、2n番地のベクタ・テーブル・アドレスに格納されます。

例 INTBT/INT4とINTT0のベクタ・テーブルの設定

VENT1 MBE = 0, RBE = 0, GOTOBT ; INTBT/INT4スタート・アドレス
 VENT5 MBE = 0, RBE = 1, GOTOT0 ; INTT0スタート・アドレス

6.3 割り込み機能を制御する各種ハードウェア

(1) 割り込み要求フラグ、割り込み許可フラグ

割り込み要求フラグ (IRQ x x x) には、割り込みソースに対応して次に示す7つがあります。

INT0割り込み要求フラグ (IRQ0) シリアル・インタフェース割り込み要求フラグ (IRQCSI)
 タイマ/イベント・カウンタ0割り込み要求フラグ (IRQT0)
 INT4割り込み要求フラグ (IRQ4) タイマ・カウンタ1割り込み要求フラグ (IRQT1)
 BT割り込み要求フラグ (IRQBT) タイマ・カウンタ2割り込み要求フラグ (IRQT2)

割り込み要求フラグは、割り込み要求発生で“1”にセットされ、割り込み処理が実行されると自動的に“0”にクリアされます。ただし、IRQBTとIRQ4、およびIRQT1とIRQT2はベクタ・アドレスを共用しているのでクリア動作が異なります(6.6 ベクタ・アドレス共用割り込み処理参照)。

割り込み許可フラグ (IE x x x) には、割り込み要求フラグに対応して次に示す7つがあります。

INT0割り込み許可フラグ (IE0) シリアル・インタフェース割り込み許可フラグ (IECSI)
 タイマ/イベント・カウンタ0割り込み許可フラグ (IET0)
 INT4割り込み許可フラグ (IE4) タイマ・カウンタ1割り込み許可フラグ (IET1)
 BT割り込み許可フラグ (IEBT) タイマ・カウンタ2割り込み許可フラグ (IET2)

割り込み許可フラグは、内容が“1”のとき割り込みを許可し、“0”のとき禁止します。

割り込み要求フラグがセットされ、割り込み許可フラグが割り込みを許可している場合、ベクタ割り込み要求 (VRQn) が発生します。この信号はスタンバイ・モードの解除にも使用されます。

割り込み要求フラグと、割り込み許可フラグは、ビット操作命令、および4ビット・メモリ操作命令により操作します。ビット操作命令の場合は、MBEの設定にかかわらず常に直接操作できます。また、割り込み許可フラグはEI IE x x x命令、DI IE x x x命令で操作します。割り込み要求フラグのテストには通常SKTCLR命令を使用します。

例 EI IE0 ; INT0を許可
 DI IET1 ; INTT1を禁止
 SKTCLR IRQCSI ; IRQCSIが1ならばスキップしてクリアする。

割り込み要求フラグを命令によりセットすると、割り込みが発生していなくても発生した場合と同様にベクタ割り込みが実行されます。

$\overline{\text{RESET}}$ 信号発生により、割り込み要求フラグと割り込み許可フラグは“0”にクリアされ、すべての割り込みは禁止されます。

表 6 - 2 割り込み要求フラグのセット信号

割り込み要求フラグ	割り込み要求フラグのセット信号	割り込み許可フラグ
IRQBT	ベーシック・インターバル・タイマ/ウォッチドッグ・タイマによる基準時間間隔信号でセット。	IEBT
IRQ4	INT4/P00端子入力信号の立ち上がり, 立ち下がりどちらのエッジ検出によってもセット。	IE4
IRQ0	INT0/P10端子入力信号のエッジ検出によりセット。検出エッジはINT0エッジ検出モード・レジスタ (IM0) により選択。	IE0
IRQCSI	シリアル・インタフェースのシリアル・データ転送動作終了信号によりセット。	IECSI
IRQT0	タイマ/イベント・カウンタ0からの一致信号でセット。	IET0
IRQT1	タイマ・カウンタ1からの一致信号でセット。	IET1
IRQT2	タイマ・カウンタ2からの一致信号でセット。	IET2

(2) 割り込みプライオリティ選択レジスタ (IPS)

割り込みプライオリティ選択レジスタは、多重割り込みが可能な高位の割り込みを選択するレジスタで、下位3ビットにより指定します。

ビット3は、すべての割り込みを禁止するか否かを指定する、割り込みマスタ許可フラグ (IME) となっています。

IPSは4ビット・メモリ操作命令で設定されますが、ビット3はEI/DI命令により、セット/リセットされます。

IPSの下位3ビットの内容を変更する場合は、割り込みが禁止されている (IME = 0) 状態で行う必要があります。

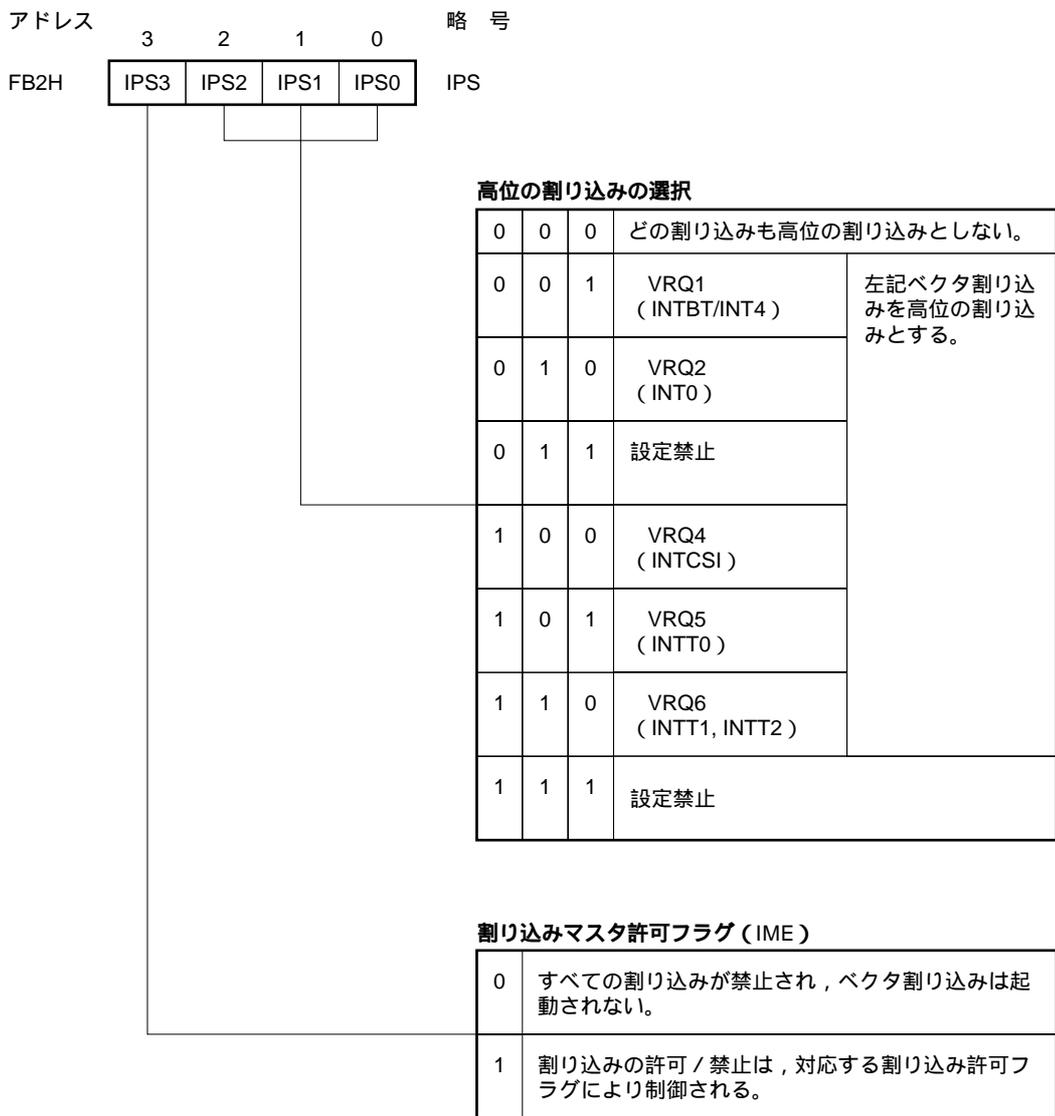
```

例  DI                ; 割り込み禁止
    CLR1   MBE
    MOV    A, #1010B
    MOV    IPS, A      ; INT0を高位の割り込みにし、割り込みを許可する

```

RESET信号発生により全ビットが“0”にクリアされます。

図6-3 割り込みプライオリティ選択レジスタ



(3) INT0, INT4のハードウェア

(a) INT0の構成を図6-4(a)に示します。検出エッジを立ち上がりとするか、立ち下がりとするかを選択できる外部割り込み入力となっています。

また、INT0は、サンプリング・クロックによるノイズ除去機能を持っています(図6-5 ノイズ除去回路の入出力タイミング参照)。ノイズ除去回路によって、サンプリング・クロックの2周期分^注より幅の狭いパルスはノイズとして除去されます。ただし、サンプリング・クロックの1周期分以上のパルスは、サンプリングを行うタイミングによっては、割り込み信号として受け付けられる場合もあります(図6-5 (a)参照)。サンプリング・クロックの2周期分以上のパルスは確実に割り込み信号として受け付けることができます。

INT0は、サンプリング・クロックとして $f_x/64$ の2つを持っており、どちらかを選択して使用できるようになっています。その選択はINT0エッジ検出モード・レジスタ(IM0)のビット3(IM03)で行います(図6-6(a)参照)。

また、検出エッジの選択は、INT0エッジ検出モード・レジスタ(IM0)のビット0, 1(IM00, IM01)で行います。

IM0のフォーマットを図6-6(a)に示します。IM0は4ビット操作命令によって設定します。リセット信号の発生により全ビットが“0”にクリアされ、立ち上がりエッジ指定となります。

注 サンプリング・クロックが のとき : $2t_{cy}$
サンプリング・クロックが $f_x/64$ のとき : $128/f_x$

注意1 . INT0/P10端子は、ポートとして入力する場合にも、ノイズ除去回路を通して入力されますので、サンプリング・クロックの2周期分以上のパルスを入力してください。

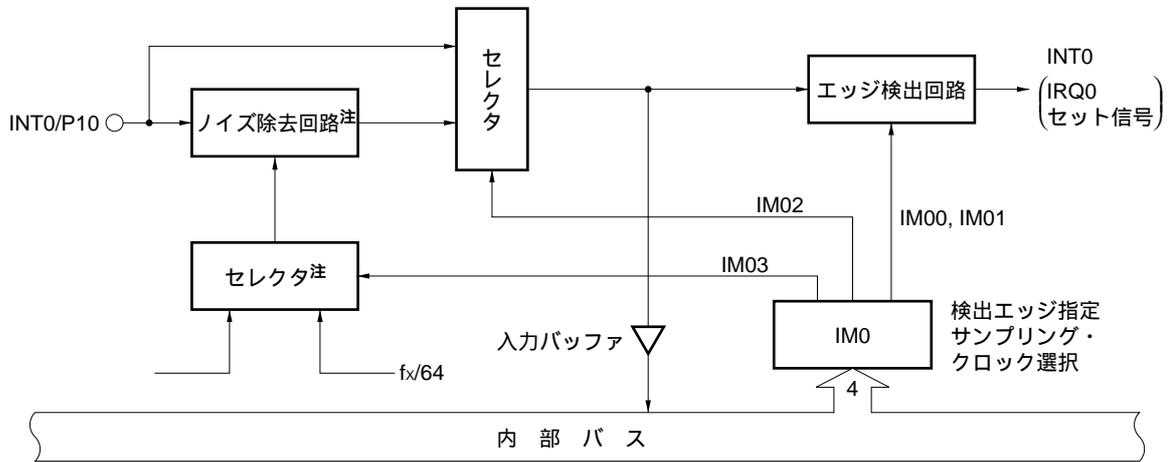
2 . ノイズ除去回路を選択する(IM02 = 0 に設定する)と、INT0はクロックによるサンプリングを行うため、スタンバイ・モード時は動作しません(ノイズ除去回路はCPUクロック が供給されていないときは動作しません)。したがって、スタンバイ・モード時にINT0によるスタンバイ・リリースを行う必要がある場合は、ノイズ除去回路を選択しないでください(IM02 = 1 に設定してください)。

★

(b) INT4の構成を図6-4(b)に示します。立ち上がり、立ち下がりの両エッジを検出できる外部割り込み入力となっています。

図6-4 INT0, INT4の構成

(a) INT0のハードウェア



★ 注 fx/64を選択しても、INT0によるHALTモードの解除はできません。

(b) INT4のハードウェア

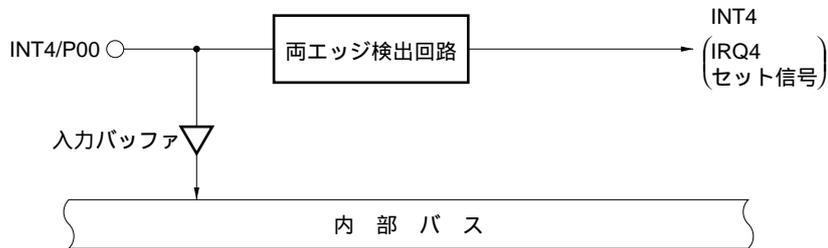
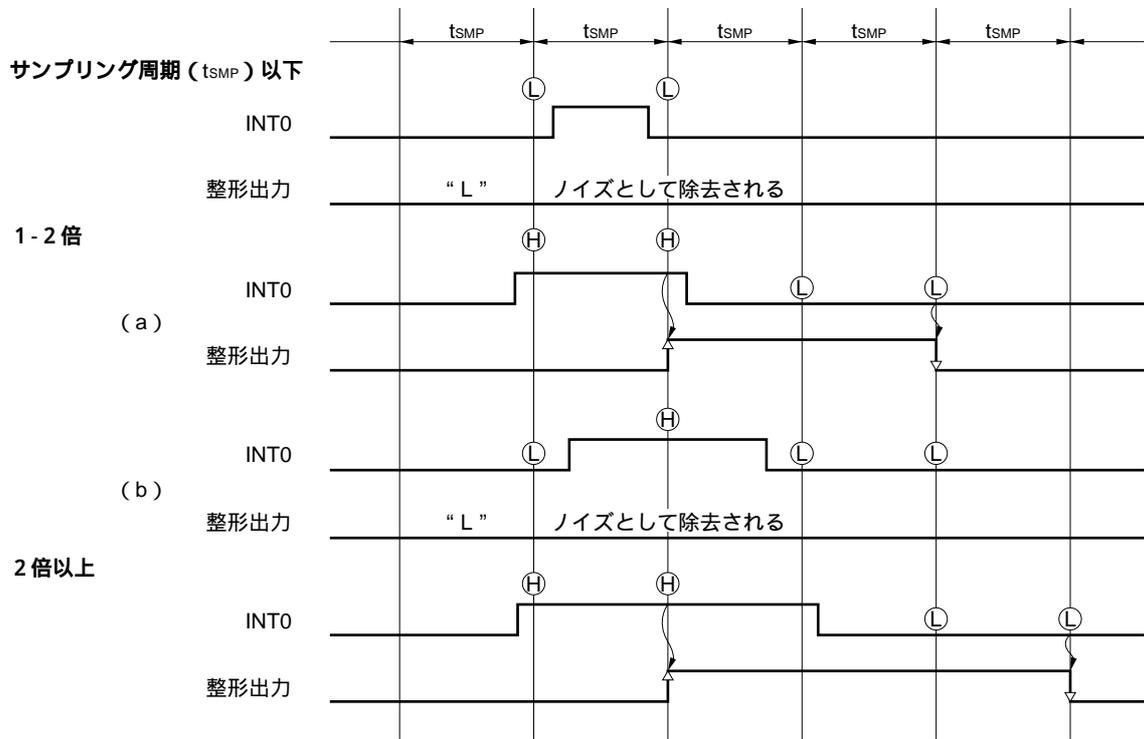
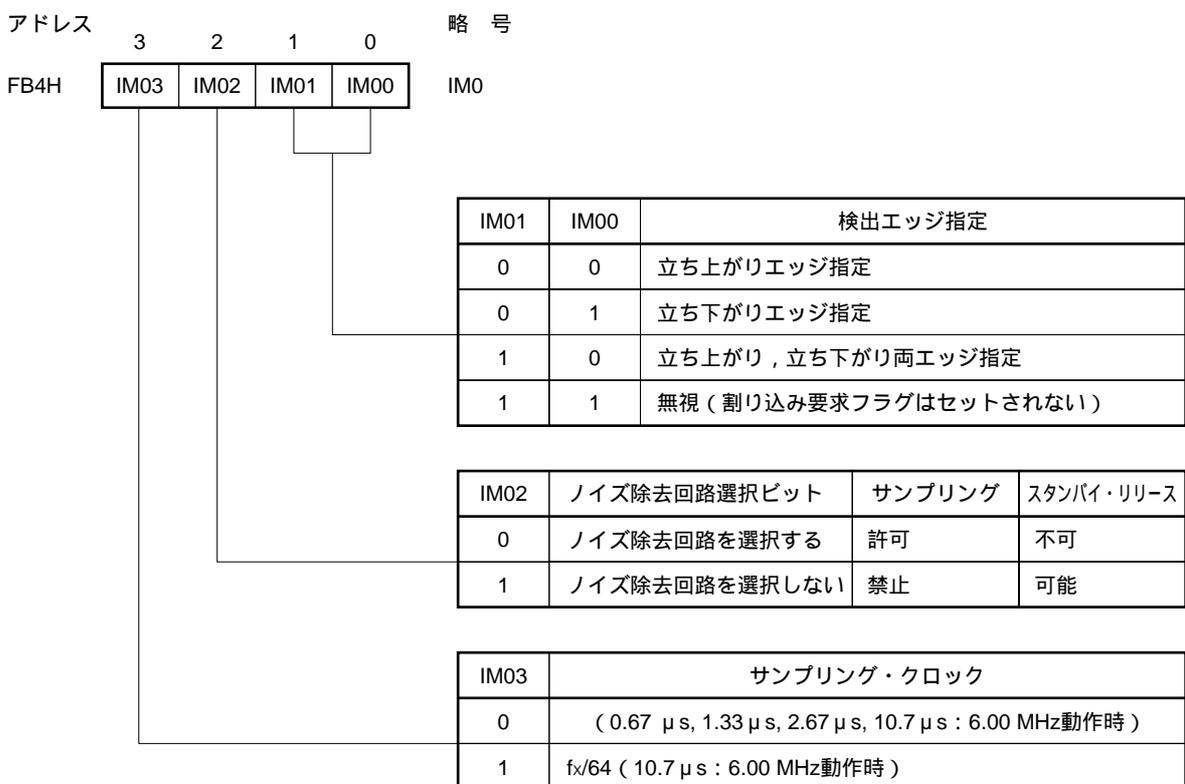


図6 - 5 ノイズ除去回路の入出力タイミング



備考 $t_{SMP} = t_{CY}$ または $64/f_x$

図6-6 INTOエッジ検出モード・レジスタ (IM0) のフォーマット



注意 エッジ検出モード・レジスタを変更すると、割り込み要求フラグがセットされる場合がありますので、あらかじめ割り込みを禁止してモード・レジスタを変更し、CLR1命令によって割り込み要求フラグをクリアしてから、割り込みを許可してください。なお、IM0の変更で、サンプリング・クロックとして $f_x/64$ を選択した場合には、モード・レジスタ変更後16マシン・サイクル経過してから、割り込み要求フラグをクリアする必要があります。

(4) 割り込みステータス・フラグ

割り込みステータス・フラグ (IST0, IST1) はCPUが現在実行中の処理のステータスを示すフラグでPSWに含まれています。

割り込みプライオリティ・コントロール回路は、このフラグの内容によって表6-3に示されるように、多重割り込みの制御を行います。

IST0, IST1は4ビット操作命令またはビット操作命令により変更することができるため、実行中のステータスを変えて多重割り込みを行うこともできます。IST0, IST1をビット単位で操作する場合は、MBEの設定に関係なくいつでも行うことができます。

なお、IST0, IST1を操作する場合は、操作前に必ずDI命令を実行して割り込みを禁止し、操作後にEI命令を実行して割り込みを許可するようにしてください。

IST1, IST0は、割り込み受け付け時にほかのPSWとともにスタック・メモリに退避されたのち、自動的に1つ上位のステータスに変更されます。RETI命令が実行されると、もとのIST1, IST0の値が復帰します。

RESET信号の発生によりフラグの内容は“0”にクリアされます。

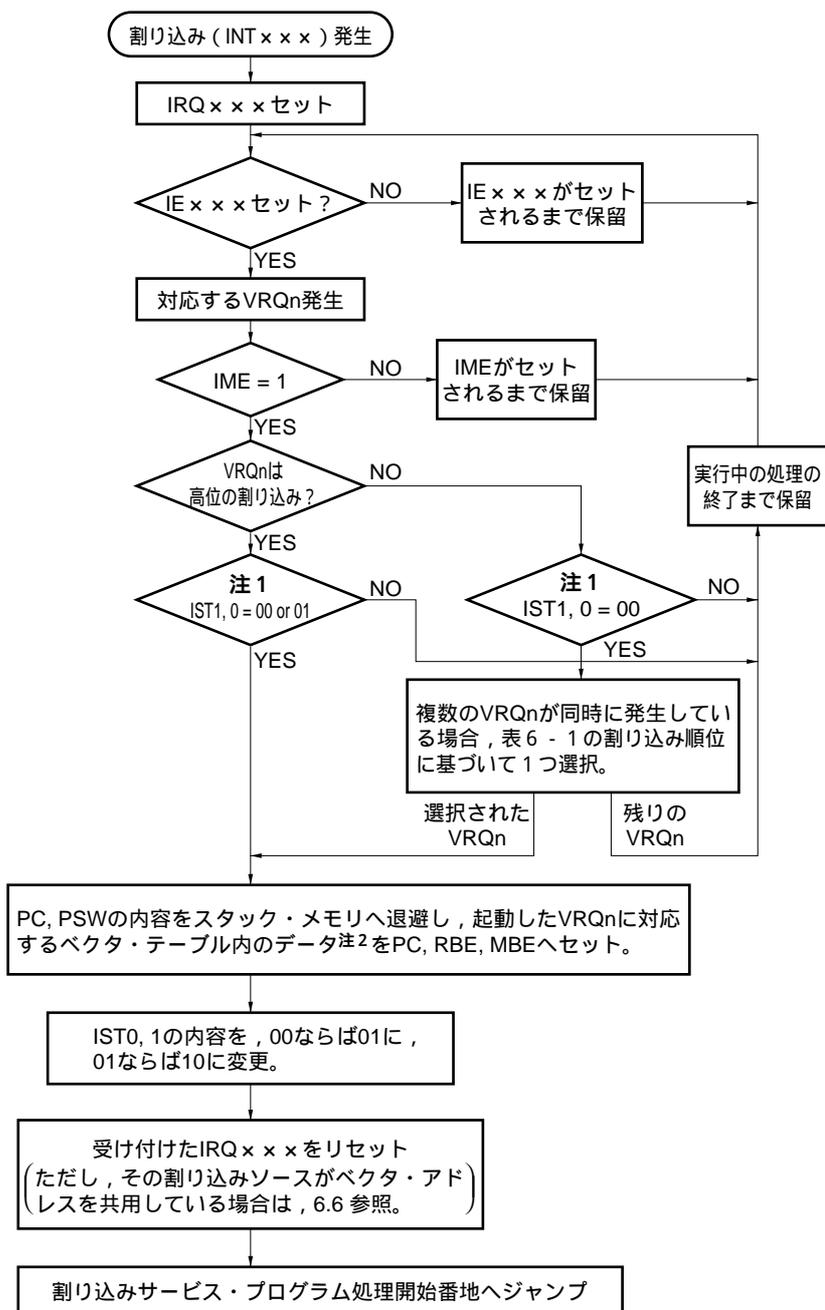
表6-3 IST1, IST0と割り込み処理状態

IST1	IST0	実行中の処理のステータス	CPUの処理内容	受け付け可能な割り込み要求	割り込み受け付け後	
					IST1	IST0
0	0	ステータス0	通常のプログラム処理中	すべての割り込みを受け付け可能	0	1
0	1	ステータス1	低位の割り込み、または高位の割り込み処理中	高位の割り込みのみ受け付け可能	1	0
1	0	ステータス2	高位の割り込み処理中	すべての割り込みの受け付け禁止	-	-
1	1	設定禁止				

6.4 割り込みシーケンス

割り込みが発生すると、以下の手順で処理されます。

図6-7 割り込み処理シーケンス



注1 . IST1, 0 : 割り込みステータス・フラグ (PSWのビット3, 2 ; 表6-3参照)

2 . 各ベクタ・テーブルには、割り込みサービス・プログラムの開始アドレスと、割り込み開始時のMBE, RBEの設定値をストアしておきます。

6.5 多重割り込み処理制御

μ PD753208は、次の2つの方法により多重割り込みができます。

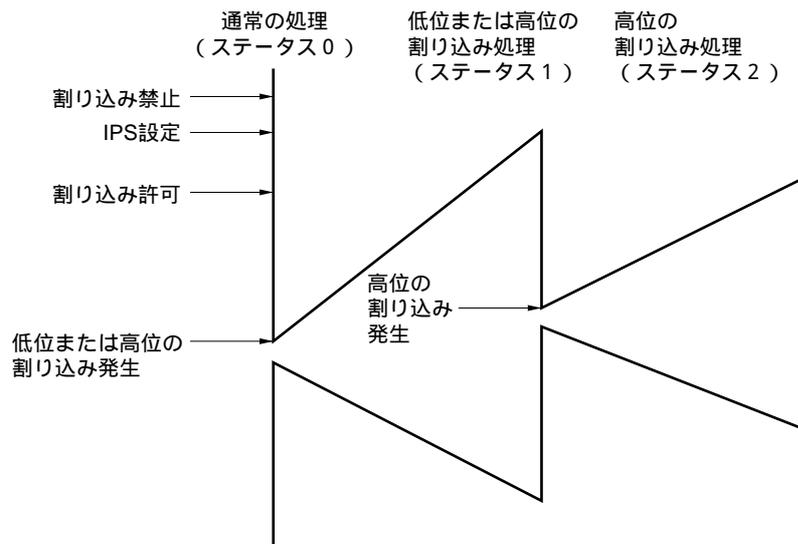
(1) 高位の割り込みを指定する多重割り込み

この方法は、 μ PD753208の標準的な多重割り込み方法で、割り込みソースのうち1つを選び、その割り込みの多重割り込み（二重割り込み）を可能にするものです。

すなわち、割り込みプライオリティ選択レジスタ（IPS）によって指定された高位の割り込みは、実行中の処理のステータスが0および1のときに割り込み可能で、それ以外の割り込み（低位の割り込み）はステータス0のときのみ割り込み可能です（図6-8、表6-3参照）。

したがって、使用する割り込みのうち1つだけを二重割り込みできるようにしたい場合には、この方法を使えば、割り込み処理中の割り込み許可、禁止などの動作を行わなくても二重割り込みを実現し、かつ、ネスティング・レベルを2レベルに抑えることができます。

図6-8 高位の割り込みによる多重割り込み



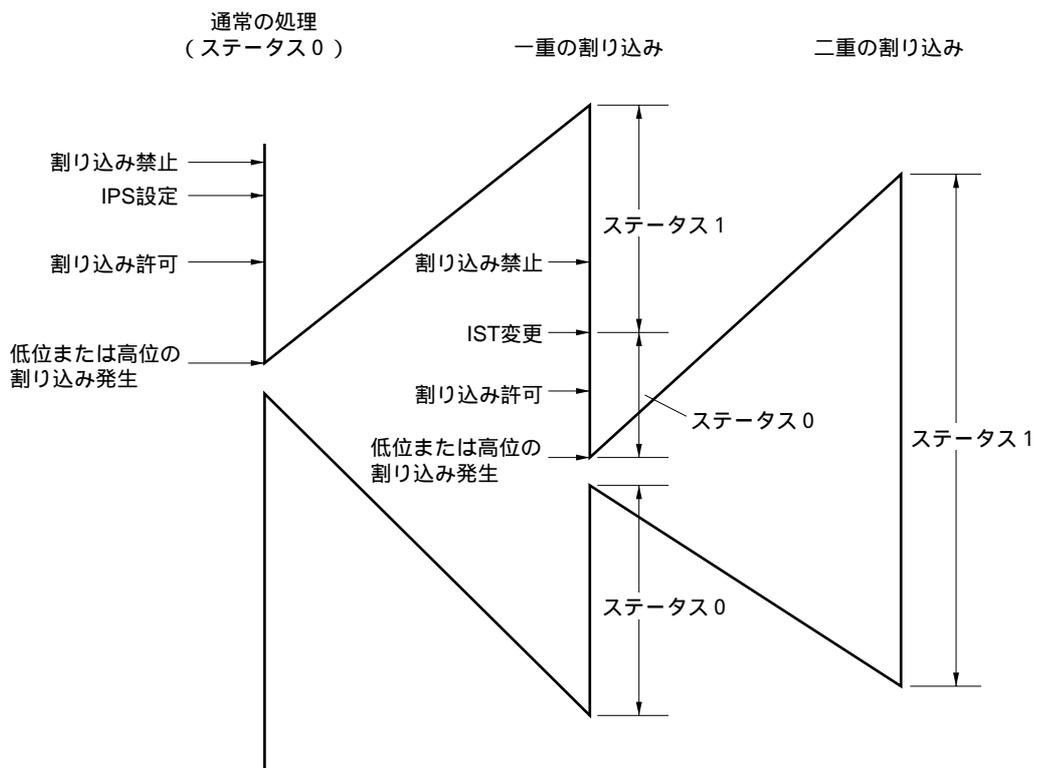
(2) 割り込みステータス・フラグを変更する多重割り込み

割り込みステータス・フラグをプログラムにより変更すれば、多重割り込み可能になります。すなわち、割り込み処理プログラムでIST1, IST0を“0, 0”に変更し、ステータス0にすれば多重割り込みできます。

この方法は2つ以上の割り込みの多重割り込みを許可したい場合、あるいは三重以上の多重割り込みを行いたい場合に使用します。

IST1, IST0の変更は、あらかじめDI命令により割り込みを禁止した状態で行います。

図6-9 割り込みステータス・フラグの変更による多重割り込み



6.6 ベクタ・アドレス共用割り込み処理

INTBTとINT4およびINTT1とINTT2の各割り込みソースはベクタ・テーブルを共用しているため、以下に示すようにして割り込みソースの選択を行います。

(1) 一方の割り込みのみ使用する場合

ベクタ・テーブルを共用する2種類の割り込みソースのうち、必要とする割り込みソースの割り込み許可フラグを“1”にセット、もう一方の割り込み許可フラグを“0”にクリアします。この場合、許可した(IE×××=1)方の割り込みソースにより割り込み要求を発生し、受け付けられるとその割り込み要求フラグはリセットされます。

(2) 両方の割り込みを使用する場合

2種類の割り込みソースに対応する割り込み許可フラグを共に“1”にセットします。この場合、2種類の割り込みソースの割り込み要求フラグの論理和が割り込み要求となります。

この場合、一方あるいは両方の割り込み要求フラグのセットによる割り込み要求が受け付けられてもどちらの割り込み要求フラグもリセットされません。

したがって、この場合は割り込みサービス・ルーチンでどちらの割り込みソースによる割り込みかを判断する必要があります。それは、割り込みサービス・ルーチンの最初にSKTCLR命令で割り込み要求フラグをチェックすることにより行います。

この割り込み要求フラグのテスト&クリアを行ったとき、両方の要求フラグがセットされていると、一方の要求フラグがクリアされても割り込み要求は残ります。この割り込みが「高位の割り込み」として選択されていた場合には、残った割り込み要求により二重割り込み処理に入ります。

すなわち、テストしない方の割り込み要求が先に処理されることとなります。一方、「低位の割り込み」の場合には、残った割り込みは保留されますから、テストした方の割り込み要求が先に処理されます。したがって、表6-4のように、「高位の割り込み」であるか否かによって共用割り込み判別方法は異なります。

表6-4 共用割り込みの判別

「高位の割り込み」の場合	割り込みを禁止し、優先したい割り込み要求フラグをテストする。
「低位の割り込み」の場合	優先したい割り込みソースの割り込み要求フラグをテストする。

例1 . INTBTとINT4の両方を「高位の割り込み」として使用し，INT4を優先する場合。

```

DI
SKTCLR      IRQ4      ; IRQ4 = 1?
BR          VSUBBT
  ⋮
EI
RETI
  ⋮

VSUBBT : CLR1      IRQBT
  ⋮
EI
RETI
  
```

} INT4の処理
} ルーチン

} INTBTの処理
} ルーチン

2 . INTBTとINT4の両方を「低位の割り込み」として使用しINT4を優先する場合。

```

SKTCLR      IRQ4      ; IRQ4 = 1?
BR          VSUBBT
  ⋮
RETI
  ⋮

VSUBBT : CLR1      IRQBT
  ⋮
RETI
  
```

} INT4の処理
} ルーチン

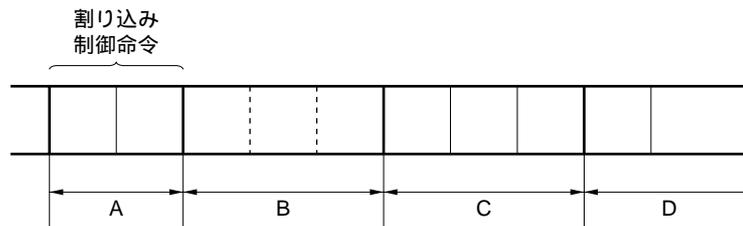
} INTBTの処理
} ルーチン

6.7 割り込み処理までのマシン・サイクル

μPD753208では、割り込み要求フラグ（IRQ×××）がセットされてから割り込みルーチンのプログラムを実行するまでのマシン・サイクルは、次のようになります。

（1）割り込み制御命令実行中にIRQ×××がセットされた場合

割り込み制御命令実行中にIRQ×××がセットされた場合には、続く1命令を実行したのち、3マシン・サイクルの割り込み処理を行ったあと割り込みルーチンのプログラムを実行します。



- A : IRQ×××セット
- B : 続く1命令を実行（1-3マシン・サイクル；命令により異なる）
- C : 割り込み処理（3マシン・サイクル）
- D : 割り込みルーチンの実行

備考1．割り込み制御命令とは、割り込み関係のハードウェア（データ・メモリのFB×H番地）を操作する命令です。この命令には、DI命令、EI命令があります。

2．割り込み処理の3マシン・サイクルは、割り込み受け付け時に操作されるスタックの操作時間などです。

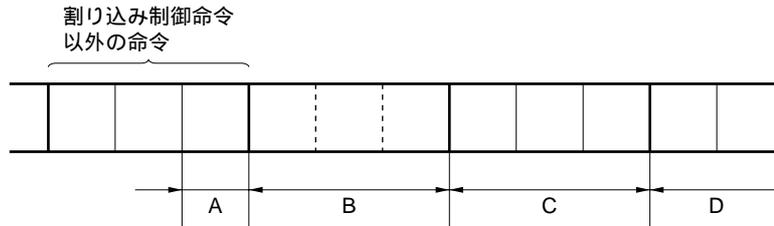
注意1．割り込み制御命令が連続している場合には、最後に実行した割り込み制御命令に続く1命令を実行したのち、3マシン・サイクルの割り込み処理を行ったあと割り込みルーチンのプログラムを実行します。

2．IRQ×××がセットされるタイミング（上図のA）または、そのあとのタイミングでDI命令を実行すると、セットされたIRQ×××の割り込み要求は次にEI命令を実行するまで保留されます。

(2)(1) 以外の命令実行中にIRQxxxがセットされた場合

(a) 実行中の命令の最終マシン・サイクルでIRQxxxがセットされた場合

この場合は、実行中の命令に続く1命令を実行したのち、3マシン・サイクルの割り込み処理を行ったあと割り込みルーチンのプログラムを実行します。

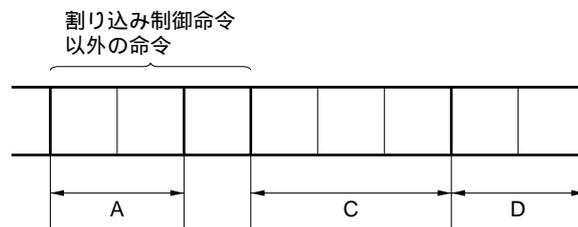


- A : IRQxxxセット
- B : 続く1命令を実行(1-3マシン・サイクル; 命令により異なる)
- C : 割り込み処理(3マシン・サイクル)
- D : 割り込みルーチンの実行

注意 続く命令が割り込み制御命令の場合には、最後に実行した割り込み制御命令に続く1命令を実行したのち、3マシン・サイクルの割り込み処理を行ったあと割り込みルーチンのプログラムを実行します。またIRQxxxがセットされたあとに実行される割り込み制御命令がDI命令の場合には、セットされたIRQxxxの割り込み要求は保留されません。

(b) 実行中の命令の最終マシン・サイクル以前でIRQxxxがセットされた場合

この場合は、実行中の命令ののち、3マシン・サイクルの割り込み処理を行ったあと割り込みルーチンのプログラムを実行します。



- A : IRQxxxセット
- C : 割り込み処理(3マシン・サイクル)
- D : 割り込みルーチンの実行

6.8 割り込みの効果的な使い方

割り込み機能は、次のように使用すると効果的です。

(1) 割り込み処理ルーチンではMBE = 0 とする

割り込み処理ルーチン中で使用するデータ・メモリは優先的に00H-7FH番地に割り付け、割り込みベクタ・テーブルでMBE = 0 を指定すれば、メモリ・バンクを意識しないでプログラムすることができます。

プログラムの都合でやむを得ず、メモリ・バンク 1 を使用しなければならない場合は、PUSH BS命令によってメモリ・バンク選択レジスタを退避して、メモリ・バンク 1 を選択します。

(2) 通常ルーチンと割り込みルーチンでレジスタ・バンクを使い分ける

通常ルーチンではRBE = 1, RBS = 2 としてレジスタ・バンク 2, 3 を使用します。一重の割り込み処理ルーチンではRBE = 0 として、レジスタ・バンク 0 を使用することによってレジスタの退避 / 復帰は一切必要なくなります。二重の割り込み処理ではRBE = 1 として、PUSH BS命令によりレジスタ・バンクを退避し、RBS = 1 としてレジスタ・バンク 1 を使用します。

(3) ソフトウェア割り込みをディバグに使う

割り込み要求フラグを命令によってセットしても、割り込みが発生したときと同じ動作をします。不特定の割り込みのディバグや割り込みが同時に発生した場合のディバグなどは、命令によって、割り込み要求フラグをセットすることによって効率的に行うことができます。

6.9 割り込みの応用

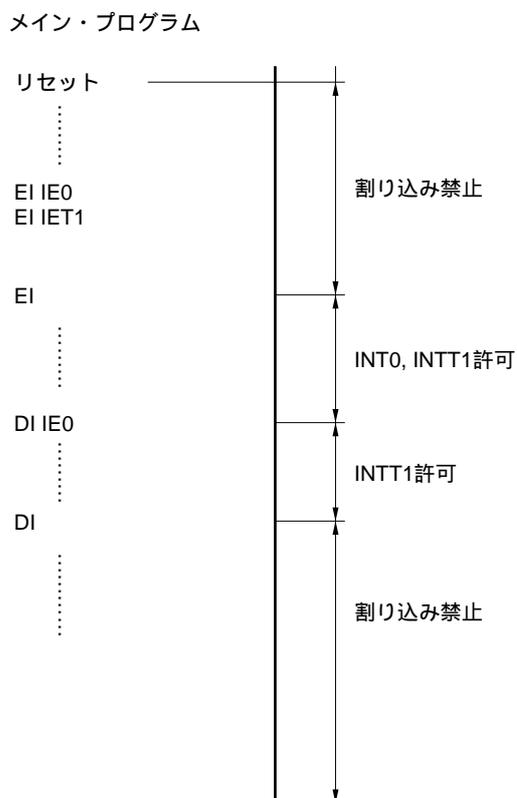
割り込み機能を使用する場合は、まず、メイン・プログラムで次のように設定します。

- (a) 使用する割り込み許可フラグをセット (EI IE × × × 命令)。
- (b) INT0を使用する場合は、アクティブ・エッジを選択 (IM0のセット)。
- (c) 二重割り込み (高位の割り込みによる) を使用する場合は、IPSを設定 (同時にIMEセットもできる)。
- (d) 割り込みマスタ許可フラグをセット (EI命令)。

割り込み処理プログラムでは、ベクタ・テーブルによってMBE, RBEが設定されます。ただし、「高位の割り込み」に指定された割り込みでは、レジスタ・バンクの退避と設定が必要です。

割り込み処理プログラムからの復帰はRETI命令を使用します。

(1) 割り込みの許可・禁止



$\overline{\text{RESET}}$ 信号により，すべての割り込みは禁止。

EI IE × × 命令により，割り込み許可フラグをセット。

この段階ではまだすべての割り込みは禁止。

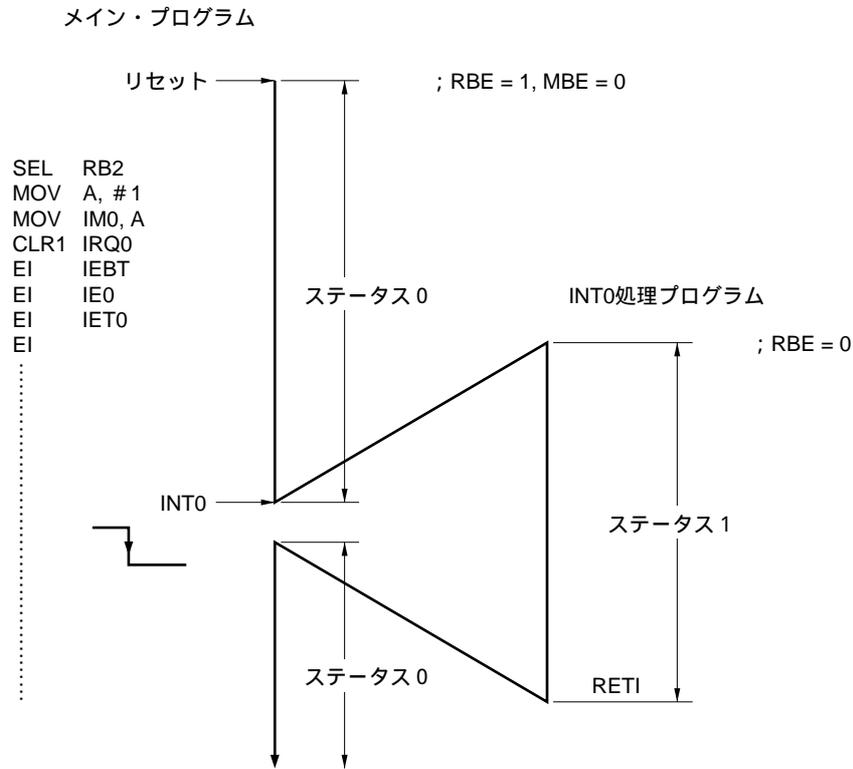
EI命令により，割り込みマスタ許可フラグをセット。

この段階で，INT0，INTT1が許可される。

DI IE × × 命令により，割り込み許可フラグをクリア，INT0が禁止される。

DI命令によりすべての割り込みが禁止。

(2) INTBT, INT0 (立ち下がりエッジ・アクティブ), INTT0を使用する例, 多重割り込みはしない
(すべて低位の割り込み)



$\overline{\text{RESET}}$ 信号により, すべての割り込みが禁止され, ステータス0になる。

RBE = 1 をリセット・ベクタ・テーブルで指定し, SEL RB2命令でレジスタ・バンク 2, 3 を使用。

INT0を立ち下がりエッジ・アクティブに設定。

EI, EI IE × × × 命令により割り込み許可。

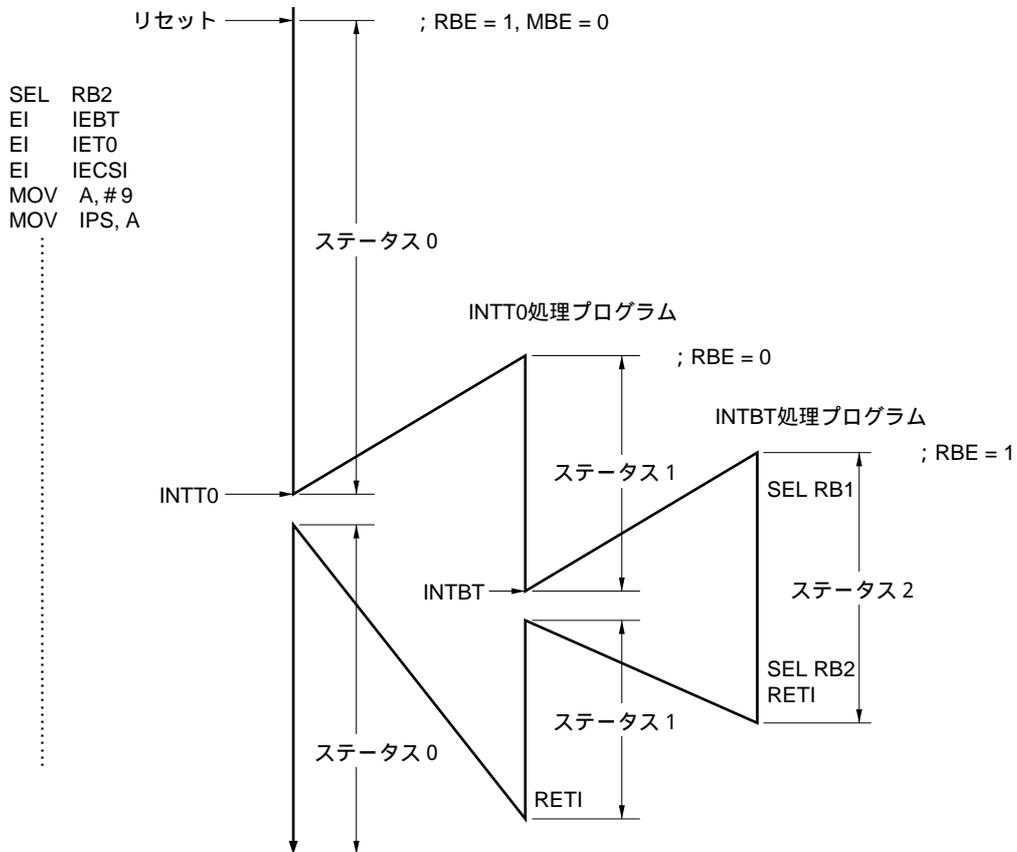
INT0の立ち下がりにより, INT0割り込み処理プログラムをスタート。ステータスは1に変更され, すべての割り込みは禁止される。

RBE = 0 となり, レジスタ・バンク 0, 1 を使用。

RETI命令により割り込みから復帰。ステータスは0に戻り, 割り込みは許可される。

備考 この例のようにすべての割り込みを「低位の割り込み」として使用する場合は, メイン・プログラムではRBE = 1, RBS = 2としてレジスタ・バンク 2, 3 を, 割り込み処理プログラムではRBE = 0としてレジスタ・バンク 0, 1 を使えば, レジスタの退避/復帰はまったく不要となります。

(3) 「高位の割り込み」による多重割り込み (INTBTを高位, INTT0, INTCSIを低位の割り込み)



IPSの設定により, INTBTを「高位の割り込み」とし, 同時に割り込み許可。

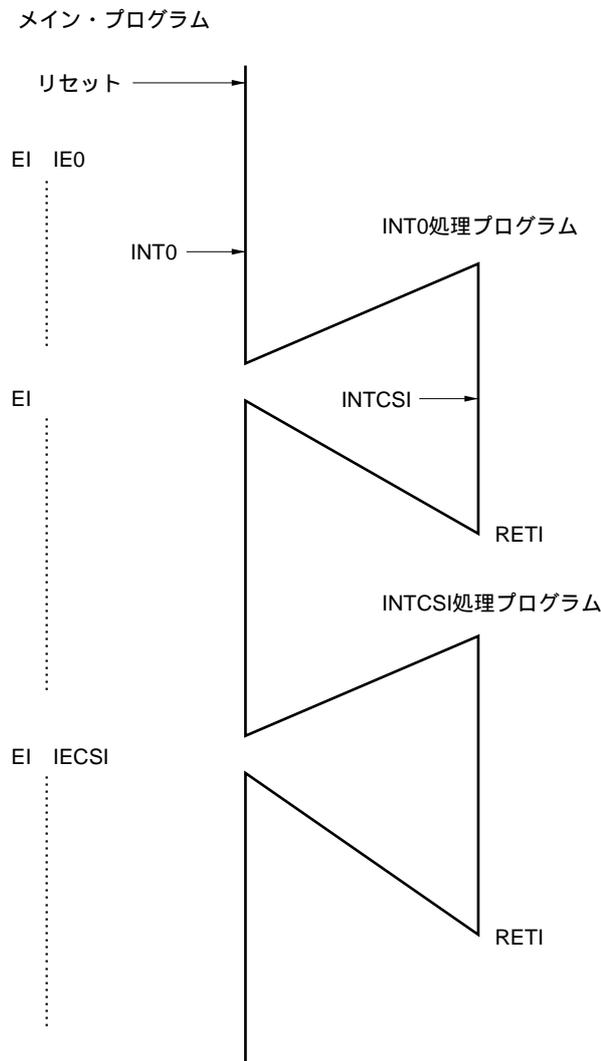
低位の割り込みINTT0発生により, INTT0処理プログラム開始, ステータス 1 となり低位の割り込みは禁止。RBE = 0 としてレジスタ・バンク 0 を使用。

高位の割り込みINTBT発生により, 二重割り込み実行。ステータスとなりすべての割り込み禁止。

RBE = 1, RBS = 1 としてレジスタ・バンク 1 を使用 (使用するレジスタだけをPUSH命令で回避してもよい)。

RBSを 2 に戻し, リターン。ステータスは 1 に戻る。

(4) 保留割り込みの実行——割り込み禁止中の割り込み入力——



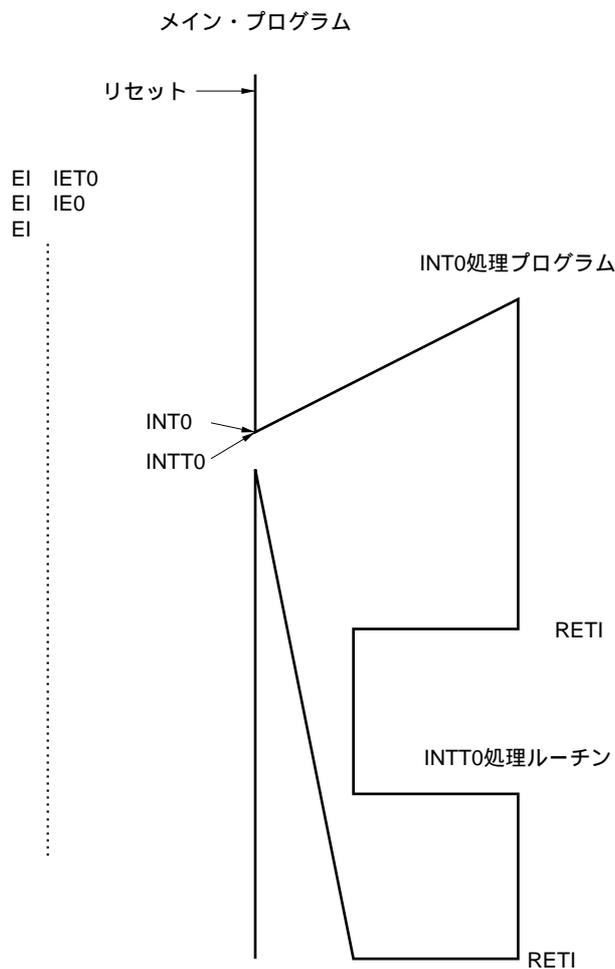
割り込み禁止中にINT0がセットされても要求フラグは保留される。

EI命令で割り込みが許可された時点で、INT0処理プログラム・スタート。

と同様

保留されていたINTCSIが許可された時点で、INTCSI処理プログラム・スタート。

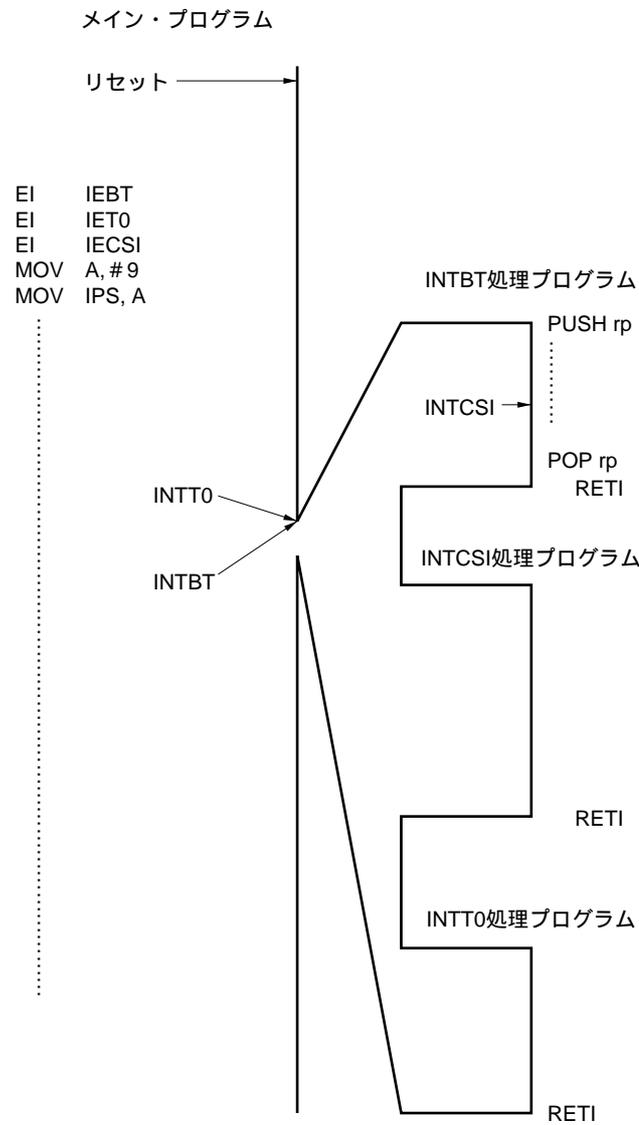
(5) 保留割り込みの実行——低位の割り込みが同時に2つ発生——



「低位の割り込み」INT0とINTT0が同時（同一命令実行中）に発生した場合は、割り込み順位の高いINT0を先に実行（INTT0は保留される）。

RETI命令により、INT0処理プログラムが終わると保留されていたINTT0処理プログラムをスタート。

(6) 保留割り込みの実行——割り込み処理中の割り込み発生 (INTBTを高位, INTT0, INTCSIを低位の割り込み)——



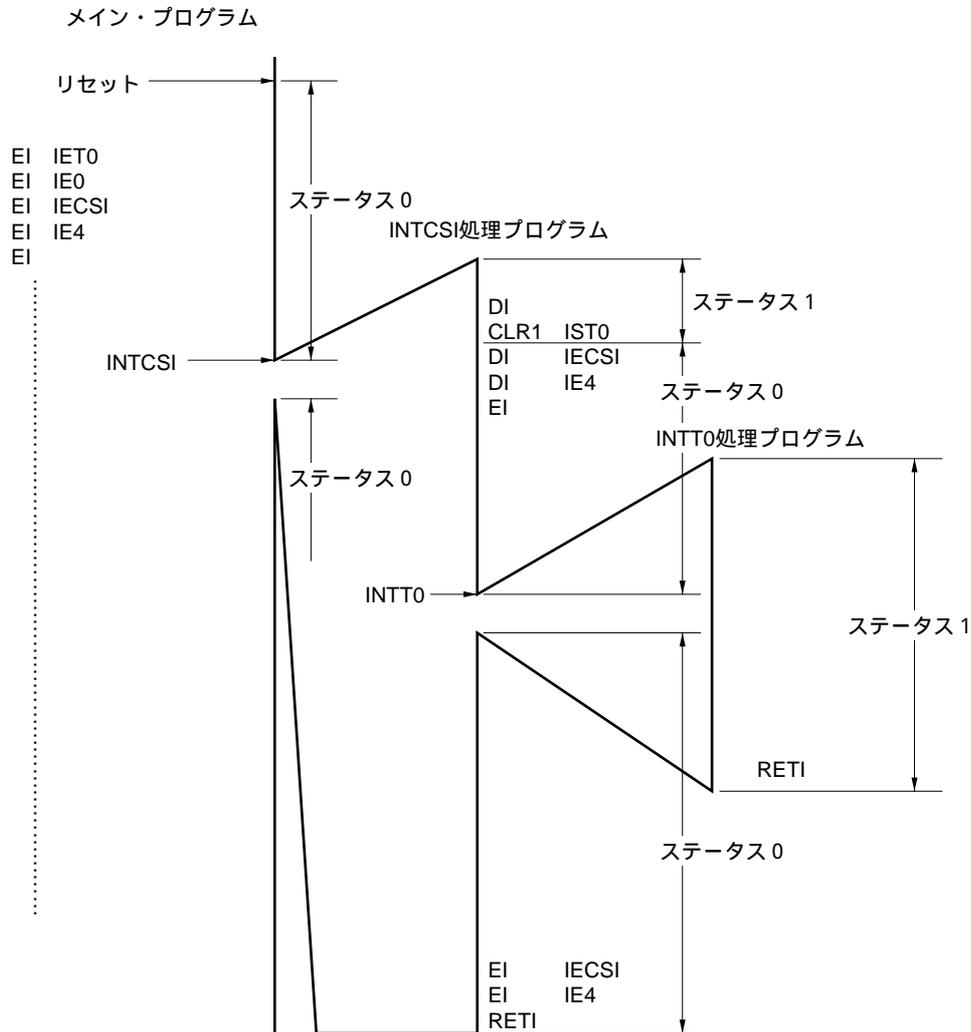
高位の割り込みINTBTと低位の割り込みINTT0が同時に発生すると、高位の割り込み処理を開始（高位の割り込み処理中に、高位の割り込みが発生するおそれがないことが確実ならDI IE × ×は不要）。

高位の割り込み実行中に低位の割り込みが発生すると、割り込みは保留される。

高位の割り込みを終了すると、保留されていた低位の割り込みのうち割り込み順位の高いINTCSI処理が実行される。

INTCSI処理が終了すると保留されていたINTT0が処理される。

(7) 二重割り込みを2つ許可する。——INTT0とINT0は二重割り込みを許可。INTCSI, INT4は一重割り込み——



二重割り込みを許可しない割り込みINTCSIの発生によりINTCSI処理プログラムをスタート。ステータスは1となる。

IST0のクリアにより、ステータスを0にする。二重割り込みを許可しないINTCSIとINT4を禁止。

二重割り込みを許可するINTT0発生により、二重割り込み実行、ステータス1となり、すべての割り込みは禁止される。

INTT0処理終了により、ステータスは0に戻る。

禁止していたINTCSI, INT4を許可して復帰。

6.10 テスト機能

6.10.1 テスト・ソースの種類

μPD753208には、2種類のテスト・ソースがあります。テスト・ソースのうちINT2は、2種類のエッジ検出テストابل入力を備えています。

表6 - 5 テスト・ソースの種類

テスト・ソース	内/外
INT2 (KR0-KR3のいずれかの端子への立ち下がりエッジ検出)	外
INTW (時計用タイマからの信号)	内

6.10.2 テスト機能を制御する各種ハードウェア

(1) テスト要求フラグ, テスト許可フラグ

テスト要求フラグ (IRQ $\times\times\times$) は、テスト要求発生 (INT $\times\times\times$) で“1”にセットされます。テスト処理が実行されたら、ソフトウェアで“0”にクリアしてください。

テスト許可フラグ (IE $\times\times\times$) は、各テスト要求フラグに対応して個別に備わっており、内容が“1”のときスタンバイ・リリース信号を許可し、“0”のとき禁止します。

テスト要求フラグ, テスト許可フラグがともに“1”にセットされている場合、スタンバイ・リリース信号を発生します。

表6 - 6 にテスト要求フラグのセット信号を示します。

表6 - 6 テスト要求フラグのセット信号

テスト要求フラグ	テスト要求フラグのセット信号	テスト許可フラグ
IRQW	時計用タイマからの信号でセット。	IEW
IRQ2	KR0/P60-KR3/P63のいずれかの端子への入力の立ち下がりエッジ検出によりセット。検出エッジはINT2エッジ検出モード・レジスタ (IM2) により選択。	IE2

(2) キー割り込み (KR0-KR3) のハードウェア

KR0-KR3の構成を図6 - 10に示します。

KR0-KR3のうち、割り込み入力に使用する端子をINT2エッジ検出モード・レジスタ (IM2) によって選択します。選択された端子の入力のいずれかの立ち下がりエッジを検出するとIRQ2がセットされます。

IM2のフォーマットを図6 - 11に示します。IM2は4ビット操作命令によって設定します。リセット信号の発生により全ビット“0”にクリアされ、INT2の立ち上がりエッジ指定となります。

図 6 - 10 KR0-KR3のブロック図

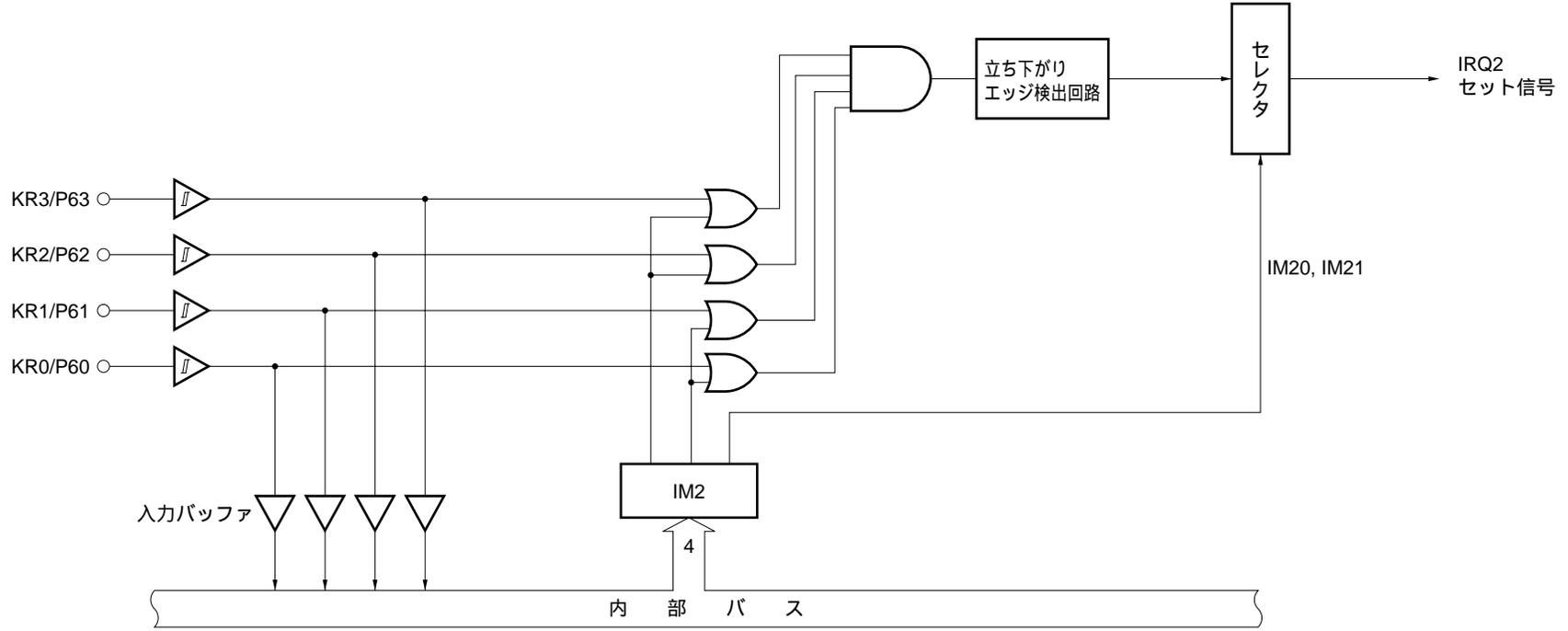
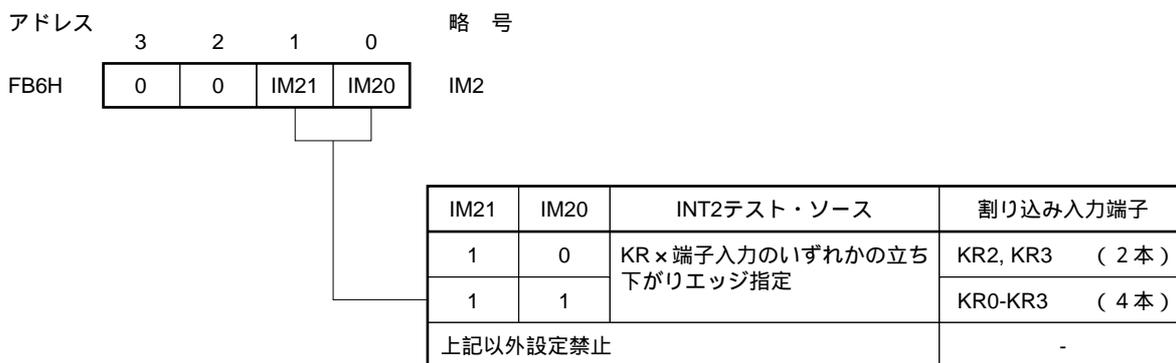


図6 - 11 INT2エッジ検出モード・レジスタ (IM2) のフォーマット



- 注意 1 . エッジ検出モード・レジスタを変更すると、テスト要求フラグがセットされる場合がありますので、あらかじめテスト入力を禁止してモード・レジスタを変更し、CLR1命令によってテスト要求フラグをクリアしてから、テスト入力を許可してください。
- 2 . 立ち下がりエッジ検出として選択した端子のうち1本でもロウ・レベルが入力されていると、ほかの端子に立ち下がりエッジが入力されても、IRQ2はセットされません。

〔メモ〕

第7章 スタンバイ機能

μPD753208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2つのモードがあります。

STOPモード

HALTモード

STOPモード、HALTモードはそれぞれ以下に示すような機能を持っています。

(1) STOPモード

システム・クロック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。CPUの消費電流は、かなり低減されます。

また、データ・メモリの低電圧 ($V_{DD} = 1.8\text{ V}$ まで) 保持が可能です。したがって、超低消費電流で、データ・メモリの内容を保持する場合に有効です。

μPD753208のSTOPモードは、割り込み要求によって解除できるため、間欠動作も可能です。しかし、STOPモードの解除時に発振安定時間確保のためのウェイト時間がとられるため、割り込み要求によって、すぐ処理を開始しなければならない場合にはHALTモードを選択してください。

(2) HALTモード

CPUの動作クロックを停止させるモードです。システム・クロック発振回路の発振は続きます。このモードでは、STOPモードほどの消費電流の低減はできませんが、割り込み要求により、すぐに処理を再開したい場合や、時計動作のような間欠動作をさせたい場合に有効です。

いずれのモードでも、スタンバイ・モードに設定される直前のレジスタ、フラグ、データ・メモリの内容はすべて保持されます。また、入出力ポートの出力ラッチ、出力バッファの状態も保持されますから、入出力ポートの状態を、あらかじめ、システム全体の消費電流が最も小さくなるように処理しておきます。

使用上の注意を次ページに示します。

注意1 . LCDコントローラ/ドライバ, 時計用タイマの動作クロックは, STOPモードに設定するとその動作が停止します。

したがって, 動作を継続する場合はHALTモードを使用してください。

2 . スタンバイ・モードとCPUクロックを切り替えることで効率の良い低消費電流, 低電圧動作が可能ですが, いずれもコントロール・レジスタを操作して新しいクロックを選択してから切り替え後のクロックで動作を開始するまでに, 5.2.3 CPUクロックの設定で述べたような時間が必要です。このため, CPUクロック切り替え機能とスタンバイ・モードを組み合わせる場合には, 切り替わりに必要な時間経過後, スタンバイ・モードに設定してください。

3 . スタンバイ・モードを使用する場合には, 入出力ポートは消費電流が最小となるよう処理してください。

特に, 入力ポートはオープンにしないでください。必ずロウ・レベルかハイ・レベルを入力してください。

7.1 スタンバイ・モードの設定および動作状態

表7-1 スタンバイ・モード時の各動作状態

項目	モード	STOPモード	HALTモード
設定命令		STOP命令	HALT命令
★ 動作状態	クロック発生回路	システム・クロックのみ発振停止	CPUクロックのみ停止（発振継続）
	ベーシック・インターバル・ タイム/ウォッチドッグ・タイム	動作停止	システム・クロック発振時のみ動作可能 (BTモード：基準時間間隔でIRQBTをセット) (WTモード：BTのオーバフローによりリセット発生)
	シリアル・インタ フェース	シリアル・クロックに外部 $\overline{\text{SCK}}$ 入力を選択し た場合のみ、動作可能	動作可能
	タイマ/イベント・ カウンタ	カウント・クロックにTI0の端子入力を指定し た場合のみ、動作可能	動作可能
	時計用タイマ	動作停止	動作可能
	LCDコントローラ/ ドライバ	動作停止	動作可能
	外部割り込み	INT2, 4は動作可能 INT0のみ動作不可能 ^注	
	CPU	動作停止	
解除信号		割り込み許可フラグで許可されている動作可能なハードウェアからの割り込み要求信号または $\overline{\text{RESET}}$ 信号発生	

注 エッジ検出モード・レジスタ (IM0) のビット2により、ノイズ除去回路を選択しない場合 (IM02 = 1 のとき) のみ動作可能。

STOPモードにはSTOP命令で、HALTモードにはHALT命令で設定します（STOP命令、HALT命令は、それぞれPCCのビット3、ビット2をセットする命令です）。

STOP命令、HALT命令の次には必ずNOP命令を書くようにしてください。

PCCの下位2ビットによってCPU動作クロックを変更する場合、表5-5 CPUクロックの切り替えに要する最大時間に示すようにPCCを書き換えてから、CPUクロックが変更されるまでには、時間的なずれを生じる場合があります。したがって、スタンバイ・モード前の動作クロックとスタンバイ・モード解除後のCPUクロックを変更するような場合には、PCCを書き換えてからCPUクロックの変更に必要なマシン・サイクル経過後にスタンバイ・モードを設定します。

スタンバイ・モードでは、汎用レジスタ、フラグ類、モード・レジスタ類、出力ラッチなど、スタンバイ・モード中動作を停止しているすべてのレジスタとデータ・メモリのデータが保持されます。

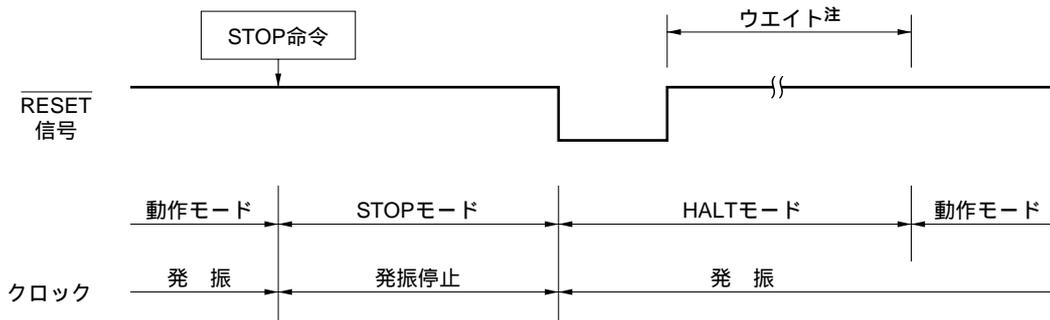
- ★ 注意1 . STOPモードに設定すると、X2端子が内部で V_{DD} に50 k（TYP.）の抵抗でプルアップされません。
- 2 . スタンバイ・モードを設定する前に、あらかじめすべての割り込み要求フラグをリセットしてください。割り込み要求フラグと割り込み許可フラグの両方がセットされている割り込みソースがある場合には、スタンバイ・モードに入ってもただちに解除されず（図6-1 割り込み制御回路ブロック図を参照）。ただし、STOPモードを設定した場合は、STOP命令実行後すぐHALTモードに入り、BTMレジスタによる設定時間だけウエイトしたあと動作モードに戻ります。

7.2 スタンバイ・モードの解除

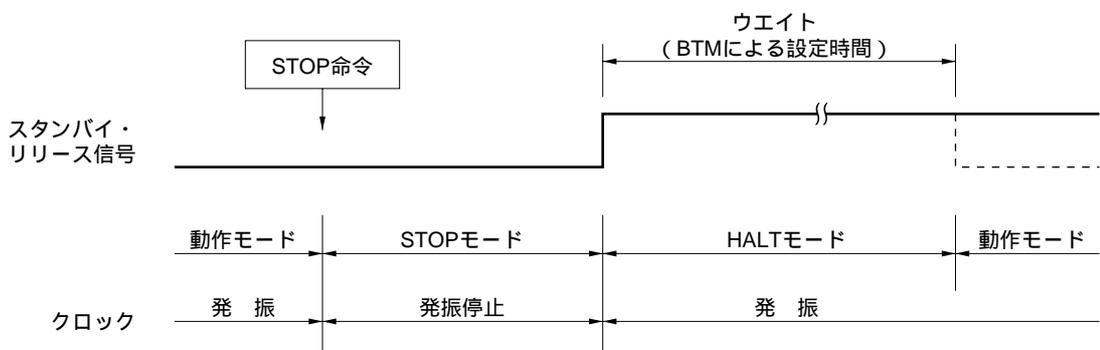
STOPモード、HALTモードいずれも割り込み許可フラグで許可されている割り込み要求信号の発生、および $\overline{\text{RESET}}$ 信号発生により解除されます。各モードの解除動作は図7-1のようになります。

図7-1 スタンバイ・モードの解除動作(1/2)

(a) STOPモードの $\overline{\text{RESET}}$ 信号発生による解除



(b) STOPモードの割り込み発生による解除



注 マスク・オプションにより次の2つの時間を選択することができます。

$2^{17}/f_x$ (21.8 ms : 6.00 MHz動作時, 31.3 ms : 4.19 MHz動作時)

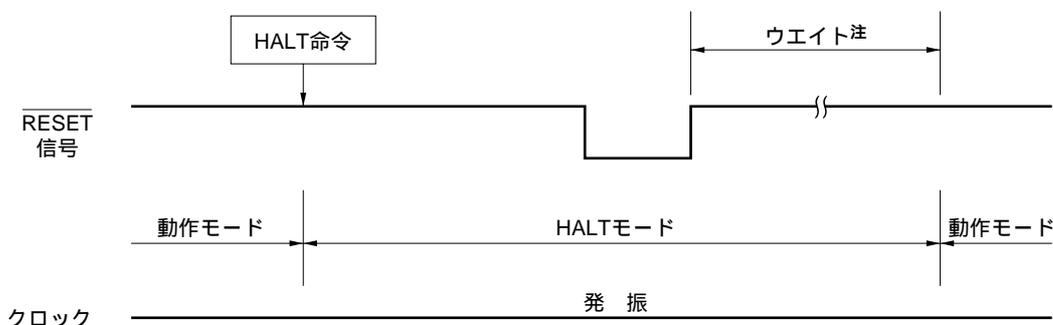
$2^{15}/f_x$ (5.46 ms : 6.00 MHz動作時, 7.81 ms : 4.19 MHz動作時)

★ ただし、 μ PD75P3216にはマスク・オプションがなく、ウェイト時間は $2^{15}/f_x$ に固定されています。

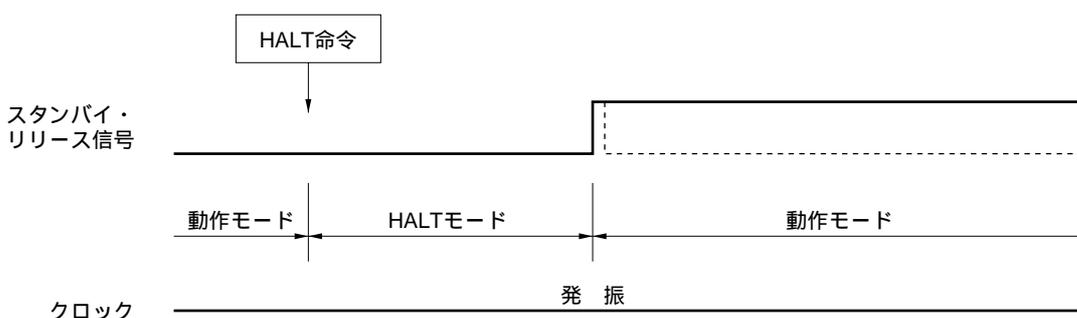
備考 破線は、スタンバイを解除した割り込み要求が受け付けられた場合です。

図7-1 スタンバイ・モードの解除動作 (2/2)

(c) HALTモードの $\overline{\text{RESET}}$ 信号発生による解除



(d) HALTモードの割り込み発生による解除



注 マスク・オプションにより次の2つの時間を選択することができます。

$2^{17}/f_x$ (21.8 ms : 6.00 MHz動作時, 31.3 ms : 4.19 MHz動作時)

$2^{15}/f_x$ (5.46 ms : 6.00 MHz動作時, 7.81 ms : 4.19 MHz動作時)

★ ただし、 μ PD75P3216にはマスク・オプションがなく、ウエイト時間は $2^{15}/f_x$ に固定されています。

備考 破線は、スタンバイを解除した割り込み要求が受け付けられた場合です。

STOPモードを割り込み発生により解除した場合、ウエイト時間はBTMの設定により決まります(表7-2参照)。

発振が安定するまでの時間は、使用する発振子の種類と、STOPモード解除時の電源電圧によって変わります。したがって、使用状況に応じてウエイト時間を選択し、STOPモードを設定する前に、BTMをセットします。

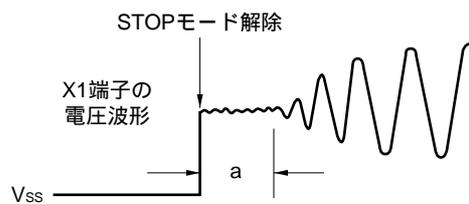
表7 - 2 BTMによるウェイト時間の選択

BTM3	BTM2	BTM1	BTM0	ウェイト時間 ^注	
				f _x = 6.00 MHz時	f _x = 4.19 MHz時
-	0	0	0	約2 ²⁰ /f _x (約175 ms)	約2 ²⁰ /f _x (約250 ms)
-	0	1	1	約2 ¹⁷ /f _x (約21.8 ms)	約2 ¹⁷ /f _x (約31.3 ms)
-	1	0	1	約2 ¹⁵ /f _x (約5.46 ms)	約2 ¹⁵ /f _x (約7.81 ms)
-	1	1	1	約2 ¹³ /f _x (約1.37 ms)	約2 ¹³ /f _x (約1.95 ms)
上記以外				設定禁止	

注 この時間は、STOPモード解除後、発振を開始するまでの時間は含みません。

注意 STOPモード解除時のウェイト時間は、 $\overline{\text{RESET}}$ 信号発生による場合も、割り込み発生による場合もSTOPモード解除後クロック発振を開始するまでの時間(図7 - 2のa)は含みません。

図7 - 2 STOPモード解除時のウェイト時間



7.3 スタンバイ・モード解除後の動作

- (1) $\overline{\text{RESET}}$ 信号発生により解除された場合は、通常のリセット動作を実行します。
- (2) 割り込み発生により解除された場合、CPUが命令実行を再開したときにベクタ割り込みを行うかどうかは、割り込みマスタ許可フラグ(IME)の内容により決まります。

(a) IME = 0 のとき

スタンバイ・モード解除後、スタンバイ・モード設定の次の命令から実行を再開します。
割り込み要求フラグは保持されています。

(b) IME = 1 のとき

スタンバイ・モード解除後、2命令を実行してからベクタ割り込みが実行されます。ただし、INTW, KR0-KR3によって解除された場合はベクタ割り込みは発生しませんから、(a)と同様の処理を行います。

★ 7.4 マスク・オプションの選択

μPD753208のスタンバイ機能では、 $\overline{\text{RESET}}$ 信号発生によるスタンバイ機能解除後のウェイト時間を、次の2種類からマスク・オプションにより選択することができます。

$2^{17}/f_x$ (21.8 ms : 6.0 MHz動作時, 31.3 ms : 4.19 MHz動作時)

$2^{15}/f_x$ (5.46 ms : 6.0 MHz動作時, 7.81 ms : 4.19 MHz動作時)

ただし、μPD75P3216にはマスク・オプションはなく、 $2^{15}/f_x$ に固定されています。

7.5 スタンバイ・モードの応用

スタンバイ・モードを使用する場合には次のような手順で行います。

割り込み入力あるいはポート入力による電源断などスタンバイ・モード設定要因の検出（電源断検出にはINT4を使用すると効果的です）。

入出力ポートの処理（消費電流が最小となるよう処理する）。

特に、入力ポートはオープンにしないでください。必ずロウ・レベルかハイ・レベルを入力してください。

スタンバイ・モードを解除する割り込みの指定（INT4を使用すると効果的、解除しない割り込み許可フラグはクリアする）。

解除後の動作指定（割り込み処理をするかしないかによりIMEを操作）。

解除後のCPUクロックの指定。

解除時のウェイト時間の選択。

スタンバイ・モード設定（STOP, HALT命令）。

例 次の条件でSTOPモードを使用する場合

INT4の立ち下がりエッジ入力によりSTOPモードを設定し、立ち上がりエッジ入力で解除する（INTBTは使用しない）。

入出力ポートは、すべてハイ・インピーダンスとする（ハイ・インピーダンスにおいて消費電流が少なくなるように、外部で端子が処理されている場合）。

プログラムで使用する割り込みはINT0、INTT0。ただし、これらをSTOP解除には使用しない。解除後も割り込み許可する。

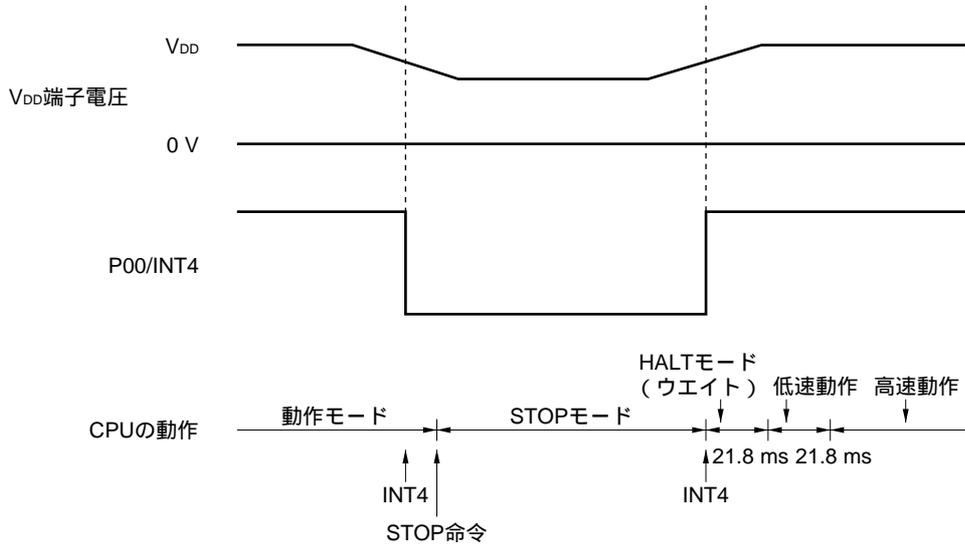
解除後は、最低速のCPUクロックで動作スタートする。

解除時のウェイト時間は約21.8 msとする。

解除後、電源の安定のためさらに21.8 msウェイトする。またP00/INT4端子は2回チェックして、チャタリングを除去する。

タイミング・チャート

★



プログラム例

(INT4処理プログラム, MBE = 0)

```

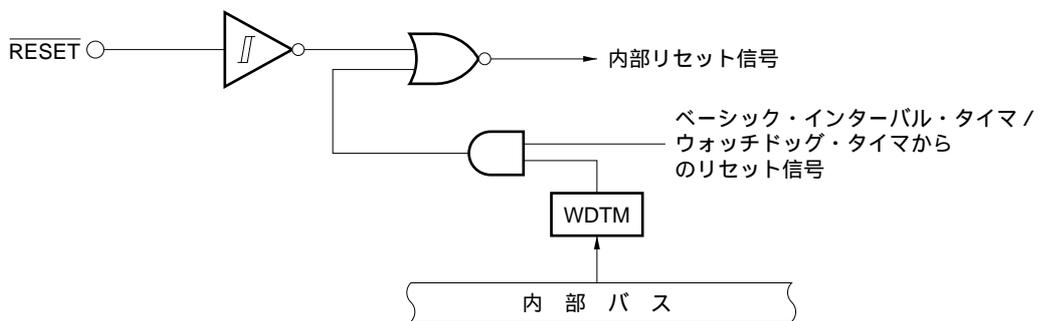
VSB4 :   SKT   PORT0.0      ; P00 = 1 ?
         BR    PDOWN       ; パワーダウン
         SET1  BTM. 3       ; パワーオン
WAIT :   SKT   IRQBT       ; 21.8 msのウェイト
         BR    WAIT
         SKT   PORT0.0     ; チャタリングをチェック
         BR    PDOWN
         MOV   A,  0011B
         MOV   PCC, A      ; 高速モード設定
         ( MOV  XA,  x x H ) ; ポート・モード・レジスタをセット
         ( MOV  PMGm, XA )
         EI    IE0
         EI    IET0
         RETI
PDOWN :  MOV   A,  0        ; 最低速モード
         MOV   PCC, A
         MOV   XA,  00H
         MOV   LCDM, XA    ; LCD表示オフ
         MOV   LCDC, A
         MOV   PMGA, XA   ; 入出力ポート・ハイ・インピーダンス
         MOV   PMGB, XA
         DI    IE0        ; INT0, INTT0禁止
         DI    IET0
         MOV   A,  1011B
         MOV   BTM, A     ; ウェイト時間 21.8 ms
         NOP
         STOP          ; STOPモードをセット
         NOP
         RETI

```

第8章 リセット機能

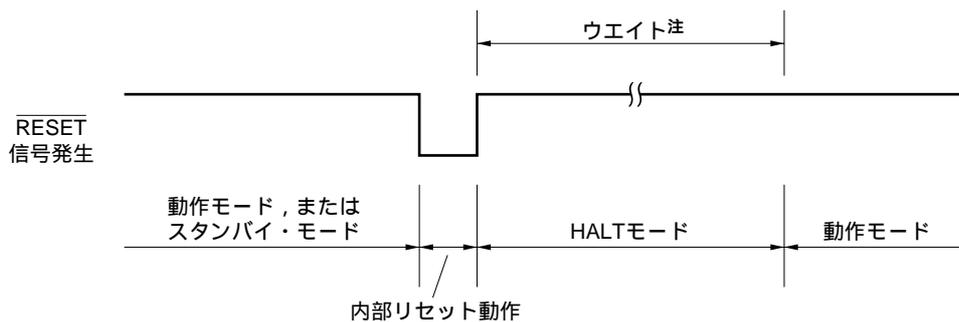
リセット入力には、外部リセット信号 ($\overline{\text{RESET}}$) とベーシック・インターバル・タイマ / ウォッチドッグ・タイマからのリセット信号の2種類があります。どちらか一方のリセット信号が入力されると、内部リセット信号が発生します。図8 - 1 にそれぞれの構成を示します。

図8 - 1 リセット機能の構成



$\overline{\text{RESET}}$ 信号発生によって、各ハードウェアは表8 - 1 に示すようにイニシャライズされます。リセット動作のタイミングを図8 - 2 に示します。

図8 - 2 $\overline{\text{RESET}}$ 信号発生によるリセット動作



注 マスク・オプションにより次の2つの時間を選択することができます。

$2^{17}/f_x$ (21.8 ms : 6.00 MHz動作時, 31.3 ms : 4.19 MHz動作時)

$2^{15}/f_x$ (5.46 ms : 6.00 MHz動作時, 7.81 ms : 4.19 MHz動作時)

★ ただし、 μ PD75P3216にはマスク・オプションがなく、ウェイト時間は $2^{15}/f_x$ に固定されています。

表8-1 各ハードウェアのリセット後の状態(1/2)

ハードウェア		スタンバイ・モード中の $\overline{\text{RESET}}$ 信号発生	動作中の $\overline{\text{RESET}}$ 信号発生
プログラム・カウンタ(PC)	μ PD753204	プログラム・メモリの0000H番地の下位4ビットをPC11-8に, 0001H番地の内容をPC7-0にセット	左に同じ
	μ PD753206, 753208	プログラム・メモリの0000H番地の下位5ビットをPC12-8に, 0001H番地の内容をPC7-0にセット	
	μ PD75P3216	プログラム・メモリの0000H番地の下位6ビットをPC13-8に, 0001H番地の内容をPC7-0にセット	
PSW	キャリー・フラグ(CY)	保持	不定
	スキップ・フラグ(SK0-2)	0	0
	割り込みステータス・フラグ(IST0, 1)	0	0
	バンク許可フラグ(MBE, RBE)	プログラム・メモリの0000H番地のビット6をRBE, ビット7をMBEにセット	左に同じ
スタック・ポインタ(SP)		不定	不定
スタック・バンク選択レジスタ(SBS)		1000B	1000B
データ・メモリ(RAM)		保持	不定
汎用レジスタ(X, A, H, L, D, E, B, C)		保持	不定
バンク選択レジスタ(MBS, RBS)		0, 0	0, 0
ベーシック・インターバル・タイマ/ウォッチドッグ・タイマ	カウンタ(BT)	不定	不定
	モード・レジスタ(BTM)	0	0
	ウォッチドッグ・タイマ許可フラグ(WDTM)	0	0
タイマ/イベント・カウンタ(T0)	カウンタ(T0)	0	0
	モジュロ・レジスタ(TMOD0)	FFH	FFH
	モード・レジスタ(TM0)	0	0
	TOE0, TOUT F/F	0, 0	0, 0
タイマ・カウンタ(T1)	カウンタ(T1)	0	0
	モジュロ・レジスタ(TMOD1)	FFH	FFH
	モード・レジスタ(TM1)	0	0
	TOE1, TOUT F/F	0, 0	0, 0

表8-1 各ハードウェアのリセット後の状態(2/2)

ハードウェア		スタンバイ・モード中の $\overline{\text{RESET}}$ 信号発生	動作中の $\overline{\text{RESET}}$ 信号発生
タイマ・ カウンタ(T2)	カウンタ(T2)	0	0
	モジュロ・レジスタ(TMOD2)	FFH	FFH
	ハイ・レベル期間設定用モジュロ・ レジスタ(TMOD2H)	FFH	FFH
	モード・レジスタ(TM2)	0	0
	TOE2, TOUT F/F	0, 0	0, 0
	REMC, NRZ, NRZB	0, 0, 0	0, 0, 0
	TGCE	0	0
時計用タイマ	モード・レジスタ(WM)	0	0
シリアル・ インタフェース	シフト・レジスタ(SIO)	保 持	不 定
	動作モード・レジスタ(CSIM)	0	0
	SBIコントロール・レジスタ(SBIC)	0	0
	スレーブ・アドレス・レジスタ (SVA)	保 持	不 定
クロック発生回路, クロック出力回路	プロセッサ・クロック・コントロー ル・レジスタ(PCC)	0	0
	クロック出力モード・レジスタ (CLOM)	0	0
LCDコントローラ/ ドライバ	表示モード・レジスタ(LCDM)	0	0
	表示制御レジスタ(LCDC)	0	0
	LCD/ポート選択レジスタ(LPS)	0	0
割り込み機能	割り込み要求フラグ (IRQ $\times\times\times$)	リセット(0)	リセット(0)
	割り込み許可フラグ (IE $\times\times\times$)	0	0
	割り込みマスタ許可フラグ (IME)	0	0
	プライオリティ選択レジスタ(IPS)	0	0
	INT0, 2モード・レジスタ (IM0, 2)	0, 0	0, 0
デジタル・ ポート	出力バッファ	オフ	オフ
	出力ラッチ	クリア(0)	クリア(0)
	入出力モード・レジスタ (PMGA, B, C)	0	0
	プルアップ抵抗指定レジスタ (POGA, B)	0	0
ビット・シーケンシャル・バッファ(BSB0-3)		保 持	不 定

★

〔メモ〕

第9章 PROM（プログラム・メモリ）の書き込みとベリファイ

μPD75P3216に内蔵されているプログラム・メモリはワン・タイムPROMです。メモリ容量は次のとおりです。

μPD75P3216：16384ワード×8ビット

このワン・タイムPROMの書き込み/ベリファイのために、表9-1に示すような端子を使用します。なお、アドレスの入力はなく、代わりにX1端子からクロック入力することでアドレスを更新します。

表9-1 プログラム・メモリ書き込み/ベリファイ時の使用端子

端子名	機能
V _{PP}	プログラム・メモリ書き込み/ベリファイ時のプログラム電圧印加端子（通常はV _{DD} 電位）。
X1, X2	プログラム・メモリ書き込み/ベリファイ時のアドレス更新クロック入力。X2端子にはX1端子の逆相信号を入力。
MD0-MD3	プログラム・メモリ書き込み/ベリファイ時の動作モード選択端子。
D0/P60-D3/P63（下位4） D4/P50-D7/P53（上位4）	プログラム・メモリ書き込み/ベリファイ時の8ビット・データ入出力端子。
V _{DD}	電源電圧印加端子。通常動作時は1.8~5.5V、プログラム・メモリ書き込み/ベリファイ時は+6Vを印加。

注意1．μPD75P3216は、消去用窓を持っていないので、紫外線消去はできません。

2．プログラム・メモリの書き込み/ベリファイ時に使用しない端子は、V_{SS}に接続します。

9.1 プログラム・メモリ書き込み/ベリファイ時の動作モード

μPD75P3216は、V_{DD}端子に+6V、V_{PP}端子に+12.5Vを印加すると、プログラム・メモリ書き込み/ベリファイ・モードになります。このモードは、MD0-MD3端子の設定により次のような動作モードとなります。

表9 - 2 動作モード

動作モードの指定						動作モード
V _{PP}	V _{DD}	MD0	MD1	MD2	MD3	
+12.5V	+6V	H	L	H	L	プログラム・メモリ・アドレスの0クリア
		L	H	H	H	書き込みモード
		L	L	H	H	ベリファイ・モード
		H	x	H	H	プログラム・インヒビット・モード

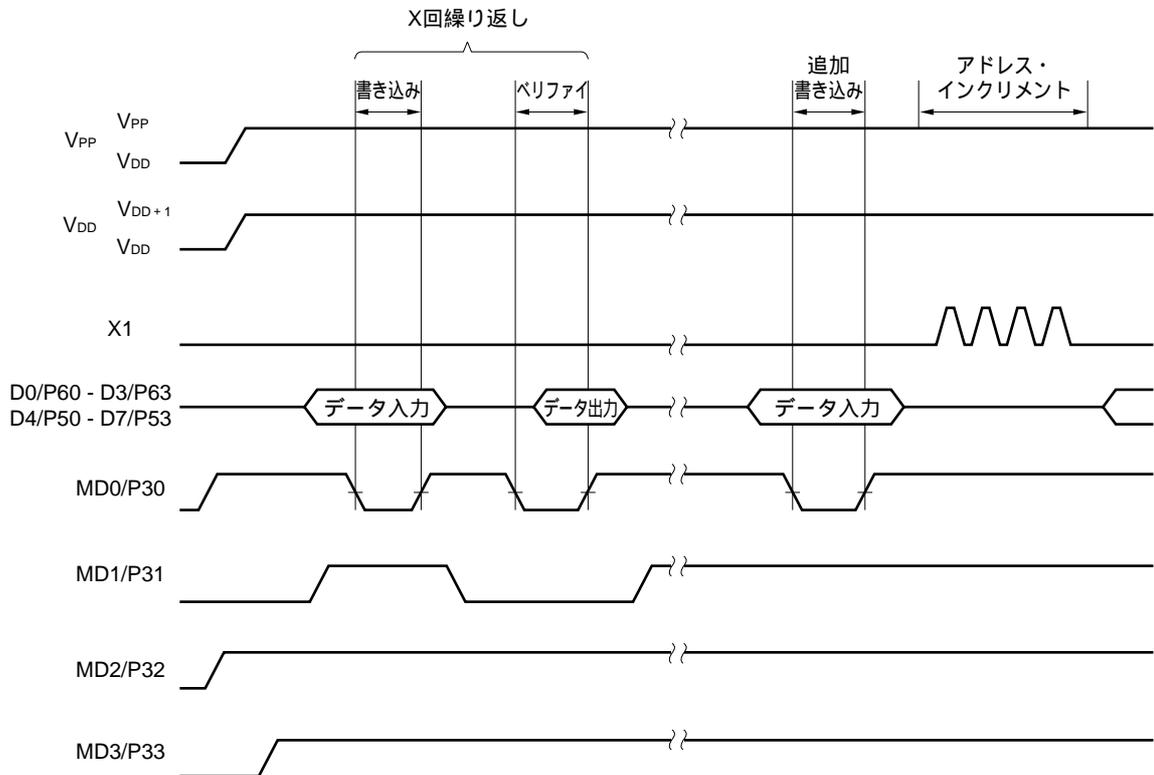
x : LまたはH

★ 9.2 プログラム・メモリ書き込みの手順

プログラム・メモリ書き込みの手順は次のようになっており、高速書き込みが可能です。

- (1) 使用しない端子を抵抗を介してV_{SS}にプルダウン。X1端子はロウ・レベル。
- (2) V_{DD}, V_{PP}端子に5 Vを供給。
- (3) 10 μsウエイト。
- (4) プログラム・メモリ・アドレスの0クリア・モード。
- (5) V_{DD}に6 V, V_{PP}に12.5 Vを供給。
- (6) 1 msの書き込みモードでデータを書き込む。
- (7) ベリファイ・モード。書き込めていれば(8)へ、書き込めていなければ(6), (7)を繰り返す。
- (8) ((6), (7)で書き込んだ回数:X) × 1 msの追加書き込み。
- (9) X1端子にパルスを4発入力することにより、プログラム・メモリ・アドレスを更新(+1)。
- (10) (6) ~ (9)を最終アドレスまで繰り返す。
- (11) プログラム・メモリ・アドレスの0クリア・モード。
- (12) V_{DD}, V_{PP}端子の電圧を5 Vに変更。
- (13) 電源オフ。

(2) ~ (9)の手順を下図に示します。

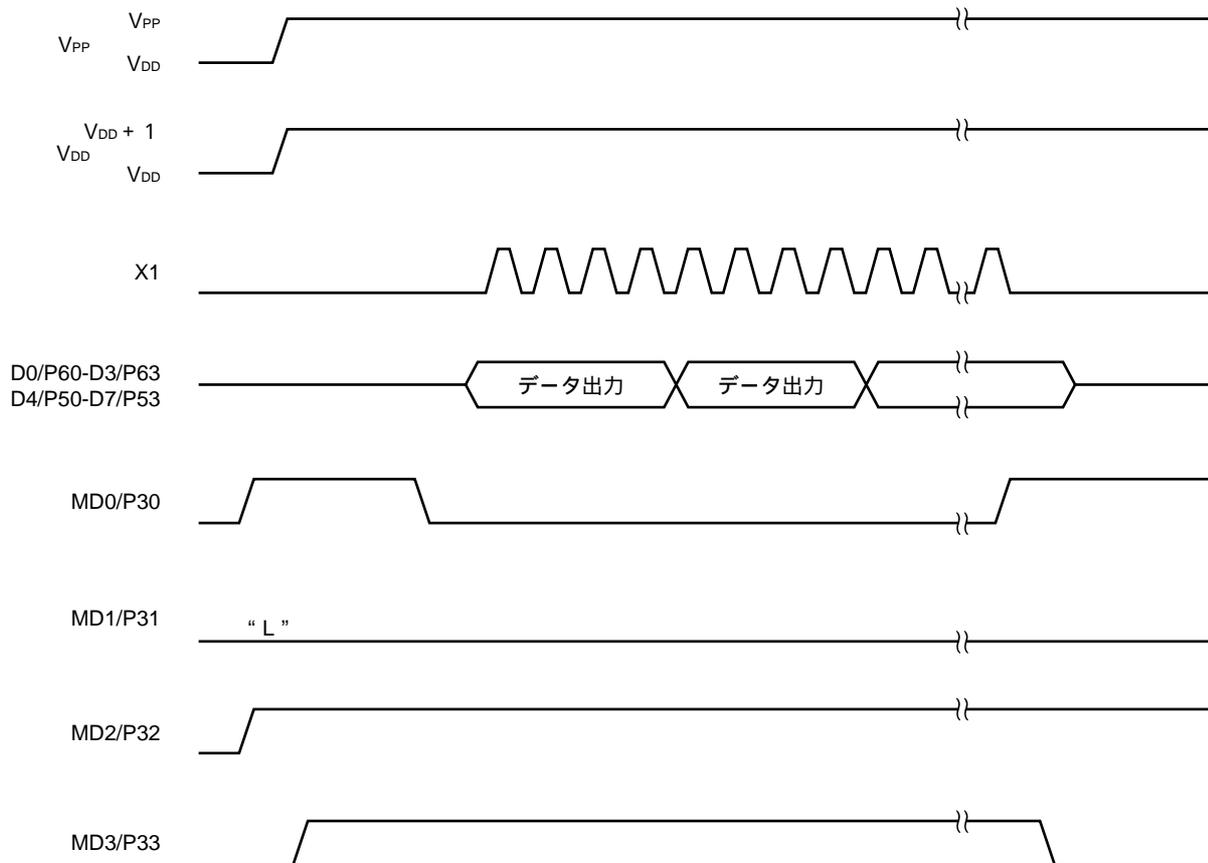


★ 9.3 プログラム・メモリの読み出しの手順

μPD75P3216は、次の手順によりプログラム・メモリの内容の読み出しができます。読み出しは、ベリファイ・モードで行います。

- (1) 使用しない端子を抵抗を介してV_{SS}にプルダウン。X1端子はロウ・レベル。
- (2) V_{DD} , V_{PP}端子に5 Vを供給。
- (3) 10 μsウエイト。
- (4) プログラム・メモリ・アドレスの0クリア・モード。
- (5) V_{DD}に6 V , V_{PP}に12.5 Vを供給。
- (6) ベリファイ・モード。X1端子にクロック・パルスを入力すると4発入力する周期でデータを1アドレスずつ順次出力。
- (7) プログラム・メモリ・アドレスの0クリア・モード。
- (8) V_{DD} , V_{PP}端子の電圧を5 Vに変更。
- (9) 電源オフ。

この(2) ~ (7) の手順を下図に示します。



9.4 ワン・タイムPROMのスクリーニングについて

ワン・タイムPROM製品は、その構造上、当社にて完全な試験をして出荷することはできません。必要なデータを書き込んだあと、下記の条件で高温保管後、PROMのベリファイを行うスクリーニングを実施することを推奨します。

保管温度	保管時間
125	24時間

- ★ なお、NECでは、QTOP™マイコンの名称でワン・タイムPROMの書き込みから捺印、スクリーニング、ベリファイを有料で行うサービスを実施しています。詳細につきましては、販売員にご相談ください。

〔メモ〕

第10章 マスク・オプション

10.1 端子

μPD753208の端子には、次のようなマスク・オプションがあります。

表10 - 1 端子のマスク・オプションの選択

端子	マスク・オプション
P50-P53	ビット単位でプルアップ抵抗の内蔵を指定可能
V _{LC0} -V _{LC2} , BIAS	4本同時にLCD駆動電源供給用分割抵抗を内蔵可能

10.1.1 P50-P53のマスク・オプション

P50-P53（ポート5）は、マスク・オプションによりプルアップ抵抗の内蔵を指定することができます。マスク・オプションはビット単位で指定することができます。

ポート5のリセット後の状態は、マスク・オプションによるプルアップ抵抗の内蔵を指定した場合、ハイ・レベルになります。内蔵を指定しない場合は、ハイ・インピーダンスになります。

μPD75P3216にはマスク・オプションによるプルアップ抵抗の内蔵機能はありません。

10.1.2 V_{LC0}-V_{LC2}のマスク・オプション

V_{LC0}-V_{LC2}端子（LCD駆動用電源）とBIAS端子（外付け分割抵抗カット用端子）には、マスク・オプションにより分割抵抗を内蔵することができます。これによって、各バイアスに応じたLCD駆動用電源を外付け分割抵抗なしで供給することができます（詳細は5.7.8 LCD駆動用電源V_{LC0}, V_{LC1}, V_{LC2}の供給を参照）。

マスク・オプションは、下の3通りの場合から選択することができます。

- 分割抵抗を内蔵しない
- 10 k（typ.）の分割抵抗を内蔵する
- 100 k（typ.）の分割抵抗を内蔵する

マスク・オプションは、V_{LC0}-V_{LC2}端子とBIAS端子の4本を同時に指定します。1本単位での指定はできません。

BIAS端子のリセット後の状態は、マスク・オプションによる分割抵抗を内蔵した場合、ロウ・レベルになります。分割抵抗を内蔵しない場合は、ハイ・インピーダンスになります。

μPD75P3216にはマスク・オプションはなく、分割抵抗を内蔵することはできません。μPD75P3216で分割抵抗を使用したい場合は、外付けにしてください。

10.2 スタンバイ機能のマスク・オプション

μPD753208のスタンバイ機能では、マスク・オプションによりウェイト時間を選択することができます。これは、 $\overline{\text{RESET}}$ 信号によってスタンバイ機能を解除したあと、通常動作モードに戻るまでの時間です（詳細は7.2 スタンバイ・モードの解除参照）。

ウェイト時間は次の2つの時間から選択することができます。

$2^{17}/f_x$ (21.8 ms : $f_x = 6.00$ MHz動作時, 31.3 ms : $f_x = 4.19$ MHz動作時)

$2^{15}/f_x$ (5.46 ms : $f_x = 6.00$ MHz動作時, 7.81 ms : $f_x = 4.19$ MHz動作時)

μPD75P3216にはマスク・オプションはなく、ウェイト時間は $2^{15}/f_x$ に固定されています。

第11章 命令セット

μPD753208の命令セットは、75Xシリーズの命令セットを改良・発展させたもので、75Xシリーズからの継承性を保持した命令セットとなっており、次のような特色があります。

- (1) 多彩な応用ができるビット操作命令
- (2) 効率のよい4ビット操作命令
- (3) 8ビット・マイコンに匹敵する8ビット操作命令
- (4) プログラム・サイズ短縮のためのGETI命令
- (5) プログラムの効率を上げるたてづみ命令，進数補正命令
- (6) 連続参照に適したテーブル参照命令
- (7) 1バイト相対分岐命令
- (8) わかりやすく整理されたNEC標準二モニック

なお，データ・メモリを操作する場合に適用できるアドレッシング・モードおよび，命令実行時に有効となるレジスタ・バンクについては3.2 汎用レジスタのバンク構成を参照してください。

11.1 特徴的な命令

ここでは，μPD753208の命令セットのうち，特徴的な命令の概要を説明します。

11.1.1 GETI命令

GETI (ゲットアイ) 命令は，以下の命令を1バイト命令に変換するための命令です。

- (a) 16Kバイト空間 (0000H-3FFFH) のサブルーチン・コール命令
- (b) 16Kバイト空間 (0000H-3FFFH) への分岐命令
- (c) 任意の2バイト，2マシン・サイクルの命令 (ただし，BRCB命令およびCALLF命令を除く)
- (d) 1バイト命令2つの組み合わせ

GETI命令では，プログラム・メモリの0020H-007FH番地のテーブルを参照し，参照された2バイトのデータを，(a) ~ (d) の命令として実行します。したがって，48通りもの(a) ~ (d) の命令が1バイト命令に変換できることになります。

このGETI命令を使って使用頻度の高い(a) ~ (d) の命令を1バイトに変換すれば，プログラムのバイト数を大幅に短縮することができます。

11.1.2 ビット操作命令

μPD753208は、通常のビット操作命令（セット、クリア命令）に加え、ビット・テスト命令、ビット転送命令、およびビット・ブーリアン命令（AND, OR, XOR）が強化されています。

操作ビットの指定は、ビット操作アドレッシングにより行います。ビット操作アドレッシングには、3種類あり、各アドレッシングにより操作されるビットを表11 - 1 に示します。

表11 - 1 ビット操作アドレッシングの種類と指定可能な範囲

アドレッシング	操作可能な周辺ハードウェア	操作可能なビットのアドレッシング範囲
fmem. bit	RBE, MBE, IST1, IST0, IE x x x , IRQ x x x	FB0H-FBFH
	PORT0-3, 5, 6, 8, 9	FF0H-FFFH
pmem. @L	BSB0-3, PORT0-3, 5, 6, 8, 9	FC0H-FFFH
@H + mem. bit	ビット操作可能なすべての周辺ハードウェア	MBで指定されるメモリ・バンクのビット操作可能なすべてのビット

備考 1 . x x x : 0, 2, 4, BT, T0, T1, T2, W, CSI

2 . MB = MBE · MBS

11.1.3 たてづみ命令

μPD753208には、次の2種類のたてづみ命令が用意されています。

(a) MOV A, n4 または MOV XA, n8

(b) MOV HL, n8

「たてづみ」とはこの2種類の命令を、それぞれ連続したアドレスに置くことを示します。

例 A0 : MOV A, 0
A1 : MOV A, 1
XA7 : MOV XA, 07

この例のように、たてづみ命令が並べられていると、最初に行うアドレスがA0の場合はそれに続く2命令が、最初に行うアドレスがA1の場合は続く1命令が、NOP命令に置き換えられて実行されます。すなわち、最初に行われた命令だけが有効で、それに続くたてづみ命令はすべてNOP命令として処理されるわけです。

このたてづみ命令を使うことによって、アキュムレータ（Aレジスタ、レジスタ・ペアXA）への定数設定や、データ・ポインタ（レジスタ・ペアHL）への定数設定を効率よく行うことができます。

11.1.4 進数補正命令

応用によっては、4ビット・データの加算または減算（2進数で行われる）の結果を10進数に変換したり、あるいは、時刻のように6進補正したりする必要があります。

このためにμPD753208の命令セットでは、4ビット・データの加算あるいは減算の結果を、任意の進数に補正するための進数補正命令が用意されています。

(a) 加算時の進数補正

補正したい進数値をmとすると

```
ADDS A, 16 - m
ADDC A, @HL ; A, CY A + (HL) + CY
ADDS A, #m
```

の組み合わせによって、アキュムレータとメモリ（HL）を加算し、加算結果をm進補正します。オーバーフローはキャリー・フラグに残ります。

ADDC A, @HL命令の実行の結果、キャリーが出ると続くADDS A, #n4命令をスキップします。またキャリーが出なければADDS A, #n4命令が実行されますが、このときこの命令のスキップ機能は禁止され、加算の結果キャリーが出ても続く命令をスキップしません。したがってADDS A, #n4命令に続けてプログラムを書くことができます。

例 アキュムレータと、メモリを10進加算する。

```
ADDS A, 6
ADDC A, @HL ; A, CY A + (HL) + CY
ADDS A, #10
    :
```

(b) 減算時の進数補正

補正したい進数値をmとすると

```
SUBC A, @HL
ADDS A, #m
```

の組み合わせにより、アキュムレータからメモリ（HL）を減算し、結果をm進補正します。アンダフローはキャリー・フラグに残ります。

SUBC A, @HL命令の実行の結果、ボローが出なければ続くADDS A, #n4命令をスキップします。またボローが出るとADDS A, #n4命令が実行されますが、このときこの命令のスキップ機能は禁止され、加算の結果キャリーが出ても続く命令をスキップしません。したがって、ADDS A, #n4命令に続けてプログラムを書くことができます。

11.1.5 スキップ命令とスキップに要するマシン・サイクル数

μPD753208の命令セットでは、スキップによって条件判断をして、プログラムを構成するようになっています。

スキップ命令を実行したとき、スキップ条件が満足されると、続く1命令を飛ばして(スキップして)、1命令あとの命令が実行されます。

スキップが発生したとき、スキップに要するマシン・サイクル数は

(a) スキップ命令に続く命令(スキップされる命令)が、3バイト命令(BR !addr命令, BRA !addr1命令, CALL !addr命令, CALLA !addr1命令)の場合: 2マシン・サイクル

(b) (a) 以外の命令の場合: 1マシン・サイクル

11.2 命令セットとそのオペレーション

(1) オペランドの表現形式と記述方法

各命令のオペランド欄には、その命令のオペランド表現形式に対する記述方法に従ってオペランドを記述します（詳細はRA75X **アセンブラ・パッケージ ユーザーズ・マニュアル 言語編** (U12385J) を参照してください）。記述方法の中で、複数個あるものは、それらの要素の1つを選択します。大文字で書かれた英字および+、-記号はキー・ワードであり、そのまま記述します。

イミディエイト・データの場合は、適当な数値またはラベルを記述します。

ラベルとしてmem, fmem, pmem, bitなどの代わりに、図3-7に書かれた各種レジスタ・フラグの略号を記述できます（ただし、fmem, pmemは記述できるラベルに制限があります。詳しくは、表3-1 **アドレッシング・モード**、図3-7 **μPD753208 I/Oマップ**を参照してください）。

表現形式	記述方法
reg	X, A, B, C, D, E, H, L
reg1	X, B, C, D, E, H, L
rp	XA, BC, DE, HL
rp1	BC, DE, HL
rp2	BC, DE
rp'	XA, BC, DE, HL, XA', BC', DE', HL'
rp'1	BC, DE, HL, XA', BC', DE', HL'
rpa	HL, HL+, HL-, DE, DL
rpa1	DE, DL
n4	4ビット・イミディエイト・データまたはラベル
n8	8ビット・イミディエイト・データまたはラベル
mem	8ビット・イミディエイト・データまたはラベル ^注
bit	2ビット・イミディエイト・データまたはラベル
fmem	FB0H-FBFH, FF0H-FFFHイミディエイト・データまたはラベル
pmem	FC0H-FFFHイミディエイト・データまたはラベル
addr	000H-FFFHイミディエイト・データまたはラベル (μPD753204) 0000H-17FFHイミディエイト・データまたはラベル (μPD753206) 0000H-1FFFHイミディエイト・データまたはラベル (μPD753208) 0000H-3FFFHイミディエイト・データまたはラベル (μPD75P3216)
addr1	000H-FFFHイミディエイト・データまたはラベル (μPD753204) 0000H-17FFHイミディエイト・データまたはラベル (μPD753206) 0000H-1FFFHイミディエイト・データまたはラベル (μPD753208) 0000H-3FFFHイミディエイト・データまたはラベル (μPD75P3216)
caddr	12ビット・イミディエイト・データまたはラベル
faddr	11ビット・イミディエイト・データまたはラベル
taddr	20H-7FHイミディエイト・データ (ただしbit0=0) またはラベル
PORTn	PORT0-PORT3, PORT5, PORT6, PORT8, PORT9
IE x x x	IEBT, IET0-IET2, IE0, IE2, IE4, IECSi, IEW
RBn	RB0-RB3
MBn	MB0, MB1, MB15

注 memは、8ビット・データ処理の場合は偶数アドレスのみ記述できます。

(2) オペレーション説明上の凡例

A	: Aレジスタ; 4ビット・アキュムレータ
B	: Bレジスタ
C	: Cレジスタ
D	: Dレジスタ
E	: Eレジスタ
H	: Hレジスタ
L	: Lレジスタ
X	: Xレジスタ
XA	: レジスタ・ペア (XA); 8ビット・アキュムレータ
BC	: レジスタ・ペア (BC)
DE	: レジスタ・ペア (DE)
HL	: レジスタ・ペア (HL)
XA'	: 拡張レジスタ・ペア (XA')
BC'	: 拡張レジスタ・ペア (BC')
DE'	: 拡張レジスタ・ペア (DE')
HL'	: 拡張レジスタ・ペア (HL')
PC	: プログラム・カウンタ
SP	: スタック・ポインタ
CY	: キャリー・フラグ; ビット・アキュムレータ
PSW	: プログラム・ステータス・ワード
MBE	: メモリ・バンク許可フラグ
RBE	: レジスタ・バンク許可フラグ
PORTn	: ポートn (n=0-3, 5, 6, 8, 9)
IME	: 割り込みマスタ許可フラグ
IPS	: 割り込みプライオリティ選択レジスタ
IE x x x	: 割り込み許可フラグ
RBS	: レジスタ・バンク選択レジスタ
MBS	: メモリ・バンク選択レジスタ
PCC	: プロセッサ・クロック・コントロール・レジスタ
.	: アドレス, ビット区切り
(x x)	: x xでアドレスされる内容
x xH	: 16進データ

(3) アドレッシング・エリアの欄の記号説明

* 1	MB = MBE · MBS (MBS = 0, 1, 15)		データ・メモリ・アドレッシング
* 2	MB = 0		
* 3	MBE = 0 : MB = 0 (00H-7FH) MB = 15 (F80H-FFFH) MBE = 1 : MB = MBS (MBS = 0, 1, 15)		
* 4	MB = 15, fmem = FB0H-FBFH, FF0H-FFFH		
* 5	MB = 15, pmem = FC0H-FFFH		
* 6	μ PD753204	addr = 000H-FFFH	プログラム・メモリ・アドレッシング
	μ PD753206	addr = 0000H-17FFH	
	μ PD753208	addr = 0000H-1FFFH	
	μ PD75P3216	addr = 0000H-3FFFH	
* 7	addr = (Current PC) - 15 ~ (Current PC) - 1 (Current PC) + 2 ~ (Current PC) + 16 addr1 = (Current PC) - 15 ~ (Current PC) - 1 (Current PC) + 2 ~ (Current PC) + 16		
* 8	μ PD753204	caddr = 000H-FFFH	
	μ PD753206	caddr = 0000H-0FFFH (PC ₁₂ = 0) or 1000H-17FFH (PC ₁₂ = 1)	
	μ PD753208	caddr = 0000H-0FFFH (PC ₁₂ = 0) or 1000H-1FFFH (PC ₁₂ = 1)	
	μ PD75P3216	caddr = 0000H-0FFFH (PC _{13, 12} = 00B) or 1000H-1FFFH (PC _{13, 12} = 01B) or 2000H-2FFFH (PC _{13, 12} = 10B) or 3000H-3FFFH (PC _{13, 12} = 11B)	
* 9	faddr = 0000H-07FFH		
* 10	taddr = 0020H-007FH		
* 11	μ PD753204	addr1 = 000H-FFFH (Mk モード時のみ)	
	μ PD753206	addr1 = 0000H-17FFH (Mk モード時のみ)	
	μ PD753208	addr1 = 0000H-1FFFH (Mk モード時のみ)	
	μ PD75P3216	addr1 = 0000H-3FFFH (Mk モード時のみ)	

備考1 . MBはアクセス可能なメモリ・バンクを示します。

- 2 . * 2 ではMBE, MBSに関係なくMB = 0です。
- 3 . * 4 , * 5 ではMBE, MBSに関係なくMB = 15です。
- 4 . * 6 ~ * 11は, それぞれアドレッシング可能な領域を示します。

(4) マシン・サイクルの欄の説明

Sは、スキップ付き命令がスキップ動作をするときに要するマシン・サイクル数を示します。Sの値は次のように変わります。

- ・スキップしないとき..... S = 0
- ・スキップされる命令が、1バイト命令、または2バイト命令のとき..... S = 1
- ・スキップされる命令が、3バイト命令^注のとき..... S = 2

注 3バイト命令：BR !addr, BRA !addr1, CALL !addr, CALLA !addr1命令

注意 GETI命令は1マシン・サイクルでスキップされます。

1マシン・サイクルはCPUクロック の1サイクル分 (= t_{CY}) に等しく、PCCの設定により4通りの時間が選択できます(図5 - 12 プロセッサ・クロック・コントロール・レジスタのフォーマット参照)。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
転送	MOV	A, #n4	1	1	A n4		たてづみA
		reg1, #n4	2	2	reg1 n4		
		XA, #n8	2	2	XA n8		たてづみA
		HL, #n8	2	2	HL n8		たてづみB
		rp2, #n8	2	2	rp2 n8		
		A, @HL	1	1	A (HL)	* 1	
		A, @HL +	1	2 + S	A (HL), then L L + 1	* 1	L = 0
		A, @HL -	1	2 + S	A (HL), then L L - 1	* 1	L = FH
		A, @rpa1	1	1	A (rpa1)	* 2	
		XA, @HL	2	2	XA (HL)	* 1	
		@HL, A	1	1	(HL) A	* 1	
		@HL, XA	2	2	(HL) XA	* 1	
		A, mem	2	2	A (mem)	* 3	
		XA, mem	2	2	XA (mem)	* 3	
		mem, A	2	2	(mem) A	* 3	
		mem, XA	2	2	(mem) XA	* 3	
		A, reg	2	2	A reg		
		XA, rp'	2	2	XA rp'		
	reg1, A	2	2	reg1 A			
	rp' 1, XA	2	2	rp' 1 XA			
	XCH	A, @HL	1	1	A (HL)	* 1	
		A, @HL +	1	2 + S	A (HL), then L L + 1	* 1	L = 0
		A, @HL -	1	2 + S	A (HL), then L L - 1	* 1	L = FH
		A, @rpa1	1	1	A (rpa1)	* 2	
		XA, @HL	2	2	XA (HL)	* 1	
		A, mem	2	2	A (mem)	* 3	
		XA, mem	2	2	XA (mem)	* 3	
		A, reg1	1	1	A reg1		
XA, rp'	2	2	XA rp'				

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
テーブル参照	MOVT	XA, @PCDE	1	3	μ PD753204 XA (PC ₁₁₋₈ + DE) _{ROM}		
					μ PD753206, 753208 XA (PC ₁₂₋₈ + DE) _{ROM}		
					μ PD75P3216 XA (PC ₁₃₋₈ + DE) _{ROM}		
		XA, @PCXA	1	3	μ PD753204 XA (PC ₁₁₋₈ + XA) _{ROM}		
					μ PD753206, 753208 XA (PC ₁₂₋₈ + XA) _{ROM}		
					μ PD75P3216 XA (PC ₁₃₋₈ + XA) _{ROM}		
		XA, @BCDE	1	3	XA (BCDE) _{ROM} ^注	* 6	
	XA, @BCXA	1	3	XA (BCXA) _{ROM} ^注	* 6		
ビット転送	MOV1	CY, fmem. bit	2	2	CY (fmem. bit)	* 4	
		CY, pmem. @L	2	2	CY (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀))	* 5	
		CY, @H + mem. bit	2	2	CY (H + mem ₃₋₀ . bit)	* 1	
		fmem. bit, CY	2	2	(fmem. bit) CY	* 4	
		pmem. @L, CY	2	2	(pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) CY	* 5	
		@H + mem. bit, CY	2	2	(H + mem ₃₋₀ . bit) CY	* 1	
演算	ADDS	A, #n4	1	1 + S	A A + n4		carry
		XA, #n8	2	2 + S	XA XA + n8		carry
		A, @HL	1	1 + S	A A + (HL)	* 1	carry
		XA, rp'	2	2 + S	XA XA + rp'		carry
		rp' 1, XA	2	2 + S	rp' 1 rp' 1 + XA		carry
	ADDC	A, @HL	1	1	A, CY A + (HL) + CY	* 1	
		XA, rp'	2	2	XA, CY XA + rp' + CY		
		rp' 1, XA	2	2	rp' 1, CY rp' 1 + XA + CY		

注 μ PD753204を使用する場合は、Bレジスタには“0”を設定してください。

μ PD753206, 753208を使用する場合は、Bレジスタは下位1ビットのみ有効です。

μ PD75P3216を使用する場合は、下位2ビットのみ有効です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件	
演算	SUBS	A, @HL	1	1 + S	A ← A - (HL)	* 1	borrow	
		XA, rp'	2	2 + S	XA ← XA - rp'		borrow	
		rp' 1, XA	2	2 + S	rp' 1 ← rp' 1 - XA		borrow	
	SUBC	A, @HL	1	1	A, CY ← A - (HL) - CY	* 1		
		XA, rp'	2	2	XA, CY ← XA - rp' - CY			
		rp' 1, XA	2	2	rp' 1, CY ← rp' 1 - XA - CY			
	AND	A, #n4	2	2	A ← A & n4			
		A, @HL	1	1	A ← A & (HL)	* 1		
		XA, rp'	2	2	XA ← XA & rp'			
		rp' 1, XA	2	2	rp' 1 ← rp' 1 & XA			
	OR	A, #n4	2	2	A ← A n4			
		A, @HL	1	1	A ← A (HL)	* 1		
		XA, rp'	2	2	XA ← XA rp'			
		rp' 1, XA	2	2	rp' 1 ← rp' 1 XA			
	XOR	A, #n4	2	2	A ← A ⊕ n4			
		A, @HL	1	1	A ← A ⊕ (HL)	* 1		
		XA, rp'	2	2	XA ← XA ⊕ rp'			
		rp' 1, XA	2	2	rp' 1 ← rp' 1 ⊕ XA			
	操作 アキュムレタ	RORC	A	1	1	CY ← A ₀ , A ₃ ← CY, A _{n-1} ← A _n		
		NOT	A	2	2	A ← \bar{A}		
増減	INCS	reg	1	1 + S	reg ← reg + 1		reg = 0	
		rp1	1	1 + S	rp1 ← rp1 + 1		rp1 = 00H	
		@HL	2	2 + S	(HL) ← (HL) + 1	* 1	(HL) = 0	
		mem	2	2 + S	(mem) ← (mem) + 1	* 3	(mem) = 0	
	DECS	reg	1	1 + S	reg ← reg - 1		reg = FH	
		rp'	2	2 + S	rp' ← rp' - 1		rp' = FFH	
比較	SKE	reg, #n4	2	2 + S	Skip if reg = n4		reg = n4	
		@HL, #n4	2	2 + S	Skip if (HL) = n4	* 1	(HL) = n4	
		A, @HL	1	1 + S	Skip if A = (HL)	* 1	A = (HL)	
		XA, @HL	2	2 + S	Skip if XA = (HL)	* 1	XA = (HL)	
		A, reg	2	2 + S	Skip if A = reg		A = reg	
		XA, rp'	2	2 + S	Skip if XA = rp'		XA = rp'	

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
操作リ・フラグ	SET1	CY	1	1	CY 1		
	CLR1	CY	1	1	CY 0		
	SKT	CY	1	1+S	Skip if CY = 1		CY = 1
	NOT1	CY	1	1	CY \overline{CY}		
メモリ・ビット操作	SET1	mem. bit	2	2	(mem. bit) 1	* 3	
		fmem. bit	2	2	(fmem. bit) 1	* 4	
		pmem. @L	2	2	(pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) 1	* 5	
		@H + mem. bit	2	2	(H + mem ₃₋₀ . bit) 1	* 1	
	CLR1	mem. bit	2	2	(mem. bit) 0	* 3	
		fmem. bit	2	2	(fmem. bit) 0	* 4	
		pmem. @L	2	2	(pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) 0	* 5	
		@H + mem. bit	2	2	(H + mem ₃₋₀ . bit) 0	* 1	
	SKT	mem. bit	2	2+S	Skip if (mem. bit) = 1	* 3	(mem. bit) = 1
		fmem. bit	2	2+S	Skip if (fmem. bit) = 1	* 4	(fmem. bit) = 1
		pmem. @L	2	2+S	Skip if (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) = 1	* 5	(pmem. @L) = 1
		@H + mem. bit	2	2+S	Skip if (H + mem ₃₋₀ . bit) = 1	* 1	(@H + mem. bit) = 1
	SKF	mem. bit	2	2+S	Skip if (mem. bit) = 0	* 3	(mem. bit) = 0
		fmem. bit	2	2+S	Skip if (fmem. bit) = 0	* 4	(fmem. bit) = 0
		pmem. @L	2	2+S	Skip if (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) = 0	* 5	(pmem. @L) = 0
		@H + mem. bit	2	2+S	Skip if (H + mem ₃₋₀ . bit) = 0	* 1	(@H + mem. bit) = 0
	SKTCLR	fmem. bit	2	2+S	Skip if (fmem. bit) = 1 and clear	* 4	(fmem. bit) = 1
		pmem. @L	2	2+S	Skip if (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀)) = 1 and clear	* 5	(pmem. @L) = 1
		@H + mem. bit	2	2+S	Skip if (H + mem ₃₋₀ . bit) = 1 and clear	* 1	(@H + mem. bit) = 1
	AND1	CY, fmem. bit	2	2	CY CY (fmem. bit)	* 4	
		CY, pmem. @L	2	2	CY CY (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀))	* 5	
		CY, @H + mem. bit	2	2	CY CY (H + mem ₃₋₀ . bit)	* 1	
	OR1	CY, fmem. bit	2	2	CY CY (fmem. bit)	* 4	
		CY, pmem. @L	2	2	CY CY (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀))	* 5	
CY, @H + mem. bit		2	2	CY CY (H + mem ₃₋₀ . bit)	* 1		
XOR1	CY, fmem. bit	2	2	CY CY ∇ (fmem. bit)	* 4		
	CY, pmem. @L	2	2	CY CY ∇ (pmem ₇₋₂ + L ₃₋₂ . bit (L ₁₋₀))	* 5		
	CY, @H + mem. bit	2	2	CY CY ∇ (H + mem ₃₋₀ . bit)	* 1		

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
分岐	BR ^注	addr	-	-	<p>μPD753204</p> <p>PC₁₁₋₀ addr</p> <p>アセンブラにより、次のうち最適な命令を選択します。</p> <ul style="list-style-type: none"> ・ BRCB !caddr ・ BR \$addr 	* 6	
					<p>μPD753206, 753208</p> <p>PC₁₂₋₀ addr</p> <p>アセンブラにより、次のうち最適な命令を選択します。</p> <ul style="list-style-type: none"> ・ BR !addr ・ BRCB !caddr ・ BR \$addr 		
					<p>μPD75P3216</p> <p>PC₁₃₋₀ addr</p> <p>アセンブラにより、次の命令から最適な命令を選択します。</p> <ul style="list-style-type: none"> ・ BR !addr ・ BRCB !caddr ・ BR \$addr 		

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
分岐	BR ^注	addr1	-	-	μ PD753204 PC ₁₁₋₀ addr1 [アセンブラにより、次のうち最適な命令を選択します。 ・BRA !addr1 ・BRCB !caddr ・BR \$addr1]	* 11	
					μ PD753206, 753208 PC ₁₂₋₀ addr1 [アセンブラにより、次のうち最適な命令を選択します。 ・BR !addr ・BRA !addr1 ・BRCB !caddr ・BR \$addr1]		
					μ PD75P3216 PC ₁₃₋₀ addr1 [アセンブラにより、次の命令から最適な命令を選択します。 ・BR !addr ・BRA !addr1 ・BRCB !caddr ・BR \$addr1]		
		! addr	3	3	μ PD753204 PC ₁₁₋₀ addr μ PD753206, 753208 PC ₁₂₋₀ addr μ PD75P3216 PC ₁₃₋₀ addr	* 6	
		\$addr	1	2	μ PD753204 PC ₁₁₋₀ addr μ PD753206, 753208 PC ₁₂₋₀ addr μ PD75P3216 PC ₁₃₋₀ addr	* 7	

注 で示す部分はMk モード時のみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
分岐	BR ^{注1}	\$addr1	1	2	μPD753204 PC ₁₁₋₀ addr1	* 7	
					μPD753206, 753208 PC ₁₂₋₀ addr1		
					μPD75P3216 PC ₁₃₋₀ addr1		
	PCDE	2	3	μPD753204 PC ₁₁₋₀ PC ₁₁₋₈ + DE			
				μPD753206, 753208 PC ₁₂₋₀ PC ₁₂₋₈ + DE			
				μPD75P3216 PC ₁₃₋₀ PC ₁₃₋₈ + DE			
	PCXA	2	3	μPD753204 PC ₁₁₋₀ PC ₁₁₋₈ + XA			
				μPD753206, 753208 PC ₁₂₋₀ PC ₁₂₋₈ + XA			
				μPD75P3216 PC ₁₃₋₀ PC ₁₃₋₈ + XA			
	BCDE	2	3	μPD753204 PC ₁₁₋₀ BCDE ^{注2}	* 6		
				μPD753206, 753208 PC ₁₂₋₀ BCDE ^{注3}			
				μPD75P3216 PC ₁₃₋₀ BCDE ^{注4}			
BCXA	2	3	μPD753204 PC ₁₁₋₀ BCXA ^{注2}	* 6			
			μPD753206, 753208 PC ₁₂₋₀ BCXA ^{注3}				
			μPD75P3216 PC ₁₃₋₀ BCXA ^{注4}				

注1 . ■で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

- 2 . Bレジスタには必ず“ 0 ”を設定してください。
- 3 . Bレジスタは、下位1ビットのみ有効です。
- 4 . Bレジスタは、下位2ビットのみ有効です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
分岐	BRA ^注	! addr1	3	3	μ PD753204 PC ₁₁₋₀ addr1 μ PD753206, 753208 PC ₁₂₋₀ addr1 μ PD75P3216 PC ₁₃₋₀ addr1	* 11	
	BRCB	! caddr	2	2	μ PD753204 PC ₁₁₋₀ caddr ₁₁₋₀ μ PD753206, 753208 PC ₁₂₋₀ PC ₁₂ + caddr ₁₁₋₀ μ PD75P3216 PC ₁₃₋₀ PC _{13, 12} + caddr ₁₁₋₀	* 8	
サブルーチン・スタック制御	CALLA ^注	! addr1	3	3	μ PD753204 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, 0 PC ₁₁₋₀ addr1, SP SP - 6 μ PD753206, 753208 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, PC ₁₂ PC ₁₂₋₀ addr1, SP SP - 6 μ PD75P3216 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, PC _{13, 12} PC ₁₃₋₀ addr1, SP SP - 6	* 11	

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
サブルーチン・スタック制御	CALL ^注	! addr	3	3	μ PD753204 (SP - 3) MBE, RBE, 0, 0 (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₁₋₀ addr, SP SP - 4	* 6	
					μ PD753206, 753208 (SP - 3) MBE, RBE, 0, PC ₁₂ (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₂₋₀ addr, SP SP - 4		
					μ PD75P3216 (SP - 3) MBE, RBE, PC _{13, 12} (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₃₋₀ addr, SP SP - 4		
				4	μ PD753204 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, 0 PC ₁₁₋₀ addr, SP SP - 6		
					μ PD753206, 753208 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, PC ₁₂ PC ₁₂₋₀ addr, SP SP - 6		
					μ PD75P3216 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, PC _{13, 12} PC ₁₃₋₀ addr, SP SP - 6		

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
サブルーチン・スタック制御	CALLF ^注	! faddr	2	2	μ PD753204 (SP - 3) MBE, RBE, 0, 0 (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₁₋₀ 0 + faddr, SP SP - 4	* 9	
					μ PD753206, 753208 (SP - 3) MBE, RBE, 0, PC ₁₂ (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₂₋₀ 00 + faddr, SP SP - 4		
					μ PD75P3216 (SP - 3) MBE, RBE, PC _{13, 12} (SP - 4) (SP - 1) (SP - 2) PC ₁₁₋₀ PC ₁₃₋₀ 000 + faddr, SP SP - 4		
			3	μ PD753204 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, 0 PC ₁₁₋₀ 0 + faddr, SP SP - 6			
				μ PD753206, 753208 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, 0, PC ₁₂ PC ₁₂₋₀ 00 + faddr, SP SP - 6			
				μ PD75P3216 (SP - 2) x, x, MBE, RBE (SP - 6) (SP - 3) (SP - 4) PC ₁₁₋₀ (SP - 5) 0, 0, PC _{13, 12} PC ₁₃₋₀ 000 + faddr, SP SP - 6			

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
サブルーチン・スタック制御	RET ^注		1	3	μ PD753204 PC ₁₁₋₀ (SP) (SP+3) (SP+2) MBE, RBE, 0, 0 (SP+1), SP SP+4		
					μ PD753206, 753208 PC ₁₁₋₀ (SP) (SP+3) (SP+2) MBE, RBE, 0, PC ₁₂ (SP+1), SP SP+4		
					μ PD75P3216 PC ₁₁₋₀ (SP) (SP+3) (SP+2) MBE, RBE, PC _{13, 12} (SP+1) SP SP+4		
					μ PD753204 x, x, MBE, RBE (SP+4) 0, 0, 0, 0 (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2), SP SP+6		
					μ PD753206, 753208 x, x, MBE, RBE (SP+4) MBE, 0, 0, PC ₁₂ (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2), SP SP+6		
					μ PD75P3216 x, x, MBE, RBE (SP+4) PC ₁₁₋₀ (SP) (SP+3) (SP+2) MBE, 0, PC _{13, 12} (SP+1) SP SP+6		

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	ニモニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
サブルーチン・スタック制御	RETS ^注		1	3+S	μ PD753204 MBE, RBE, 0, 0 (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2) SP SP+4 then skip unconditionally		無条件
					μ PD753206, 753208 MBE, 0, 0, PC ₁₂ (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2) SP SP+4 then skip unconditionally		
					μ PD75P3216 MBE, 0, PC _{13, 12} (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2) SP SP+4 then skip unconditionally		
					μ PD753204 0, 0, 0, 0 (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2) x, x, MBE, RBE (SP+4) SP SP+6 then skip unconditionally		
					μ PD753206, 753208 0, 0, 0, PC ₁₂ (SP+1) PC ₁₁₋₀ (SP) (SP+3) (SP+2) x, x, MBE, RBE (SP+4) SP SP+6 then skip unconditionally		
					μ PD75P3216 x, x, MBE, RBE (SP+4) PC ₁₁₋₀ (SP) (SP+3) (SP+2) 0, 0, PC _{13, 12} (SP+1) SP SP+6 then skip unconditionally		
					μ PD75P3216 x, x, MBE, RBE (SP+4) PC ₁₁₋₀ (SP) (SP+3) (SP+2) 0, 0, PC _{13, 12} (SP+1) SP SP+6 then skip unconditionally		

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
サブルーチン・スタック制御	RET ^注		1	3	μ PD753204 MBE, RBE, 0, 0 (SP + 1) PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
					μ PD753206, 753208 MBE, RBE, 0, PC ₁₂ (SP + 1) PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
					μ PD75P3216 MBE, RBE, PC _{13, 12} (SP + 1) PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
					μ PD753204 0, 0, 0, 0 (SP + 1) PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
					μ PD753206, 753208 0, 0, 0, PC ₁₂ (SP + 1) PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
					μ PD75P3216 0, 0, PC _{13, 12} SP + 1 PC ₁₁₋₀ (SP) (SP + 3) (SP + 2) PSW (SP + 4) (SP + 5), SP SP + 6		
	PUSH	rp	1	1	(SP - 1) (SP - 2) rp, SP SP - 2		
		BS	2	2	(SP - 1) MBS, (SP - 2) RBS, SP SP - 2		
	POP	rp	1	1	rp (SP + 1) (SP), SP SP + 2		
BS		2	2	MBS (SP + 1), RBS (SP), SP SP + 2			
割り込み制御	EI		2	2	IME (IPS. 3) 1		
		IE x x x	2	2	IE x x x 1		
	DI		2	2	IME (IPS. 3) 0		
		IE x x x	2	2	IE x x x 0		

注 で示す部分はMk モード時のみ対応可能です。そのほかはMk モード時にのみ対応可能です。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドレッシング・エリア	スキップ条件
入出力	IN ^注	A, PORT _n	2	2	A PORT _n (n=0-3, 5, 6, 8, 9)		
		XA, PORT _n	2	2	XA PORT _{n+1} , PORT _n (n=8)		
	OUT ^注	PORT _n , A	2	2	PORT _n A (n=2, 3, 5, 6, 8, 9)		
		PORT _n , XA	2	2	PORT _{n+1} , PORT _n XA (n=8)		
CPU制御	HALT		2	2	Set HALT Mode (PCC. 2 1)		
	STOP		2	2	Set STOP Mode (PCC. 3 1)		
	NOP		1	1	No Operation		
特殊	SEL	RB _n	2	2	RBS n (n=0-3)		
		MB _n	2	2	MBS n (n=0, 1, 15)		

注 IN/OUT命令実行時には、MBE = 0 または MBE = 1, MBS = 15 としておく必要があります。

命令群	二モニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
特殊	GETI ^注	taddr	1	3	μ PD753204 ・ TBR命令のとき $PC_{11-0} (taddr)_{3-0} + (taddr + 1)$ ----- ・ TCALL命令のとき $(SP - 4) (SP - 1) (SP - 2) PC_{11-0}$ $(SP - 3) MBE, RBE, 0, 0$ $PC_{11-0} (taddr)_{3-0} + (taddr + 1)$ $SP SP - 4$ ----- ・ TBR, TCALL命令以外 のとき $(taddr) (taddr + 1)$ の命令実行 ----- μ PD753206, 753208 ・ TBR命令のとき $PC_{12-0} (taddr)_{4-0} + (taddr + 1)$ ----- ・ TCALL命令のとき $(SP - 4) (SP - 1) (SP - 2) PC_{11-0}$ $(SP - 3) MBE, RBE, 0, PC_{12}$ $PC_{12-0} (taddr)_{4-0} + (taddr + 1)$ $SP SP - 4$ ----- ・ TBR, TCALL命令以外 のとき $(taddr) (taddr + 1)$ の命令実行 ----- μ PD75P3216 ・ TBR命令のとき $PC_{13-0} (taddr)_{5-0} + (taddr + 1)$ ----- ・ TCALL命令のとき $(SP - 4) (SP - 1) (SP - 2) PC_{11-0}$ $(SP + 1) MBE, RBE, PC_{13, 12}$ $PC_{13-0} (taddr)_{5-0} + (taddr + 1)$ $SP SP - 4$ ----- ・ TBR, TCALL命令以外 のとき $(taddr) (taddr + 1)$ の命令実行	* 10	参照した命令による

注 TBR, TCALL命令はGETI命令のテーブル定義用アセンブラ疑似命令です。

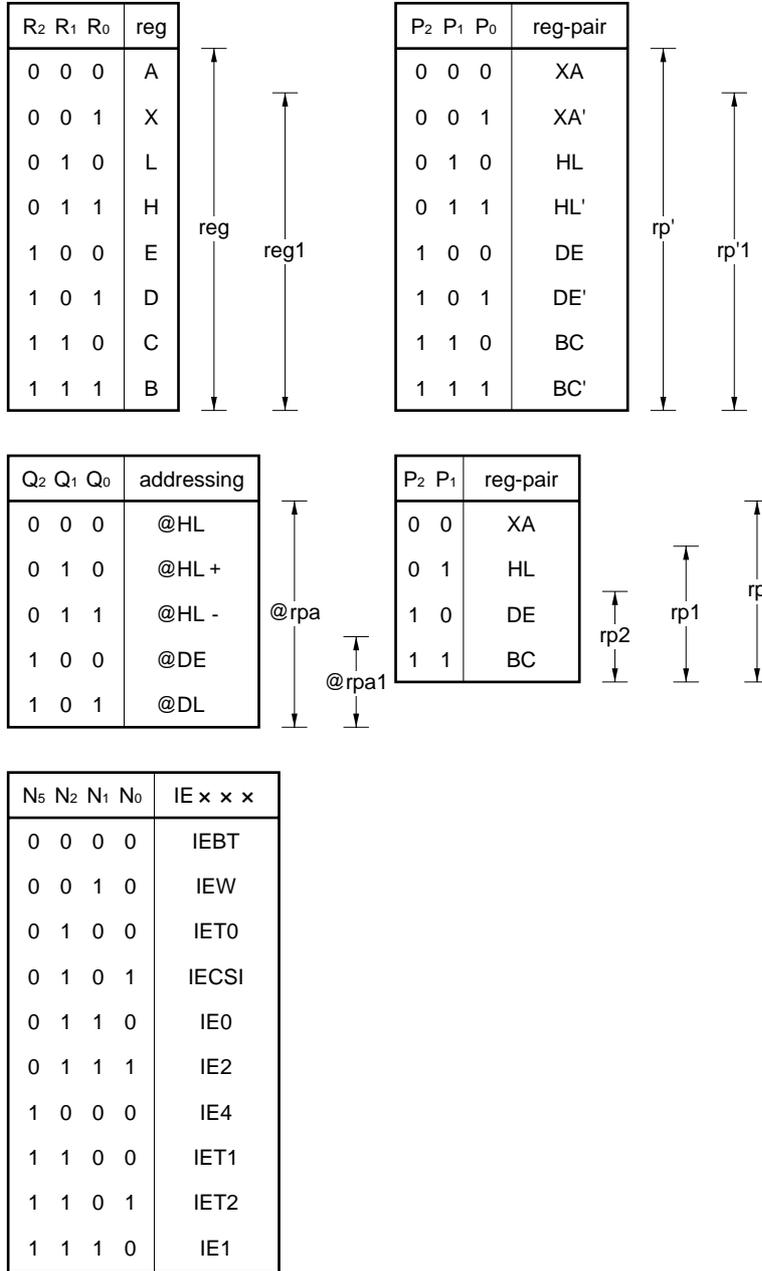
命令群	ニモニック	オペランド	バイト数	マシン・サイクル	オペレーション	アドッシング・エリア	スキップ条件
特殊	GETI ^{注1,2}	taddr	1	3	μ PD753204 ・ TBR命令のとき $PC_{11-0} (taddr)_{3-0} + (taddr + 1)$	* 10	
				4	・ TCALL命令のとき $(SP - 6) (SP - 3) (SP - 4) PC_{11-0}$ $(SP - 5) 0, 0, 0, 0$ $(SP - 2) x, x, MBE, RBE$ $PC_{11-0} (taddr)_{3-0} + (taddr + 1)$ $SP SP - 6$		
				3	・ TBR, TCALL命令以外のとき $(taddr) (taddr + 1)$ の命令実行		参照した命令による
				3	μ PD753206, 753208 ・ TBR命令のとき $PC_{12-0} (taddr)_{4-0} + (taddr + 1)$		
				4	・ TCALL命令のとき $(SP - 6) (SP - 3) (SP - 4) PC_{11-0}$ $(SP - 5) 0, 0, 0, PC_{12}$ $(SP - 2) x, x, MBE, RBE$ $PC_{12-0} (taddr)_{4-0} + (taddr + 1)$ $SP SP - 6$		
				3	・ TBR, TCALL命令以外のとき $(taddr) (taddr + 1)$ の命令実行		参照した命令による
				3	μ PD75P3216 ・ TBR命令のとき $PC_{13-0} (taddr)_{5-0} + (taddr + 1)$		
				4	・ TCALL命令のとき $(SP - 6) (SP - 3) (SP - 4) PC_{11-0}$ $(SP - 5) MBE, RBE, PC_{13, 12}$ $(SP - 2) x, x, MBE, RBE$ $PC_{13-0} (taddr)_{5-0} + (taddr + 1)$ $SP SP - 6$		
3	・ TBR, TCALL命令以外のとき $(taddr) (taddr + 1)$ の命令実行	参照した命令による					

注1 . TBR, TCALL命令はGETI命令のテーブル定義用アセンブラ疑似命令です。

2 . で示す部分はMk モード時のみ対応可能です。そのほかは、Mk モード時にのみ対応可能です。

11.3 各命令の命令コード

(1) 命令コードの記号説明



I_n : n4, n8に対するイミューディエト・データ

D_n : memに対するイミューディエト・データ

B_n : bitに対するイミューディエト・データ

N_n : n, IE x x xに対するイミューディエト・データ

T_n : taddr × $\frac{1}{2}$ に対するイミューディエト・データ

A_n : [分岐先アドレスとの相対アドレス距離 (2 - 16)] - 1 に対するイミューディエト・データ

S_n : [分岐先アドレスとの相対アドレス距離 (15 - 1)] の1の補数に対するイミューディエト・データ

(2) ビット操作アドレッシングの命令コード

オペランドの欄の[* 1]は、以下の3種類があることを示します。

- fmem. bit
- pmem. @L
- @H + mem. bit

上記アドレッシングに対応する命令コードの2バイト目[* 2]は、下表のようになります。

* 1	命令コード・2バイト目	アクセス可能なビット
fmem. bit	1 0 B ₁ B ₀ F ₃ F ₂ F ₁ F ₀	FB0H-FBFHの操作可能ビット
	1 1 B ₁ B ₀ F ₃ F ₂ F ₁ F ₀	FF0H-FFFHの操作可能ビット
pmem. @L	0 1 0 0 G ₃ G ₂ G ₁ G ₀	FC0H-FFFHの操作可能ビット
@H + mem. bit	0 0 B ₁ B ₀ D ₃ D ₂ D ₁ D ₀	アクセス可能なメモリ・バンクの操作可能ビット

B_n : bitに対するイミディエト・データ

F_n : fmemに対するイミディエト・データ
(アドレスの下位4ビットを示す)

G_n : pmemに対するイミディエト・データ
(アドレスのビット5-2を示す)

D_n : memに対するイミディエト・データ
(アドレスの下位4ビットを示す)

命令群	二モニック	オペランド	命令コード		
			B1	B2	B3
転送	MOV	A, #n4	0 1 1 1 l ₃ l ₂ l ₁ l ₀		
		reg1, #n4	1 0 0 1 1 0 1 0	l ₃ l ₂ l ₁ l ₀ 1 R ₂ R ₁ R ₀	
		rp, #n8	1 0 0 0 1 P ₂ P ₁ 1	l ₇ l ₆ l ₅ l ₄ l ₃ l ₂ l ₁ l ₀	
		A, @rpa1	1 1 1 0 0 Q ₂ Q ₁ Q ₀		
		XA, @HL	1 0 1 0 1 0 1 0	0 0 0 1 1 0 0 0	
		@HL, A	1 1 1 0 1 0 0 0		
		@HL, XA	1 0 1 0 1 0 1 0	0 0 0 1 0 0 0 0	
		A, mem	1 0 1 0 0 0 1 1	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		XA, mem	1 0 1 0 0 0 1 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ 0	
		mem, A	1 0 0 1 0 0 1 1	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		mem, XA	1 0 0 1 0 0 1 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ 0	
		A, reg	1 0 0 1 1 0 0 1	0 1 1 1 1 R ₂ R ₁ R ₀	
		XA, rp'	1 0 1 0 1 0 1 0	0 1 0 1 1 P ₂ P ₁ P ₀	
		reg1, A	1 0 0 1 1 0 0 1	0 1 1 1 0 R ₂ R ₁ R ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	0 1 0 1 0 P ₂ P ₁ P ₀	
	XCH	A, @rpa1	1 1 1 0 1 Q ₂ Q ₁ Q ₀		
		XA, @HL	1 0 1 0 1 0 1 0	0 0 0 1 0 0 0 1	
		A, mem	1 0 1 1 0 0 1 1	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		XA, mem	1 0 1 1 0 0 1 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ 0	
		A, reg1	1 1 0 1 1 R ₂ R ₁ R ₀		
		XA, rp'	1 0 1 0 1 0 1 0	0 1 0 0 0 P ₂ P ₁ P ₀	
	MOVT	XA, @PCDE	1 1 0 1 0 1 0 0		
		XA, @PCXA	1 1 0 1 0 0 0 0		
XA, @BCDE		1 1 0 1 0 1 0 1			
XA, @BCXA		1 1 0 1 0 0 0 1			
転送ビット	MOV1	CY, *1	1 0 1 1 1 1 0 1	* 2	
		*1, CY	1 0 0 1 1 0 1 1	* 2	

命令群	二モニック	オペランド	命 令 コ ー ド		
			B ₁	B ₂	B ₃
演算	ADDS	A, #n4	0 1 1 0 l ₃ l ₂ l ₁ l ₀		
		XA, #n8	1 0 1 1 1 0 0 1	l ₇ l ₆ l ₅ l ₄ l ₃ l ₂ l ₁ l ₀	
		A, @HL	1 1 0 1 0 0 1 0		
		XA, rp'	1 0 1 0 1 0 1 0	1 1 0 0 1 P ₂ P ₁ P ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	1 1 0 0 0 P ₂ P ₁ P ₀	
	ADDC	A, @HL	1 0 1 0 1 0 0 1		
		XA, rp'	1 0 1 0 1 0 1 0	1 1 0 1 1 P ₂ P ₁ P ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	1 1 0 1 0 P ₂ P ₁ P ₀	
	SUBS	A, @HL	1 0 1 0 1 0 0 0		
		XA, rp'	1 0 1 0 1 0 1 0	1 1 1 0 1 P ₂ P ₁ P ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	1 1 1 0 0 P ₂ P ₁ P ₀	
	SUBC	A, @HL	1 0 1 1 1 0 0 0		
		XA, rp'	1 0 1 0 1 0 1 0	1 1 1 1 1 P ₂ P ₁ P ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	1 1 1 1 0 P ₂ P ₁ P ₀	
	AND	A, #n4	1 0 0 1 1 0 0 1	0 0 1 1 l ₃ l ₂ l ₁ l ₀	
		A, @HL	1 0 0 1 0 0 0 0		
		XA, rp'	1 0 1 0 1 0 1 0	1 0 0 1 1 P ₂ P ₁ P ₀	
		rp'1, XA	1 0 1 0 1 0 1 0	1 0 0 1 0 P ₂ P ₁ P ₀	
	OR	A, #n4	1 0 0 1 1 0 0 1	0 1 0 0 l ₃ l ₂ l ₁ l ₀	
		A, @HL	1 0 1 0 0 0 0 0		
XA, rp'		1 0 1 0 1 0 1 0	1 0 1 0 1 P ₂ P ₁ P ₀		
rp'1, XA		1 0 1 0 1 0 1 0	1 0 1 0 0 P ₂ P ₁ P ₀		
XOR	A, #n4	1 0 0 1 1 0 0 1	0 1 0 1 l ₃ l ₂ l ₁ l ₀		
	A, @HL	1 0 1 1 0 0 0 0			
	XA, rp'	1 0 1 0 1 0 1 0	1 0 1 1 1 P ₂ P ₁ P ₀		
	rp'1, XA	1 0 1 0 1 0 1 0	1 0 1 1 0 P ₂ P ₁ P ₀		
アキュムレータ操作	RORC	A	1 0 0 1 1 0 0 0		
	NOT	A	1 0 0 1 1 0 0 1	0 1 0 1 1 1 1 1	

命令群	二モニック	オペランド	命 令 コ ー ド		
			B ₁	B ₂	B ₃
増減	INCS	reg	1 1 0 0 0 R ₂ R ₁ R ₀		
		rp1	1 0 0 0 1 P ₂ P ₁ 0		
		@HL	1 0 0 1 1 0 0 1	0 0 0 0 0 0 1 0	
		mem	1 0 0 0 0 0 1 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
	DECS	reg	1 1 0 0 1 R ₂ R ₁ R ₀		
		rp'	1 0 1 0 1 0 1 0	0 1 1 0 1 P ₂ P ₁ P ₀	
比較	SKE	reg, #n4	1 0 0 1 1 0 1 0	l ₃ l ₂ l ₁ l ₀ 0 R ₂ R ₁ R ₀	
		@HL, #n4	1 0 0 1 1 0 0 1	0 1 1 0 l ₃ l ₂ l ₁ l ₀	
		A, @HL	1 0 0 0 0 0 0 0		
		XA, @HL	1 0 1 0 1 0 1 0	0 0 0 1 1 0 0 1	
		A, reg	1 0 0 1 1 0 0 1	0 0 0 0 1 R ₂ R ₁ R ₀	
		XA, rp'	1 0 1 0 1 0 1 0	0 1 0 0 1 P ₂ P ₁ P ₀	
キャリー・フラグ操作	SET1	CY	1 1 1 0 0 1 1 1		
	CLR1	CY	1 1 1 0 0 1 1 0		
	SKT	CY	1 1 0 1 0 1 1 1		
	NOT1	CY	1 1 0 1 0 1 1 0		
メモリ・ビット操作	SET1	mem. bit	1 0 B ₁ B ₀ 0 1 0 1	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		* 1	1 0 0 1 1 1 0 1	* 2	
	CLR1	mem. bit	1 0 B ₁ B ₀ 0 1 0 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		* 1	1 0 0 1 1 1 0 0	* 2	
	SKT	mem. bit	1 0 B ₁ B ₀ 0 1 1 1	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		* 1	1 0 1 1 1 1 1 1	* 2	
	SKF	mem. bit	1 0 B ₁ B ₀ 0 1 1 0	D ₇ D ₆ D ₅ D ₄ D ₃ D ₂ D ₁ D ₀	
		* 1	1 0 1 1 1 1 1 0	* 2	
	SKTCLR	* 1	1 0 0 1 1 1 1 1	* 2	
	AND1	CY, * 1	1 0 1 0 1 1 0 0	* 2	
	OR1	CY, * 1	1 0 1 0 1 1 1 0	* 2	
	XOR1	CY, * 1	1 0 1 1 1 1 0 0	* 2	

命令群	二モニック	オペランド	命令コード			
			B ₁	B ₂	B ₃	
分岐	BR	! addr	1 0 1 0 1 0 1 1	0 0 _____	— addr _____	
		\$ addr	(+ ₃ ¹⁶)	0 0 0 0 A ₃ A ₂ A ₁ A ₀		
			(- ₃ ¹)	1 1 1 1 S ₃ S ₂ S ₁ S ₀		
		PCDE	1 0 0 1 1 0 0 1	0 0 0 0 0 1 0 0		
		PCXA	1 0 0 1 1 0 0 1	0 0 0 0 0 0 0 0		
		BCDE	1 0 0 1 1 0 0 1	0 0 0 0 0 1 0 1		
	BCXA	1 0 0 1 1 0 0 1	0 0 0 0 0 0 0 1			
分岐	BRA	! addr1	1 0 1 1 1 0 1 0	0 _____	addr1 _____	
	BRCB	! caddr	0 1 0 1 _____	— caddr _____		
サブリーチン・スタック	CALLA	! addr1	1 0 1 1 1 0 1 1	0 _____	addr _____	
	CALL	! addr	1 0 1 0 1 0 1 1	0 1 _____	addr _____	
	CALLF	! faddr	0 1 0 0 0 _____	— faddr _____		
	RET		1 1 1 0 1 1 1 0			
	RETS		1 1 1 0 0 0 0 0			
	RETI		1 1 1 0 1 1 1 1			
	PUSH	rp	0 1 0 0 1 P ₂ P ₁ 1			
		BS	1 0 0 1 1 0 0 1	0 0 0 0 0 1 1 1		
POP	rp	0 1 0 0 1 P ₂ P ₁ 0				
	BS	1 0 0 1 1 0 0 1	0 0 0 0 0 1 1 0			
入出力	IN	A, PORT _n	1 0 1 0 0 0 1 1	1 1 1 1 N ₃ N ₂ N ₁ N ₀		
		XA, PORT _n	1 0 1 0 0 0 1 0	1 1 1 1 N ₃ N ₂ N ₁ N ₀		
	OUT	PORT _n , A	1 0 0 1 0 0 1 1	1 1 1 1 N ₃ N ₂ N ₁ N ₀		
		PORT _n , XA	1 0 0 1 0 0 1 0	1 1 1 1 N ₃ N ₂ N ₁ N ₀		
割り込み制御	EI		1 0 0 1 1 1 0 1	1 0 1 1 0 0 1 0		
		IE x x x	1 0 0 1 1 1 0 1	1 0 N ₅ 1 1 N ₂ N ₁ N ₀		
	DI		1 0 0 1 1 1 0 0	1 0 1 1 0 0 1 0		
		IE x x x	1 0 0 1 1 1 0 0	1 0 N ₅ 1 1 N ₂ N ₁ N ₀		
CPU制御	HALT		1 0 0 1 1 1 0 1	1 0 1 0 0 0 1 1		
	STOP		1 0 0 1 1 1 0 1	1 0 1 1 0 0 1 1		
	NOP		0 1 1 0 0 0 0 0			
特殊	SEL	RB _n	1 0 0 1 1 0 0 1	0 0 1 0 0 0 N ₁ N ₀		
		MB _n	1 0 0 1 1 0 0 1	0 0 0 1 N ₃ N ₂ N ₁ N ₀		
	GETI	taddr	0 0 T ₅ T ₄ T ₃ T ₂ T ₁ T ₀			

★

11.4 命令機能と応用

この項では、命令の機能と応用を説明します。μPD753204, 753206, 753208, 75P3216のMk モードとMk モードでは、使用できる命令、または命令の機能が異なりますので、次の読み方に従って読んでください。

読み方

- 次のすべてに共通して使用できます。

μPD753204	}	のMk モード時とMk モード時
μPD753206		
μPD753208		
μPD75P3216		
- μPD753204, 753206, 753208, 75P3216のMk モード時のみ使用できます。
- μPD753204, 753206, 753208, 75P3216のMk モード時のみ使用できます。
- / μPD753204, 753206, 753208, 75P3216のMk モード時とMk モード時に共通して使用することができますが、Mk モードとMk モードでは、機能面で異なる箇所があります。Mk モードの場合は【Mk モード】、Mk モードの場合は【Mk モード】に示す説明書きを、それぞれの場合に応じて、読み分けてください。

備考 この項で示されている機能は、プログラム・カウンタが13ビットのμPD753206, 753208で説明してあります。μPD753204のプログラム・カウンタは12ビット、μPD75P3216のプログラム・カウンタは14ビットですので、それぞれ読み替えてお使いください。

11.4.1 転送命令

○ MOV A, #n4

機能 : A n4 n4 = 13-0 : 0-FH

4ビット・イミューディエト・データn4をAレジスタ(4ビット・アキュムレータ)に転送します。たてづみ効果(グループA)があり,同じ命令か,またはMOV XA, #n8命令が続けて置かれると実行した命令以降のたてづみ命令はNOPとして処理されます。

応用例

(1) アキュムレータに0BHをセットする。

```
MOV A, #0BH
```

(2) ポート3に出力するデータを0-2から選ぶ。

```
A0 : MOV A, #0
```

```
A1 : MOV A, #1
```

```
A2 : MOV A, #2
```

```
OUT PORT3, A
```

○ MOV reg1, #n4

機能 : reg1 n4 n4 = 13-0 0-FH

4ビット・イミューディエト・データn4をAレジスタreg1(X, H, L, D, E, B, C)に転送します。

○ MOV XA, #n8

機能 : XA n8 n8 = 17-0 : 00H-FFH

8ビット・イミューディエト・データn8をレジスタ・ペアXAに転送します。たてづみ効果があり,同じ命令か,またはMOV A, #n4命令が続けて置かれると実行した命令以降のたてづみ命令は, NOPとして処理されます。

○ MOV HL, #n8

機能: HL n8 n8 = 17-0 : 00H-FFH

8ビット・イミディエト・データn8をレジスタ・ペアHLに転送します。たてづみ効果があり、同じ命令が続けて置かれると実行した命令以降のたてづみ命令は、NOPとして処理されます。

○ MOV rp2, #n8

機能: rp2 n8 n8 = 17-0 : 00H-FFH

8ビット・イミディエト・データn8をレジスタ・ペアrp2 (BC, DE) に転送します。

○ MOV A, @HL

機能: A (HL)

レジスタ・ペアHLでアドレスされるデータ・メモリの内容をAレジスタに転送します。

○ MOV A, @HL +

機能: A (HL), L L + 1
skip if L = 0H

レジスタ・ペアHLでアドレスされるデータ・メモリの内容をAレジスタに転送します。そのあとLレジスタの内容を自動的に+ 1し、その結果が0Hになると、続く1命令をスキップします。

○ MOV A, @HL -

機能: A (HL), L L - 1
skip if L = FH

レジスタ・ペアHLでアドレスされるデータ・メモリの内容をAレジスタに転送します。そのあとLレジスタの内容を自動的に- 1し、その結果がFHになると、続く1命令をスキップします。

○ MOV A, @rpa1

機能 : A (rpa)

rpa = HL + のとき : skip if L = 0

rpa = HL - のとき : skip if L = FH

レジスタ・ペアrpa (HL, HL +, HL -, DE, DL) でアドレスされるデータ・メモリの内容をAレジスタに転送します。

rpaとしてオートインクリメント (HL +) を指定した場合は、データの転送後、Lレジスタの内容を自動的に + 1 し、その結果が0になると続く1命令をスキップします。

rpaとしてオートデクリメント (HL -) を指定した場合は、データの転送後、Lレジスタの内容を自動的に - 1 し、その結果がFHになると続く1命令をスキップします。

○ MOV XA, @HL

機能 : A (HL), X (HL + 1)

レジスタ・ペアHLでアドレスされるデータ・メモリの内容をAレジスタに転送し、メモリの次の番地の内容をXレジスタに転送します。

ただし、Lレジスタの内容が奇数の場合、最下位ビットが無視されたアドレスが転送されます。

応用例

3EH, 3FH番地のデータをレジスタ・ペアXAに転送する。

```
MOV HL, #3EH
```

```
MOV XA, @HL
```

○ MOV @HL, A

機能 : (HL) A

Aレジスタの内容をレジスタ・ペアHLでアドレスされるデータ・メモリへ転送します。

○ MOV @HL, XA

機能 : (HL) ← A, (HL+1) ← X

Aレジスタの内容をレジスタ・ペアHLでアドレスされるデータ・メモリに転送し、Xレジスタの内容をメモリの次の番地に転送します。

ただし、Lレジスタの内容が奇数の場合は、最下位ビットを無視したアドレスが指定されます。

○ MOV A, mem

機能 : A ← (mem) mem = D7-0 : 00H-FFH

8ビット・イミディエト・データmemでアドレスされるデータ・メモリの内容をAレジスタに転送します。

○ MOV XA, mem

機能 : A ← (mem), X ← (mem+1) mem = D7-0 : 00H-FEH

8ビット・イミディエト・データmemでアドレスされるデータ・メモリの内容をAレジスタに、その次の番地の内容をXレジスタに転送します。

memで指定できるアドレスは偶数アドレスです。

応用例

40H, 41H番地のデータをレジスタ・ペアXAに転送する。

```
MOV XA, 40H
```

○ MOV mem, A

機能 : (mem) ← A mem = D7-0 : 00H-FFH

Aレジスタの内容を8ビット・イミディエト・データmemでアドレスされるデータ・メモリに転送します。

○ MOV mem, XA

機能 : (mem) A, (mem + 1) X mem = D7-0 : 00H-FEH

Aレジスタの内容を8ビット・イミディエト・データmemでアドレスされるデータ・メモリに転送し、Xレジスタの内容をメモリの次の番地に転送します。

memで指定できるアドレスは偶数アドレスです。

○ MOV A, reg

機能 : A reg

レジスタreg (X, A, H, L, D, E, B, C) の内容をAレジスタに転送します。

○ MOV XA, rp'

機能 : XA rp'

レジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容をレジスタ・ペアXAへ転送します。

応用例

レジスタ・ペアXA'のデータをレジスタ・ペアXAに転送する。

```
MOV XA, XA'
```

○ MOV reg1, A

機能 : reg1 A

Aレジスタの内容をレジスタreg1 (X, H, L, D, E, B, C) に転送します。

○ MOV rp'1, XA

機能 : rp'1 XA

レジスタ・ペアXAの内容をレジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') に転送します。

○ XCH A, @HL

機能: A (HL)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容を交換します。

○ XCH A, @HL +

機能: A (HL), L L + 1

skip if L = 0H

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容を交換します。そのあとLレジスタの内容を自動的に + 1 し、その結果が0Hになると、続く 1 命令をスキップします。

○ XCH A, @HL -

機能: A (HL), L L - 1

skip if L = FH

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容を交換します。そのあとLレジスタの内容を自動的に - 1 し、その結果がFHになると、続く 1 命令をスキップします。

○ XCH A, @rpa1

機能: A (rpa)

rpa = HL + のとき: skip if L = 0

rpa = HL - のとき: skip if L = FH

Aレジスタの内容とレジスタ・ペアrpa (HL, HL +, HL -, DE, DL) でアドレスされるデータ・メモリの内容を交換します。rpaとしてオートインクリメント (HL +) / オートデクリメント (HL -) を指定した場合は、データの交換後、Lレジスタの内容を自動的に + 1 / - 1 し、その結果がHL + の場合は0に、HL - の場合はFHになると続く 1 命令をスキップします。

応用例

データ・メモリの20H-2FH番地のデータと30H-3FH番地のデータを交換する。

```

SEL   MB0
MOV   D, # 2
MOV   HL, #30H
LOOP: XCH  A, @HL    ; A   (3×)
      XCH  A, @DL    ; A   (2×)
      XCH  A, @HL+   ; A   (3×)
BR    LOOP

```

○ XCH XA, @HL

機能: A (HL), X (HL+1)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容を交換し、Xレジスタの内容とメモリの次の番地の内容を交換します。

ただし、Lレジスタの内容が奇数の場合は、最下位ビットを無視したアドレスが指定されます。

○ XCH A, mem

機能: A (mem) mem = D7-0 : 00H-FEH

Aレジスタの内容と8ビット・イミディエト・データmemでアドレスされるデータ・メモリの内容を交換します。

○ XCH XA, mem

機能: A (mem), X (mem+1) mem = D7-0 : 00H-FEH

Aレジスタの内容と8ビット・イミディエト・データmemでアドレスされるデータ・メモリの内容を交換し、Xレジスタの内容とメモリの次の番地の内容を交換します。

memで指定できるアドレスは偶数アドレスです。

○ XCH A, reg1

機能 : A reg1

Aレジスタの内容とレジスタreg1 (X, H, L, D, E, B, C) の内容を交換します。

○ XCH XA, rp'

機能 : XA rp'

レジスタ・ペアXAの内容とレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容を交換します。

11.4.2 テーブル参照命令

○ MOV_T XA, @PCDE

機能：μPD753206, 753208の場合 XA ROM (PC₁₂₋₈ + DE)

プログラム・カウンタ (PC) の下位 8 ビット (PC₇₋₀) をレジスタ・ペア DE の内容で置き換えたときにアドレスされるプログラム・メモリ内のテーブル・データの下位 4 ビットを A レジスタに、上位 4 ビットを X レジスタに転送します。

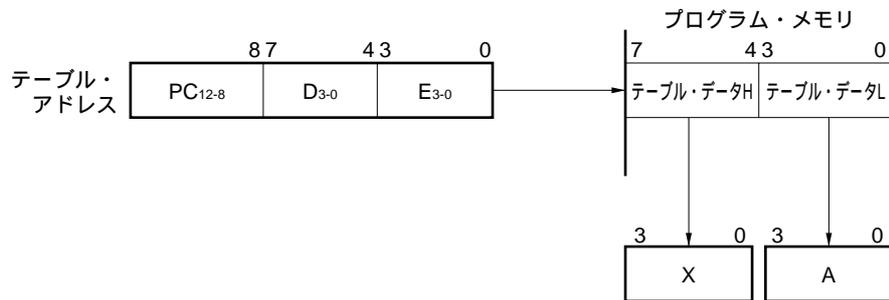
テーブル・アドレスはこの命令実行時のプログラム・カウンタ (PC) の内容によって決定されます。

テーブル領域には、必要なデータをあらかじめアセンブラ疑似命令 (DB 命令) によりプログラムしておく必要があります。

プログラム・カウンタはこの命令の実行により影響を受けません。

この命令は、テーブル・データを連続して参照する場合に有効です。

例 μPD753206, 753208の場合



備考 この項で示されている機能は、プログラム・カウンタが13ビットのμPD753206, 753208で説明してあります。μPD753204のプログラム・カウンタは12ビット、μPD75P3216のプログラム・カウンタは14ビットですので、それぞれ読み替えてお使いください。

○ MOVT XA, @PCXA

機能 : μ PD753206, 753208の場合 XA ROM ($PC_{12-8} + XA$)

プログラム・カウンタ (PC) の下位 8 ビット (PC_{7-0}) をレジスタ・ペアXAの内容で置き換えたときアドレスされるプログラム・メモリ内のテーブル・データの下位 4 ビットをAレジスタに, 上位 4 ビットをXレジスタに転送します。

テーブル・アドレスはこの命令実行時のPCの内容によって決まります。

テーブル領域には, 必要なデータをあらかじめアセンブラ疑似命令 (DB命令) によりプログラムしておく必要があります。

PCは, この命令の実行により影響を受けません。

注 意

MOV_T XA, @PCDEと同様に, 命令が $x \times FFH$ 番地にある場合は, 次のページのテーブル・データが転送されます。

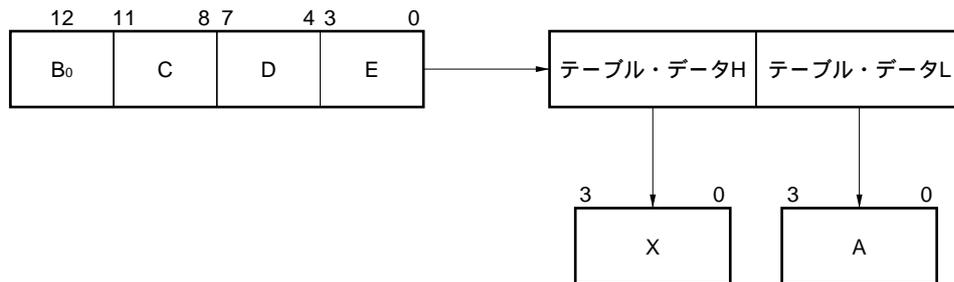
備考 この項で示されている機能は, プログラム・カウンタが13ビットの μ PD753206, 753208で説明してあります。 μ PD753204のプログラム・カウンタは12ビット, μ PD75P3216のプログラム・カウンタは14ビットですので, それぞれ読み替えてお使いください。

○ MOVT XA, @BCDE

機能：μPD753206, 753208の場合 XA ROM (B₀ + CDE)

レジスタBの下位1ビットとレジスタC, D, Eの内容でアドレスされるプログラム・メモリ内のテーブル・データ(8ビット)の下位4ビットをAレジスタに, 上位4ビットをXレジスタに転送します。

テーブル領域には必要なデータをあらかじめアセンブラ疑似命令(DB命令)によりプログラムしておく必要があります。PCは, この命令実行により影響を受けません。

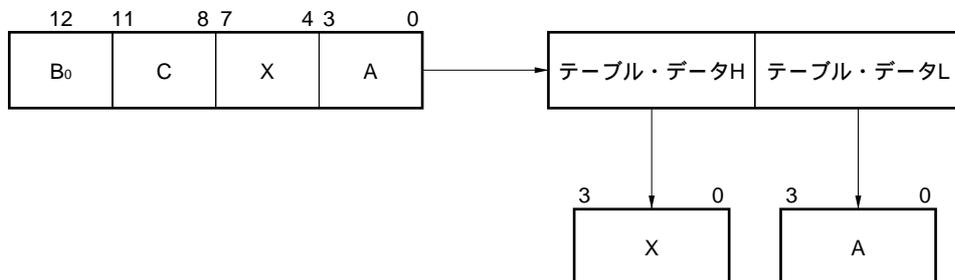


○ MOVT XA, @BCXA

機能：μPD753206, 753208の場合 XA ROM (B₀ + CXA)

レジスタBの下位1ビットとレジスタC, X, Aの内容でアドレスされるプログラム・メモリ内のテーブル・データ(8ビット)の下位4ビットをAレジスタに, 上位4ビットをXレジスタに転送します。

テーブル領域には必要なデータをあらかじめアセンブラ疑似命令(DB命令)によりプログラムしておく必要があります。PCは, この命令実行により影響を受けません。



備考 この項で示されている機能は, プログラム・カウンタが13ビットのμPD753206, 753208で説明してあります。μPD753204のプログラム・カウンタは12ビット, μPD75P3216のプログラム・カウンタは14ビットですので, それぞれ読み替えてお使いください。

11.4.3 ビット転送命令

- MOV1 CY, fmem. bit
- MOV1 CY, pmem. @L
- MOV1 CY, @H + mem. bit

機能 : CY (オペランドにより指定されたビット)

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリ・ビットの内容をキャリー・フラグ (CY) に転送します。

- MOV1 fmem. bit, CY
- MOV1 pmem. @L, CY
- MOV1 @H + mem. bit, CY

機能 : (オペランドで指定されるビット) CY

キャリー・フラグ (CY) の内容をビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリ・ビットに転送します。

応用例

データ・メモリの3FH番地のビット3のフラグを、ポート3のビット2に出力する。

```

FLAG EQU 3FH. 3
SEL   MB0
MOV   H, #FLAG SHR 6 ; H FLAGの上位4ビット
MOV1  CY, @H+FLAG   ; CY FLAG
MOV1  PORT3. 2, CY   ; P32 CY

```

11.4.4 演算命令

○ ADDS A, #n4

機能 : A A + n4 ; Skip if carry. n4 = l3-0 : 0-FH

Aレジスタの内容に4ビット・イミディエト・データn4を2進加算し、加算の結果キャリーが出たときは続く1命令をスキップします。キャリー・フラグは影響を受けません。

ADDC A, @HL命令およびSUBC A, @HL命令と組み合わせると、進数補正命令となります(11.1.4 進数補正命令を参照)。

○ ADDS XA, #n8

機能 : XA XA + n8 ; Skip if carry. n8 = l7-0 : 00H-FFH

レジスタ・ペアXAの内容に8ビット・イミディエト・データn8を2進加算し、加算の結果キャリーが出たときは続く1命令をスキップします。キャリー・フラグは影響を受けません。

○ ADDS A, @HL

機能 : A A + (HL) ; Skip if carry.

Aレジスタの内容にレジスタ・ペアHLでアドレスされるデータ・メモリの内容を2進加算し、加算の結果キャリーが出たときは続く1命令をスキップします。キャリー・フラグは影響を受けません。

○ ADDS XA, rp'

機能 : XA XA + rp' ; Skip if carry.

レジスタ・ペアXAの内容にレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC')の内容を2進加算し、加算の結果キャリーが出たときは続く1命令をスキップします。キャリー・フラグは影響を受けません。

○ ADDS rp'1, XA

機能 : rp' $rp'1 + XA$; Skip if carry.

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容にレジスタ・ペアXAの内容を2進加算し、加算の結果キャリーが出たときは続く1命令をスキップします。キャリー・フラグは影響を受けません。

応用例

レジスタ・ペアを左シフトする。

```
MOV  XA, rp'1
ADDS rp'1, XA
NOP
```

○ ADDC A, @HL

機能 : A, CY $A + (HL) + CY$

Aレジスタの内容にレジスタ・ペアHLでアドレスされるデータ・メモリの内容をキャリー・フラグも含めて2進加算します。加算の結果キャリーが出たらキャリー・フラグがセットされ、キャリーが出ないときはキャリー・フラグはリセットされます。

この命令の次にADDS A, #n4命令が置かれた場合、この命令にキャリーが発生するとADDS A, #n4命令をスキップします。キャリーが発生しないときはADDS A, #n4命令を実行し、ADDS A, #n4命令のスキップ機能を禁止するという機能が生じます。したがって、これらの命令の組み合わせを進数補正に利用できます(11.1.4 **進数補正命令**を参照)。

○ ADDC XA, rp'

機能 : XA, CY $XA + rp' + CY$

レジスタ・ペアXAの内容にレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容をキャリー・フラグも含めて2進加算します。加算の結果キャリーが出たとき、キャリー・フラグはセットされ、キャリーが出ないとき、キャリー・フラグはリセットされます。

○ ADDC rp'1, XA

機能 : rp'1, CY rp'1 + XA + CY

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容にレジスタ・ペアXAの内容をキャリー・フラグを含めて2進加算します。加算の結果キャリーが出たらキャリー・フラグがセットされます。キャリーが出ないときはキャリー・フラグはリセットされます。

○ SUBS A, @HL

機能 : A A - (HL) ; Skip if borrow

Aレジスタの内容からレジスタ・ペアHLでアドレスされるデータ・メモリの内容を減算し、結果をAレジスタにセットします。減算の結果ボローが出ると続く1命令をスキップします。

キャリー・フラグには影響を与えません。

○ SUBS XA, rp'

機能 : XA XA - rp' ; Skip if borrow

レジスタ・ペアXAの内容からレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容を減算し、結果をレジスタ・ペアXAにセットします。減算の結果ボローが出ると続く1命令をスキップします。

キャリー・フラグには影響を与えません。

応用例

データ・メモリとレジスタ・ペアの大小比較をする。

```
MOV  XA, mem
SUBS XA, rp'
      ; (mem) > rp'
      ; (mem) < rp'
```

○ SUBS rp'1, XA

機能 : rp'1 rp'1 + XA ; Skip if borrow

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容からレジスタ・ペアXAの内容を減算し、結果を指定されたレジスタ・ペアrp'1にセットします。減算の結果ボローが出ると続く1命令をスキップします。

キャリー・フラグには影響を与えません。

○ SUBC A, @HL

機能 : A, CY A - (HL) - CY

Aレジスタの内容からレジスタ・ペアHLでアドレスされるデータ・メモリの内容をキャリー・フラグを含めて減算し、結果をAレジスタにセットします。減算の結果ボローが出るとき、キャリー・フラグがセットされ、ボローが出ないときキャリー・フラグはリセットされます。

この命令の次にADDS A, #n4命令が置かれた場合、この命令にボローが出なければADDS A, #n4命令をスキップします。ボローが出るとADDS A, #n4命令を実行し、ADDS A, #n4命令のスキップ機能を禁止するという機能が生じます。したがって、これらの命令の組み合わせを進数補正に利用できます (11.1.4 進数補正命令を参照)。

○ SUBC XA, rp'

機能 : XA, CY XA - rp' - CY

レジスタ・ペアXAの内容からレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容をキャリー・フラグを含めて減算し、結果をレジスタ・ペアXAにセットします。減算の結果ボローが出るとキャリー・フラグがセットされ、ボローが出ないときはキャリー・フラグはリセットされます。

○ SUBC rp'1, XA

機能 : rp'1, CY rp'1 - XA - CY

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容からレジスタ・ペアXAの内容をキャリー・フラグを含めて減算し、結果を指定されたレジスタ・ペアrp'1にセットされます。減算の結果ポローが出るとキャリー・フラグがセットされ、ポローが出ないときはキャリー・フラグはリセットされません。

○ AND A, #n4

機能 : A A n4 n4 = I3:0 : 0-FH

Aレジスタの内容と4ビット・イミディエイト・データn4との論理積をとり、結果をAレジスタにセットします。

応用例

アキュムレータの上位2ビットを0にする。

```
AND A, #0011B
```

○ AND A, @HL

機能 : A A (HL)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容との論理積をとり、結果をAレジスタにセットします。

○ AND XA, rp'

機能 : XA XA rp'

レジスタ・ペアXAの内容とレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容との論理積をとり、結果をレジスタ・ペアXAにセットします。

○ AND rp'1, XA

機能 : rp'1 rp'1 XA

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容とレジスタ・ペアXAの内容との論理積をとり、結果を指定されたレジスタ・ペアにセットします。

○ OR A, #n4

機能 : A A n4 n4 = I3-0 : 0-FH

Aレジスタの内容と4ビット・イミディエト・データn4との論理和をとり、結果をAレジスタにセットします。

応用例

アキュムレータの下位3ビットを1にする。

```
OR A, #0111B
```

○ OR A, @HL

機能 : A A (HL)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容と論理和をとり、結果をAレジスタにセットします。

○ OR XA, rp'

機能 : XA XA rp'

レジスタ・ペアXAの内容とレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容との論理和をとり、結果をレジスタ・ペアXAにセットします。

○ OR rp'1, XA

機能: rp'1 ← rp'1 OR XA

レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容とレジスタ・ペアXAの内容との論理和をとり、結果をレジスタ・ペアrp'1にセットします。

○ XOR A, #n4

機能: A ← A XOR n4 (n4 = 13:0 : 0-FH)

Aレジスタの内容と4ビット・イミューディエト・データn4との排他的論理和をとり、結果をAレジスタにセットします。

応用例

アキュムレータの最上位4ビットを反転する。

```
XOR A, #1000B
```

○ XOR A, @HL

機能: A ← A XOR (HL)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容との排他的論理和をとり、結果をAレジスタにセットします。

○ XOR XA, rp'

機能: XA ← XA XOR rp'

レジスタ・ペアXAの内容とレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容との排他的論理和をとり、結果をレジスタ・ペアXAにセットします。

 XOR rp'1, XA

機能 : rp'1 ← rp'1 ⊕ XA

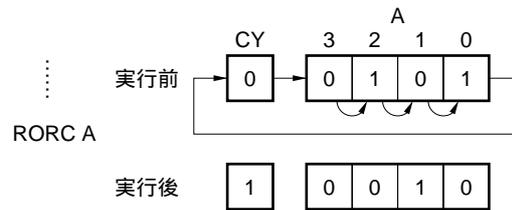
レジスタ・ペアrp'1 (HL, DE, BC, XA', HL', DE', BC') の内容とレジスタ・ペアXAの内容との排他的論理和をとり、結果をレジスタ・ペアrp'1にセットします。

11.4.5 アキュムレータ操作命令

○ RORC A

機能: CY A₀, A_{n-1} A_n, A₃ CY (n=1-3)

Aレジスタ(4ビット・アキュムレータ)の内容をキャリー・フラグも含めて1ビットずつ右回転します。



○ NOT A

機能: A \bar{A}

Aレジスタ(4ビット・アキュムレータ)の1の補数をとります(各ビットを反転させます)。

11.4.6 増減命令

○ INCS reg

機能 : reg reg + 1 ; Skip if reg = 0

レジスタreg (X, A, H, L, D, E, B, C) の内容をインクリメントします。インクリメントの結果 reg = 0 になると続く 1 命令をスキップします。

○ INCS rp1

機能 : rp1 rp1 + 1 ; Skip if rp1 = 00H

レジスタ・ペアrp1 (HL, DE, BC) の内容をインクリメントします。インクリメントの結果rp1 = 00H になると続く 1 命令をスキップします。

○ INCS @HL

機能 : (HL) (HL) + 1 ; Skip if (HL) = 0

レジスタ・ペアHLでアドレスされるデータ・メモリの内容をインクリメントします。インクリメントの結果そのデータ・メモリの内容が 0 になると続く 1 命令をスキップします。

○ INCS mem

機能 : (mem) (mem) + 1 ; Skip if (mem) = 0, mem = D7-0 : 00H-FFH

8ビット・イミディエイト・データmemでアドレスされるデータ・メモリの内容をインクリメントします。インクリメントの結果、そのデータ・メモリの内容が 0 になると続く 1 命令をスキップします。

○ DECS reg

機能 : reg = reg - 1 ; Skip if reg = FFH

レジスタreg (X, A, H, L, D, E, B, C) の内容をデクリメントします。デクリメントの結果reg = FFHになると、続く1命令をスキップします。

○ DECS rp'

機能 : rp' = rp' - 1 ; Skip if rp' = FFH

レジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容をデクリメントします。デクリメントの結果rp' = FFHになると、続く1命令をスキップします。

11.4.7 比較命令

○ SKE reg, #n4

機能 : Skip if reg = n4 n4 = I3-0 : 0-FH

レジスタreg (X, A, H, L, D, E, B, C) の内容と4ビット・イミディエト・データn4が等しければ, 続く1命令をスキップします。

○ SKE @HL, #n4

機能 : Skip if (HL) = n4 n4 = I3-0 : 0-FH

レジスタ・ペアHLでアドレスされるデータ・メモリの内容と4ビット・イミディエト・データn4が等しければ, 続く1命令をスキップします。

○ SKE A, @HL

機能 : Skip if A = (HL)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容が等しければ, 続く1命令をスキップします。

○ SKE XA, @HL

機能 : Skip if A = (HL) and X = (HL + 1)

Aレジスタの内容とレジスタ・ペアHLでアドレスされるデータ・メモリの内容が等しく, かつXレジスタの内容とメモリの次の番地の内容が等しければ, 続く1命令をスキップします。

ただし, Lレジスタの内容が奇数の場合, 最下位ビットを無視したアドレスが指定されます。

SKE A, reg

機能 : Skip if A = reg

Aレジスタの内容とレジスタreg (X, A, H, L, D, E, B, C) の内容が等しければ, 続く 1 命令をスキップします。

 SKE XA, rp'

機能 : Skip if XA = rp'

レジスタ・ペアXAの内容とレジスタ・ペアrp' (XA, HL, DE, BC, XA', HL', DE', BC') の内容が等しければ, 続く 1 命令をスキップします。

11.4.8 キャリー・フラグ操作命令

SET1 CY

機能: CY 1

キャリー・フラグをセットします。

CLR1 CY

機能: CY 0

キャリー・フラグをクリアします。

SKT CY

機能: Skip if CY = 1

キャリー・フラグが1のとき、続く1命令をスキップします。

NOT1 CY

機能: CY $\overline{\text{CY}}$

キャリー・フラグを反転させます。0なら1に、1なら0になります。

11.4.9 メモリ・ビット操作命令

 SET1 mem. bit

機能：(mem. bit) 1 mem = D7-0 : 00H-FFH, bit = B1-0 : 0-3

8ビット・イミディエト・データmemで示されるアドレスの、2ビット・イミディエト・データbitで指定されるビットをセットします。

 SET1 fmem. bit SET1 pmem. @L SET1 @H + mem. bit

機能：(オペランドで指定されるビット) 1

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリのビットをセットします。

 CLR1 mem. bit

機能：(mem. bit) 0 mem = D7-0 : 00H-FFH, bit = B1-0 : 0-3

8ビット・イミディエト・データmemで示されるアドレスの、2ビット・イミディエト・データbitで指定されるビットをクリアします。

 CLR1 fmem. bit CLR1 pmem. @L CLR1 @H + mem. bit

機能：(オペランドで指定されるビット) 0

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリのビットをクリアします。

SKT mem. bit

機能 : Skip if (mem. bit) = 1

mem = D7-0 : 00H-FFH, bit = B1-0 : 0-3

8ビット・イミディエト・データmemで示されるアドレスの、2ビット・イミディエト・データbitで指定されるビットが1なら、続く1命令をスキップします。

 SKT fmem. bit SKT pmem. @L SKT @H + mem. bit

機能 : Skip if (オペランドで指定されるビット) = 1

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリのビットが1ならば、続く1命令をスキップします。

 SKF mem. bit

機能 : Skip if (mem. bit) = 0

mem = D7-0 : 00H-FFH, bit = B1-0 : 0-3

8ビット・イミディエト・データmemで示されるアドレスの、2ビット・イミディエト・データbitで指定されるビットが0ならば、続く1命令をスキップします。

 SKF fmem. bit SKF pmem. @L SKF @H + mem. bit

機能 : Skip if (オペランドで指定されるビット) = 0

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) により指定されるデータ・メモリのビットの内容が0ならば、続く1命令をスキップします。

- SKTCLR fmem. bit
- SKTCLR pmem. @L
- SKTCLR @H + mem. bit

機能 : Skip if (オペランドで指定されるビット) = 1 then clear

ビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) で指定されるデータ・メモリのビットが1なら、続く1命令をスキップして、そのビットを“0”にクリアします。

- AND1 CY, fmem. bit
- AND1 CY, pmem. @L
- AND1 CY, @H + mem. bit

機能 : CY CY (オペランドで指定されるビット)

キャリー・フラグの内容とビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) で指定されるデータ・メモリの内容との論理積をとり、結果をキャリー・フラグにセットします。

- OR1 CY, fmem. bit
- OR1 CY, pmem. @L
- OR1 CY, @H + mem. bit

機能 : CY CY (オペランドで指定されるビット)

キャリー・フラグの内容とビット操作アドレッシング (fmem. bit, pmem. @L, @H + mem. bit) で指定されるデータ・メモリのビットの内容との論理和をとり、結果をキャリー・フラグにセットします。

- XOR1 CY, fmem. bit
- XOR1 CY, pmem. @L
- XOR1 CY, @H+mem. bit

機能 : CY ← CY − (オペランドで指定されるビット)

キャリア・フラグの内容と、ビット操作アドレッシング (fmem. bit, pmem. @L, @H+mem. bit) で指定されるデータ・メモリのビットの内容との排他的論理和をとり、結果をキャリア・フラグにセットします。

11.4.10 分岐命令

○ BR addr

機能： μ PD753208の場合 PC₁₂₋₀ addr
addr = 0000H-1FFFH

イミディエト・データaddrでアドレスされるアドレスに分岐します。
この命令はアセンブラの疑似命令であり、アセンブル時、アセンブラよりBR !addr命令、BRCB !caddr命令、BR \$addr命令の中から最適な命令に自動的に置き換えられます。

○ BR addr1

機能： μ PD753208の場合 PC₁₂₋₀ addr1
addr1 = 0000H-1FFFH

イミディエト・データaddr1でアドレスされるアドレスに分岐します。
この命令はアセンブラの疑似命令であり、アセンブル時、アセンブラよりBRA !addr1命令、BR !addr命令、BRCB !caddr命令、BR \$addr1命令の中から最適な命令に自動的に置き換えられます。

○ BRA !addr1

機能： μ PD753208の場合 PC₁₂₋₀ addr1

○ BR !addr

機能： μ PD753208の場合 PC₁₂₋₀ addr
addr = 0000H-1FFFH

イミディエト・データaddrがプログラム・カウンタ(PC)へ転送され、PCでアドレスされる番地に分岐します。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、addr = 0000H-1FFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット、addr = 000H-FFFH、 μ PD753206はプログラム・カウンタが13ビット、addr = 0000H-17FFFH、 μ PD75P3216はプログラム・カウンタが14ビット、addr = 0000H-3FFFHですので、それぞれ読み替えてお使いください。

○ BR \$ addr

機能： μ PD753208の場合 PC₁₂₋₀ addr

addr = (PC - 15) ~ (PC - 1) , (PC + 2) ~ (PC + 16)

現在のアドレスから (- 15 ~ - 1) , (+ 2 ~ + 16) の分岐範囲を持つ相対分岐命令です。ページの境界や、ブロックの境界には影響を受けません。

○ BR \$ addr1

機能： μ PD753208の場合 PC₁₂₋₀ addr1

addr1 = (PC - 15) ~ (PC - 1) , (PC + 2) ~ (PC + 16)

現在のアドレスから (- 15 ~ - 1) , (+ 2 ~ + 16) の分岐範囲を持つ相対分岐命令です。ページの境界や、ブロックの境界には影響を受けません。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、addr = 0000H-1FFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット、addr = 000H-FFFH、 μ PD753206はプログラム・カウンタが13ビット、addr = 0000H-17FFFH、 μ PD75P3216はプログラム・カウンタが14ビット、addr = 0000H-3FFFHですので、それぞれ読み替えてお使いください。

○ BRCB ! caddr

機能 : μ PD753208の場合 $PC_{12-0} \rightarrow PC_{12} + caddr_{11-0}$
 $caddr = n000H-nFFFH$
 $n = PC_{12} = 0, 1$

プログラム・カウンタの下位12ビット (PC_{11-0}) を12ビット・イミディエト・データcaddrで置き換えた番地に分岐します。

注 意

BRCB ! caddr命令は、通常、その命令のあるブロック内に分岐しますが、1バイト目が、0FFE H番地、または0FFF H番地にある場合は、ブロック0には分岐せずブロック1に分岐しますので注意してください。



上図のaまたはbの位置にBRCB ! caddr命令がある場合は、ブロック0ではなくブロック1に分岐します。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、 $addr = 0000H-1FFFH$ の μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット、 $addr = 000H-FFFH$ 、 μ PD753206はプログラム・カウンタが13ビット、 $addr = 0000H-17FFFH$ 、 μ PD75P3216はプログラム・カウンタが14ビット、 $addr = 0000H-3FFFH$ ですので、それぞれ読み替えてお使いください。

○ BR PCDE

機能： μ PD753208の場合 PC₁₂₋₀ PC₁₂₋₈ + DE
PC₇₋₄ D, PC₃₋₀ E

プログラム・カウンタの下位8ビット(PC₇₋₀)をレジスタ・ペアDEの内容で置き換えた番地に分岐します。プログラム・カウンタの上位のビットには影響を与えません。

注 意

BR PCDE命令は、通常、その命令のあるページ内に分岐しますが、命令コードの1バイト目が××FE番地、または××FFH番地にある場合は、そのページ内には分岐せず次のページに分岐します。



たとえば、上図のようにaまたはbの位置にBR PCDE命令がある場合は、ページ2ではなくページ3の、レジスタ・ペアDEの内容により指定される下位8ビット・アドレスへ分岐します。

○ BR PCXA

機能： μ PD753208の場合 PC₁₂₋₀ PC₁₂₋₈ + XA
PC₇₋₄ X, PC₃₋₀ A

プログラム・カウンタの下位8ビット(PC₇₋₀)をレジスタ・ペアXAの内容で置き換えた番地に分岐します。プログラム・カウンタの上位のビットには影響を与えません。

注 意

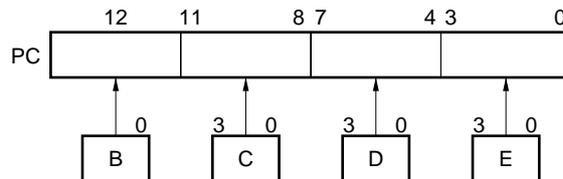
BR PCDE命令と同様に、1バイト目が××FEH、または××FFH番地にある場合は同一ページ内には分岐せず、次のページに分岐します。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、addr = 0000H-1FFFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット、addr = 000H-FFFFH、 μ PD753206はプログラム・カウンタが13ビット、addr = 0000H-17FFFH、 μ PD75P3216はプログラム・カウンタが14ビット、addr = 0000H-3FFFFHですので、それぞれ読み替えてお使いください。

○ BR BCDE

機能： μ PD753208の場合 PC₁₂₋₀ B₀ + CDE

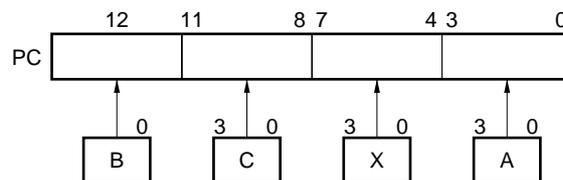
プログラム・カウンタをレジスタB₀, C, D, Eの内容で置き換えた番地に分岐します。



○ BR BCXA

機能： μ PD753208の場合 PC₁₂₋₀ B₀ + CXA

プログラム・カウンタをレジスタB₀, C, X, Aの内容で置き換えた番地に分岐します。



○ TBR addr

機能：

GETI命令のテーブル定義用アセンブラ疑似命令です。3バイトのBR !addr命令を、1バイトのGETI命令で置き換える場合に使用します。addrには12ビット・アドレス・データを記述します。詳しくは、RA75X **アセンブラ・パッケージ ユーザーズ・マニュアル 言語編**を参照してください。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、addr = 0000H-1FFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット、addr = 000H-FFFH、 μ PD753206はプログラム・カウンタが13ビット、addr = 0000H-17FFFH、 μ PD75P3216はプログラム・カウンタが14ビット、addr = 0000H-3FFFHですので、それぞれ読み替えてお使いください。

11.4.11 サブルーチン・スタック制御命令

○ CALLA ! addr1

機能: μ PD753208の場合(SP - 2) x, x, MBE, RBE, (SP - 3) PC₇₋₄(SP - 4) PC₃₋₀, (SP - 5) 0, 0, 0, PC₁₂(SP - 6) PC₁₁₋₈PC₁₂₋₀ addr1, SP SP - 6

/ ○ CALL ! addr

機能: μ PD753208の場合

【Mk モード】

(SP - 1) PC₇₋₄, (SP - 2) PC₃₋₀(SP - 3) MBE, RBE, 0, PC₁₂(SP - 4) PC₁₁₋₈, PC₁₂₋₀ addr, SP SP - 4

addr = 0000H-1FFFFH

【Mk モード】

(SP - 2) x, x, MBE, RBE

(SP - 3) PC₇₋₄, (SP - 4) PC₃₋₀(SP - 5) 0, 0, 0, PC₁₂, (SP - 6) PC₁₁₋₈PC₁₂₋₀ addr, SP SP - 6

addr = 0000H-1FFFFH

プログラム・カウンタ(リターン・アドレス)およびMBE, RBEの内容をスタック・ポインタ(SP)でアドレスされるデータ・メモリ(スタック)に退避し, SPをデクリメントしたのち, 14ビット・イミューディアト・データaddrでアドレスされる番地へ分岐します。

備考 この項で示されている機能は, プログラム・カウンタ: 13ビット, addr = 0000H-1FFFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット, addr = 000H-FFFFH, μ PD753206はプログラム・カウンタが13ビット, addr = 0000H-17FFFH, μ PD75P3216はプログラム・カウンタが14ビット, addr = 0000H-3FFFFHですので, それぞれ読み替えてお使いください。

① CALLF ! faddr

機能：μPD753208の場合

【Mk モード】

(SP - 1) PC₇₋₄, (SP - 2) PC₃₋₀

(SP - 3) MBE, RBE, 0, PC₁₂

(SP - 4) PC₁₁₋₈, SP SP - 4

PC₁₂₋₀ 00 + faddr

faddr = 0000H-07FFH

【Mk モード】

(SP - 2) x, x, MBE, RBE

(SP - 3) PC₇₋₄, (SP - 4) PC₃₋₀

(SP - 5) 0, 0, 0, PC₁₂, (SP - 6) PC₁₁₋₈

SP SP - 6

PC₁₂₋₀ 00 + faddr

faddr = 0000H-07FFH

プログラム・カウンタ (PC ; リターン・アドレス) および MBE, RBE の内容をスタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) に退避し, SP をデクリメントしたのち, 11 ビット・イミディエト・データ faddr でアドレスされる番地へ分岐します。コールできる範囲は 0000H-07FFH (0-2047) 番地に限られます。

○ TCALL ! addr

機能

GETI 命令のテーブル定義用アセンブラ疑似命令です。3 バイトの CALL ! addr 命令を, 1 バイトの GETI 命令で置き換える場合に使用します。addr には 12 ビット・アドレス・データを記述します。詳しくは, RA75X **アセンブラ・パッケージ ユーザーズ・マニュアル 言語編** を参照してください。

備考 この項で示されている機能は, プログラム・カウンタ : 13 ビット, addr = 0000H-1FFFH の μPD753208 で説明してあります。μPD753204 はプログラム・カウンタが 12 ビット, addr = 000H-FFFH, μPD753206 はプログラム・カウンタが 13 ビット, addr = 0000H-17FFFH, μPD75P3216 はプログラム・カウンタが 14 ビット, addr = 0000H-3FFFH ですので, それぞれ読み替えてお使いください。

/ RET

機能：μPD753208の場合

【Mk モード】 PC₁₁₋₈ (SP), MBE, RBE, 0, PC₁₂ (SP + 1)
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3), SP SP + 4

【Mk モード】 PC₁₁₋₈ (SP), MBE, 0, 0, PC₁₂ (SP + 1)
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3)
 x, x, MBE, RBE (SP + 4)
 SP SP + 6

スタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) の内容をプログラム・カウンタ (PC), メモリ・バンク許可フラグ (MBE), およびレジスタ・バンク許可フラグ (RBE) にリストアし, そのあとSPの内容をインクリメントします。

注 意

プログラム・ステータス・ワード (PSW) は, MBE, RBE以外リストアされません。

/ RETS

機能：μPD753208の場合

【Mk モード】 PC₁₁₋₈ (SP), MBE, 0, 0, PC₁₂ (SP + 1)
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3), SP SP + 4
 Then skip unconditionally

【Mk モード】 PC₁₁₋₈ (SP), 0, 0, 0, PC₁₂ (SP + 1)
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3)
 x, x, MBE, RBE (SP + 4), SP SP + 6
 Then skip unconditionally

スタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) の内容をプログラム・カウンタ (PC), メモリ・バンク許可フラグ (MBE), およびレジスタ・バンク許可フラグ (RBE) にリストアし, SPの内容をインクリメント後, 無条件にスキップします。

注 意

プログラム・ステータス・ワード (PSW) は, MBE, RBE以外リストアされません。

備考 この項で示されている機能は, プログラム・カウンタ: 13ビット, addr = 0000H-1FFFHのμPD753208で説明してあります。μPD753204はプログラム・カウンタが12ビット, addr = 000H-FFFH, μPD753206はプログラム・カウンタが13ビット, addr = 0000H-17FFFH, μPD75P3216はプログラム・カウンタが14ビット, addr = 0000H-3FFFHですので, それぞれ読み替えてお使いください。

RETl

機能： μ PD753208の場合

【Mk モード】 PC₁₁₋₈ (SP), MBE, RBE, 0, PC₁₂ (SP + 1),
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3)
 PSW_L (SP + 4), PSW_H (SP + 5)
 SP SP + 6

【Mk モード】 PC₁₁₋₈ (SP), 0, 0, 0, PC₁₂ (SP + 1),
 PC₃₋₀ (SP + 2), PC₇₋₄ (SP + 3)
 PSW_L (SP + 4), PSW_H (SP + 5)
 SP SP + 6

スタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) の内容をプログラム・カウンタ (PC), およびプログラム・ステータス・ワードにリストアし, そのあとSPの内容をインクリメントします。

割り込み処理ルーチンからの復帰時に使用します。

備考 この項で示されている機能は, プログラム・カウンタ: 13ビット, addr = 0000H-1FFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット, addr = 000H-FFFH, μ PD753206はプログラム・カウンタが13ビット, addr = 0000H-17FFFH, μ PD75P3216はプログラム・カウンタが14ビット, addr = 0000H-3FFFHですので, それぞれ読み替えてお使いください。

○ PUSH rp

機能 : (SP - 1) rp_H, (SP - 2) rp_L, SP SP - 2

レジスタ・ペアrp (XA, HL, DE, BC)の内容をスタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) に退避し, そのあとSPをデクリメントします。

レジスタ・ペアの上位側 (rp_H : X, H, D, B) は (SP - 1) で, 下位側 (rp_L : A, L, E, C) は (SP - 2) でアドレスされるスタックに退避されます。

○ PUSH BS

機能 : (SP - 1) MBS, (SP - 2) RBS, SP SP - 2

メモリ・バンク選択レジスタ (MBS), およびレジスタ・バンク選択レジスタ (RBS) の内容をスタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) に退避し, そのあとSPをデクリメントします。

○ POP rp

機能 : rp_L (SP), rp_H (SP + 1), SP SP + 2

スタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) の内容をレジスタ・ペアrp (XA, HL, DE, BC) へ復帰し, そのあとSPをインクリメントします。

レジスタ・ペアの下位側 (rp_L : A, L, E, C) には (SP) の内容が, 上位側 (rp_H : X, H, D, B) には (SP + 1) の内容がリストアされます。

○ POP BS

機能 : RBS (SP), MBS (SP + 1), SP SP + 2

スタック・ポインタ (SP) でアドレスされるデータ・メモリ (スタック) の内容をレジスタ・バンク選択レジスタ (RBS), メモリ・バンク選択レジスタ (MBS) へ復帰し, そのあとSPをインクリメントします。

11.4.12 割り込み制御命令

EI

機能：IME (IPS.3) 1

割り込みマスタ許可フラグ (割り込みプライオリティ選択レジスタのビット3) を, “ 1 ” にセットし, 割り込みを許可します。割り込み受け付けの可否は, 各割り込み許可フラグにより制御されます。

EI IE × × ×

機能：IE × × × 1 × × × = N₅, N₂₋₀

割り込み許可フラグ (IE × × ×) を “ 1 ” にセットし, 割り込みを受け付け可能状態にします。
(× × × = BT, CSI, T0, T, T2, W, 0, 2, 4)

DI

機能：IME (IPS.3) 0

割り込みマスタ許可フラグ (割り込みプライオリティ選択レジスタのビット3) を, “ 0 ” にリセットし, 各割り込み許可フラグの内容に関係なく, すべての割り込みを禁止します。

DI IE × × ×

機能：IE × × × 0 × × × = N₅, N₂₋₀

割り込み許可フラグ (IE × × ×) を “ 0 ” にリセットし, 割り込みの受け付けを禁止します。
(× × × = BT, CSI, T0, T1, T2, W, 0, 2, 4)

11.4.13 入出力命令

○ IN A, PORTn

機能 : A PORTn $n = N_{3:0} : 0-3, 5, 6, 8, 9$

PORTn ($n = 0-3, 5, 6, 8, 9$) で指定されるポートの内容をAレジスタに転送します。

注 意

この命令実行時には、 $MBE = 0$ とするか、もしくは ($MBE = 1, MBS = 15$) としておく必要があります。nには0-3, 5, 6, 8, 9のみ指定できます。

入出力モードの指定により、出力ラッチのデータ (出力モード) または、端子のデータ (入力モード) が取り込まれます。

○ IN XA, PORTn

機能 : A PORTn, X PORTn+1 $n = N_{3:0} : 8$

PORTn ($n = 8$) で指定されるポートの内容をAレジスタに転送し、次のポートの内容をXレジスタに転送します。

注 意

nとしては、8のみ指定できます。この命令実行時には、 $MBE = 0$ とするか、もしくは ($MBE = 1, MBS = 15$) としておく必要があります。

入出力モードの指定により、出力ラッチのデータ (出力モード) または、端子のデータ (入力モード) が取り込まれます。

○ OUT PORT_n, A

機能 : PORT_n A n = N₃₋₀ : 2, 3, 5, 6, 8, 9

Aレジスタの内容をPORT_n (n = 2, 3, 5, 6, 8, 9) で指定されるポートの出力ラッチへ転送します。

注 意

この命令実行時には、MBE = 0 とするか、もしくは (MBE = 1, MBS = 15) としておく必要があります。

nには2, 3, 5, 6, 8, 9のみ指定できます。

○ OUT PORT_n, XA

機能 : PORT_n A, PORT_{n+1} X n = N₃₋₀ : 8

Aレジスタの内容をPORT_n (n = 8) で指定されるポートの出力ラッチへ転送し、Xレジスタの内容を次のポートの出力ラッチへ転送します。

注 意

この命令実行時には、MBE = 0 とするか、もしくは (MBE = 1, MBS = 15) としておく必要があります。

nとしては、8のみ指定できます。

11.4.14 CPU制御命令

 HALT

機能：PCC.2 1

HALTモードをセットします（プロセッサ・クロック・コントロール・レジスタのビット2をセットする命令です）。

注 意

HALT命令に続く1命令は必ずNOP命令にしてください。

 STOP

機能：PCC.3 1

STOPモードをセットします（プロセッサ・クロック・コントロール・レジスタのビット3をセットする命令です）。

注 意

STOP命令に続く1命令は必ずNOP命令にしてください。

 NOP

機能：何もせず1マシン・サイクル費やします。

11.4.15 特殊命令

○ SEL R_{Bn}機能：RBS n $n = N_{1-0} : 0-3$ 2ビット・イミディエト・データ n をレジスタ・バンク選択レジスタ (RBS) にセットします。○ SEL M_{Bn}機能：MBS n $n = N_{3-0} : 0, 1, 15$ 4ビット・イミディエト・データ n をメモリ・バンク選択レジスタ (MBS) に転送します。

/ GETI taddr

機能： μ PD753208の場合taddr = T₅₋₀, 0 : 20H-7FH

【Mk モード】

TBR命令で定義したテーブルを参照した場合

PC₁₂₋₀ (taddr)₄₋₀ + (taddr + 1)

TCALL命令で定義したテーブルを参照した場合

(SP - 1) PC₇₋₄, (SP - 2) PC₃₋₀(SP - 3) MBE, RBE, 0, PC₁₂(SP - 4) PC₁₁₋₈PC₁₂₋₀ (taddr)₄₋₀ + (taddr + 1)

SP SP - 4

TBR, TCALL命令以外で定義したテーブルを参照した場合

(taddr) (taddr + 1) を命令コードとする命令を実行

備考 この項で示されている機能は、プログラム・カウンタ：13ビット，addr = 0000H-1FFFHの μ PD753208で説明してあります。 μ PD753204はプログラム・カウンタが12ビット，addr = 000H-FFFH， μ PD753206はプログラム・カウンタが13ビット，addr = 0000H-17FFFH， μ PD75P3216はプログラム・カウンタが14ビット，addr = 0000H-3FFFFHですので，それぞれ読み替えてお使いください。

【Mk モード】

TBR命令で定義したテーブルを参照した場合^注

PC₁₂₋₀ (taddr)₄₋₀ + (taddr + 1)

TCALL命令で定義したテーブルを参照した場合^注

(SP - 2) x, x, MBE, RBE

(SP - 3) PC₇₋₄, (SP - 4) PC₃₋₀

(SP - 5) 0, 0, 0, PC₁₂, (SP - 6) PC₁₁₋₈

PC₁₂₋₀ (taddr)₄₋₀ + (taddr + 1)

SP SP - 6

TBR, TCALL命令以外で定義したテーブルを参照した場合

(taddr) (taddr + 1) を命令コードとする命令を実行

注 TBR, TCALL命令で指定されるアドレスは0000H-3FFFHに限ります。

(taddr), (taddr + 1) で指定されるプログラム・メモリ・アドレスにある2バイトのデータを参照し、命令として実行します。

参照テーブルの領域は0020H-007FHで、あらかじめデータを書き込んでおきます。データの書き込みは、1バイト命令、2バイト命令の場合は、二モニックをそのまま記述します。

3バイト・コール命令および3バイト分岐命令の場合は、アセンブラ疑似命令(TCALL, TBR)により記述します。

taddrとして指定できるアドレスは偶数アドレスのみです。

備考 この項で示されている機能は、プログラム・カウンタ：13ビット、addr = 0000H-1FFFHのμPD753208で説明してあります。μPD753204はプログラム・カウンタが12ビット、addr = 000H-FFFH、μPD753206はプログラム・カウンタが13ビット、addr = 0000H-17FFFH、μPD75P3216はプログラム・カウンタが14ビット、addr = 0000H-3FFFHですので、それぞれ読み替えてお使いください。

注 意

参照テーブルにセットできる2バイト命令は、2マシン・サイクルの命令に限られます。(ただし、BRCB命令およびCALLF命令を除く)。また、1バイト命令2つをセットする場合は、次表のような組み合わせに限られます。

1バイト目の命令	2バイト目の命令							
MOV A, @HL MOV @HL, A XCH A, @HL	<table style="border: none;"> <tr><td rowspan="3" style="font-size: 2em; vertical-align: middle;">{</td><td>INCS L</td></tr> <tr><td>DECS L</td></tr> <tr><td>INCS H</td></tr> <tr><td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td><td>DECS H</td></tr> <tr><td>INCS HL</td></tr> </table>	{	INCS L	DECS L	INCS H	{	DECS H	INCS HL
{	INCS L							
	DECS L							
	INCS H							
{	DECS H							
	INCS HL							
MOV A, @DE XCH A, @DE	<table style="border: none;"> <tr><td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td><td>INCS E</td></tr> <tr><td>DECS E</td></tr> <tr><td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td><td>INCS D</td></tr> <tr><td>DECS D</td></tr> <tr><td>INCS DE</td></tr> </table>	{	INCS E	DECS E	{	INCS D	DECS D	INCS DE
{	INCS E							
	DECS E							
{	INCS D							
	DECS D							
INCS DE								
MOV A, @DL XCH A, @DL	<table style="border: none;"> <tr><td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td><td>INCS L</td></tr> <tr><td>DECS L</td></tr> <tr><td rowspan="2" style="font-size: 2em; vertical-align: middle;">{</td><td>INCS D</td></tr> <tr><td>DECS D</td></tr> </table>	{	INCS L	DECS L	{	INCS D	DECS D	
{	INCS L							
	DECS L							
{	INCS D							
	DECS D							

PCは、GETI命令実行中はインクリメントは行いませんので、参照命令実行後、GETI命令の次の番地から処理を続けます。

GETI命令の前の命令がスキップ機能を持つ場合、GETI命令はほかの1バイト命令と同様にスキップされます。また、GETI命令で参照した命令がスキップ機能を持つ場合、GETI命令の次に続く命令はスキップされます。

GETI命令でたてづみ効果を持つ命令を参照した場合は次のように実行されます。

GETI命令の前の命令も同じグループのたてづみ効果を持つ命令の場合、GETI命令を実行するとたてづみ効果は消え、参照した命令はスキップされません。

GETI命令の次の命令も同じグループのたてづみ効果を持つ命令の場合、参照した命令により生ずるたてづみ効果は有効で、次に続く命令はスキップされます。

応用例

```
{ MOV HL, #00H
  MOV XA, #FFH
  CALL SUB1
  BR SUB2 }
```

 をGETI命令に置き換える。

```
                ORG    20H
HL00 :          MOV    HL, #00H
XAFF :          MOV    XA, #FFH
CSUB1 :         TCALL  SUB1
BSUB2 :         TBR    SUB2
                .....
                GETI  HL00      ; MOV HL, #00H
                .....
                GETI  BSUB2     ; BR SUB2
                .....
                GETI  CSUB1     ; CALL SUB1
                .....
                GETI  XAFF      ; MOV XA, #FFH
```

付録A μ PD753108, 753208, 75P3216の機能一覧表

項 目		μ PD753108	μ PD753208	μ PD75P3216
プログラム・メモリ		マスクROM 0000H-1FFFH (8192 × 8 ビット)		ワン・タイムPROM 0000H-3FFFH (16384 × 8 ビット)
データ・メモリ		000H-1FFFH (512 × 4 ビット)		
CPU		75XL CPU		
命令実行時間	メイン・システム・クロック 選択時	・ 0.95, 1.91, 3.81, 15.3 μ s (4.19 MHz動作時) ・ 0.67, 1.33, 2.67, 10.7 μ s (6.0 MHz動作時)		
	サブシステム・クロック選択 時	122 μ s (32.768 kHz動作 時)	なし	
★ 入出力ポート	CMOS入力	8本 (ソフトウェアによる プルアップ抵抗内蔵可能： 7本)	6本 (ソフトウェアによるプルアップ抵抗内蔵可能：5 本)	
	CMOS入出力	20本 (ソフトウェアによるプルアップ抵抗内蔵可能)		
	N-chオープン・ドレイン入出 力	4本 (マスク・オプションによるプルアップ抵抗内蔵可 能, 13V耐圧)	4本 (マスク・オプション なし, 13V耐圧)	
	合計	32本	30本	
LCDコントローラ/ドライバ		セグメント数選択：16/20/ 24セグメント (4本単位で CMOS入出力ポートに切り 替え可能, 最大8本)	セグメント数選択：4/8/12セグメント (4本単位でCMOS 入出力ポートに切り替え可能, 最大8本)	
		表示モード選択：スタティック, 1/2デューティ (1/2バイアス), 1/3デューティ (1/2 バイアス), 1/3デューティ (1/3バイアス), 1/4デューティ (1/3バイアス)		
		LCD駆動用分割抵抗をマスク・オプションで内蔵可能		LCD駆動用分割抵抗なし
タイマ		5チャンネル ・ 8ビット・タイマ/イベ ント・カウンタ：3チャ ネル ・ ベーシック・インターバ ル・タイマ/ウォッチ ドッグ・タイマ：1チャ ネル ・ 時計用タイマ：1チャ ネル	5チャンネル ・ 8ビット・タイマ/イベント・カウンタ：1チャンネル ・ 8ビット・タイマ・カウンタ：2チャンネル (16ビット・タイマ・カウンタ, キャリア・ジェネレー タ, ゲート付きタイマとして使用可能) ・ ベーシック・インターバル・タイマ/ウォッチドッグ・ タイマ：1チャンネル ・ 時計用タイマ：1チャンネル	
クロック出力 (PCL)		・ , 524, 262, 65.5 kHz (メイン・システム・クロック：4.19 MHz動作時) ・ , 750, 375, 93.8 kHz (メイン・システム・クロック：6.0 MHz動作時)		

項 目	μ PD753108	μ PD753208	μ PD75P3216
ブザー出力 (BUZ)	<ul style="list-style-type: none"> ・ 2, 4, 32 kHz (メイン・システム・クロック : 4.19 MHz動作時, サブシステム・クロック : 32.768 kHz動作時) ・ 2.93, 5.86, 46.9 kHz (メイン・システム・クロック : 6.0 MHz動作時) 	<ul style="list-style-type: none"> ・ 2, 4, 32 kHz (メイン・システム・クロック : 4.19 MHz動作時) ・ 2.93, 5.86, 46.9 kHz (メイン・システム・クロック : 6.0 MHz動作時) 	
シリアル・インタフェース	3種のモードに対応可能 <ul style="list-style-type: none"> ・ 3線式シリアルI/Oモード...MSB/LSB先頭切り替え可能 ・ 2線式シリアルI/Oモード ・ SBIモード 		
SCCレジスタ	あり	なし	
SOSレジスタ			
ベクタ割り込み	外部 : 3本, 内部 : 5本	外部 : 2本, 内部 : 5本	
テスト入力	外部 : 1本, 内部 : 1本		
電源電圧	$V_{DD} = 1.8 \sim 5.5 V$		
★ 動作周囲温度	$T_A = -40 \sim +85$		
パッケージ	<ul style="list-style-type: none"> ・ 64ピン・プラスチック QFP (14 mm) ・ 64ピン・プラスチック QFP (12 mm) 	48ピン・プラスチック・シュリンクSOP (375 mil, 0.65 mmピッチ)	

付録B 開発ツール

μPD753208を使用するシステム開発のために次のような開発ツールを用意しております。75XLシリーズでは、シリーズ共通のリロケータブル・アセンブラを品種ごとのデバイス・ファイルと組み合わせて使用します。

言語プロセッサ

RA75X リロケータブル・アセン ブラ	ホスト・マシン			オーダ名称(品名)
		OS	供給媒体	
★	PC-9800シリーズ	MS-DOS™ (Ver.3.30) 、 (Ver.6.2注)	3.5インチ2HD	μ S5A13RA75X
			5インチ2HD	μ S5A10RA75X
	IBM PC/AT™ およびその互換機	「IBM PC用のOSに ついて」参照	3.5インチ2HC	μ S7B13RA75X
			5インチ2HC	μ S7B10RA75X

★	デバイス・ファイル	PC-9800シリーズ	MS-DOS (Ver.3.30) 、 (Ver.6.2注)	3.5インチ2HD	μ S5A13DF753208
				5インチ2HD	μ S5A10DF753208
	IBM PC/AT およびその互換機	「IBM PC用のOSに ついて」参照	3.5インチ2HC	μ S7B13DF753208	
			5インチ2HC	μ S7B10DF753208	

注 Ver.5.00以降にはタスク・スワップ機能がありますが、このソフトウェアではタスク・スワップ機能は使用できません。

備考 アセンブラ、デバイス・ファイルの動作は、上記のホスト・マシンとOS上でのみ保証されます。

PROM書き込み用ツール

ハードウェア	PG-1500	付属のボードおよび別売りのプログラマ・アダプタを接続することにより、PROM内蔵のシングルチップ・マイクロコンピュータを、スタンド・アロンまたは、ホスト・マシンからの操作によりプログラミングできるPROMプログラマです。 また、256Kビットから4Mビットまでの代表的なPROMをプログラミングすることもできます。			
	PA-75P3216GT	μPD75P3216GT用の専用PROMプログラマ・アダプタで、PG-1500に接続して使用します。			
ソフトウェア	PG-1500 コントローラ	PG-1500とホスト・マシンをシリアルおよびパラレル・インタフェースで接続し、ホスト・マシン上でPG-1500を制御します。			
		ホスト・マシン	OS	供給媒体	オーダ名称(品名)
		PC-9800シリーズ	MS-DOS (Ver.3.30 、 Ver.6.2 ^注)	3.5インチ2HD	μ S5A13PG1500
				5インチ2HD	μ S5A10PG1500
		IBM PC/AT およびその互換機	「IBM PC用のOSについて」参照	3.5インチ2HD	μ S7B13PG1500
5インチ2HC	μ S7B10PG1500				

★

注 Ver.5.00以降にはタスク・スワップ機能がありますが、このソフトウェアではタスク・スワップ機能は使用できません。

備考 PG-1500コントローラの動作は、上記のホスト・マシンとOS上でのみ保証されます。

ディバグ用ツール

μPD753208のプログラム・ディバグ用ツールとしてインサーキット・エミュレータ（IE-75000-R、IE-75001-R）を用意しています。

それぞれのシステム構成を次に示します。

ハードウェア	IE-75000-R ^{注1}	IE-75000-Rは、75Xシリーズ、75XL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、ソフトウェアのディバグを行うためのインサーキット・エミュレータです。μPD753208サブシリーズを開発する場合、IE-75000-Rと別売りのエミュレーション・ボードIE-75300-R-EM、およびエミュレーション・プローブEP-753208GT-Rを組み合わせて使用します。 ホスト・マシン、PROMプログラマと接続して効率的にディバグを行うことができます。 なお、IE-75000-R内にはエミュレーション・ボードIE-75000-R-EMが含まれており、接続されています。			
	IE-75001-R	IE-75001-Rは、75Xシリーズ、75XL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、ソフトウェアのディバグを行うためのインサーキット・エミュレータです。μPD753208サブシリーズを開発する場合、IE-75001-Rと別売りのエミュレーション・ボードIE-75300-R-EM、およびエミュレーション・プローブEP-753208GT-Rを組み合わせて使用します。 ホスト・マシン、PROMプログラマを接続して効率的にディバグを行うことができます。			
	IE-75300-R-EM	μPD753208サブシリーズを使用する応用システムの評価を行うためのエミュレーション・ボードです。 IE-75000-RまたはIE-75001-Rと組み合わせて使用します。			
	EP-753208GT-R	μPD753208GT用のエミュレーション・プローブです。 IE-75000-RまたはIE-75001-Rと、IE-75300-R-EMに接続して使用します。			
	EV-9500GT-48	ターゲット・システムとの接続を容易にするフレキシブル基板EV-9500GT-48を添付しています。			
★ ソフトウェア	IEコントロール・プログラム	IE-75000-RまたはIE-75001-Rとホスト・マシンをRS-232-C、およびセントロニクスI/Fで接続し、ホスト・マシン上でIE-75000-RまたはIE-75001-Rを制御します。			
		ホスト・マシン	OS	供給媒体	オーダ名称（品名）
		PC-9800シリーズ	MS-DOS Ver.3.30 、 Ver.6.2 ^{注2}	3.5インチ2HD	μS5A13IE75X
				5インチ2HD	μS5A10IE75X
		IBM PC/AT およびその互換機	「IBM PC用のOSについて」参照	3.5インチ2HC	μS7B13IE75X
5インチ2HC	μS7B10IE75X				

注1．保守品です。

- Ver.5.00以降にはタスク・スワップ機能がありますが、このソフトウェアではタスク・スワップ機能は使用できません。

備考1．IEコントロール・プログラムの動作は、上記のホスト・マシンとOS上でのみ保証されます。

- μPD753204、753206、753208、75P3216を総称して、μPD753208サブシリーズといいます。

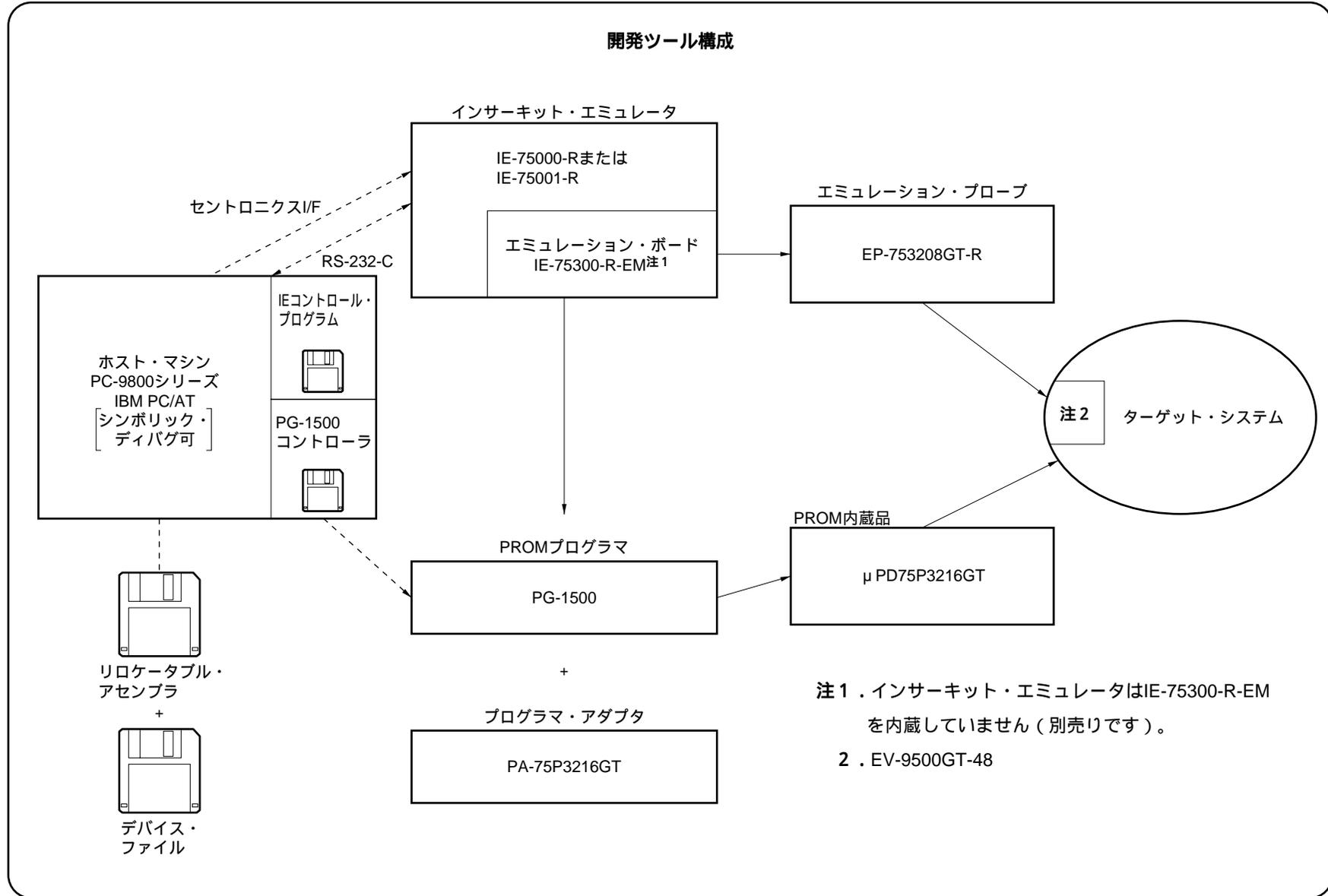
IBM PC用のOSについて

IBM PC用のOSとして、次のものがサポートされています。

OS	バージョン
★ PC DOS™	Ver.5.02 ~ Ver.6.3 J6.1/V注 ~ J6.3/V注
★ MS-DOS	Ver.5.0 ~ Ver.6.22 5.0/V注 ~ 6.2/V注
IBM DOS™	J5.02/V注

注 英語モードのみサポートしています。

注意 Ver.5.0以降にはタスク・スワップ機能がありますが、このソフトウェアでは、タスク・スワップ機能は使用できません。



〔メモ〕

付録C マスクROM発注の手順

プログラム開発が完了して、マスクROMを発注する場合には、次の手順で行ってください。

マスクROM発注の予約

特約店あるいは、当社販売部門に、マスクROM発注の予定を連絡してください（あらかじめご連絡をいただかないと処理が遅れる場合があります）。

★ 発注用媒体の作成

マスクROM発注用の媒体には、次の3つがあります。

- ・ UV-EPROM^注
- ・ 3.5インチのIBMフォーマットのフロッピー・ディスク（海外のみ）
- ・ 5インチのIBMフォーマットのフロッピー・ディスク（海外のみ）

注 UV-EPROMで発注する場合には、同じ内容のUV-EPROMを3個作成してください。
また、マスク・オプション・データはマスク・オプション情報書に記入してください。

必要書類の作成

マスクROMの発注にあたっては、下記の書類に記入してください。

- A．マスク式ROM発注書
- B．マスク式ROM発注チェック・シート
- C．マスク・オプション情報書（マスク・オプションがある製品の場合必要）

発 注

で作成した媒体と で記入した書類をまとめて、発注予約日までに特約店あるいは当社販売部門に提出してください。

注意 詳しくは、インフォメーション資料「ROMコードの発注方法」（C10302J）をご覧ください。

〔メモ〕

付録D 命令索引

D.1 命令索引（機能別）

【転送命令】

MOV A, #n4 ... 357, 380
MOV reg1, #n4 ... 357, 380
MOV XA, #n8 ... 357, 380
MOV HL, #n8 ... 357, 381
MOV rp2, #n8 ... 357, 381
MOV A, @HL ... 357, 381
MOV A, @HL + ... 357, 381
MOV A, @HL - ... 357, 381
MOV A, @rpa1 ... 357, 382
MOV XA, @HL ... 357, 382
MOV @HL, A ... 357, 382
MOV @HL, XA ... 357, 383
MOV A, mem ... 357, 383
MOV XA, mem ... 357, 383
MOV mem, A ... 357, 383
MOV mem, XA ... 357, 384
MOV A, reg ... 357, 384
MOV XA, rp' ... 357, 384
MOV reg1, A ... 357, 384
MOV rp' 1, XA ... 357, 384
XCH A, @HL ... 357, 385
XCH A, @HL + ... 357, 385
XCH A, @HL - ... 357, 385
XCH A, @rpa1 ... 357, 385
XCH XA, @HL ... 357, 386
XCH A, mem ... 357, 386
XCH XA, mem ... 357, 386
XCH A, reg1 ... 357, 387
XCH XA, rp' ... 357, 387

【テーブル参照命令】

MOVT XA, @PCDE ... 358, 388
MOVT XA, @PCXA ... 358, 390
MOVT XA, @BCDE ... 358, 391
MOVT XA, @BCXA ... 358, 391

【ビット転送命令】

MOV1 CY, fmem. bit ... 358, 392
MOV1 CY, pmem. @L ... 358, 392
MOV1 CY, @H + mem. bit ... 358, 392
MOV1 fmem. bit, CY ... 358, 392
MOV1 pmem. @L, CY ... 358, 392
MOV1 @H + mem. bit, CY ... 358, 392

【演算命令】

ADDS A, #n4 ... 358, 393
ADDS XA, #n8 ... 358, 393
ADDS A, @HL ... 358, 393
ADDS XA, rp' ... 358, 393
ADDS rp' 1, XA ... 358, 394
ADDC A, @HL ... 358, 394
ADDC XA, rp' ... 358, 394
ADDC rp' 1, XA ... 358, 395
SUBS A, @HL ... 359, 395
SUBS XA, rp' ... 359, 395
SUBS rp' 1, XA ... 359, 396
SUBC A, @HL ... 359, 396
SUBC XA, rp' ... 359, 396
SUBC rp' 1, XA ... 359, 397
AND A, #n4 ... 359, 397
AND A, @HL ... 359, 397

AND XA, rp' ... 359, 397
 AND rp' 1, XA ... 359, 398
 OR A, #n4 ... 359, 398
 OR A, @HL ... 359, 398
 OR XA, rp' ... 359, 398
 OR rp' 1, XA ... 359, 399
 XOR A, #n4 ... 359, 399
 XOR A, @HL ... 359, 399
 XOR XA, rp' ... 359, 399
 XOR rp' 1, XA ... 359, 400

[アキュムレータ操作命令]

RORC A ... 359, 401
 NOT A ... 359, 401

[増減命令]

INCS reg ... 359, 402
 INCS rp1 ... 359, 402
 INCS @HL ... 359, 402
 INCS mem ... 359, 402
 DECS reg ... 359, 403
 DECS rp' ... 359, 403

[比較命令]

SKE reg, #n4 ... 359, 404
 SKE @HL, #n4 ... 359, 404
 SKE A, @HL ... 359, 404
 SKE XA, @HL ... 359, 404
 SKE A, reg ... 359, 405
 SKE XA, rp' ... 359, 405

[キャリー・フラグ操作命令]

SET1 CY ... 360, 406
 CLR1 CY ... 360, 406
 SKT CY ... 360, 406
 NOT1 CY ... 360, 406

[メモリ・ビット操作命令]

SET1 mem. bit ... 360, 407
 SET1 fmem. bit ... 360, 407
 SET1 pmem. @L ... 360, 407
 SET1 @H+mem. bit ... 360, 407
 CLR1 mem. bit ... 360, 407
 CLR1 fmem. bit ... 360, 407
 CLR1 pmem. @L ... 360, 407
 CLR1 @H+mem. bit ... 360, 407
 SKT mem. bit ... 360, 408
 SKT fmem. bit ... 360, 408
 SKT pmem. @L ... 360, 408
 SKT @H+mem. bit ... 360, 408
 SKF mem. bit ... 360, 408
 SKF fmem. bit ... 360, 408
 SKF pmem. @L ... 360, 408
 SKF @H+mem. bit ... 360, 408
 SKTCLR fmem. bit ... 360, 409
 SKTCLR pmem. @L ... 360, 409
 SKTCLR @H+mem. bit ... 360, 409
 AND1 CY, fmem. bit ... 360, 409
 AND1 CY, pmem. @L ... 360, 409
 AND1 CY, @H+mem. bit ... 360, 409
 OR1 CY, fmem. bit ... 360, 409
 OR1 CY, pmem. @L ... 360, 409
 OR1 CY, @H+mem. bit ... 360, 409
 XOR1 CY, fmem. bit ... 360, 410
 XOR1 CY, pmem. @L ... 360, 410
 XOR1 CY, @H+mem. bit ... 360, 410

[分岐命令]

BR addr ... 361, 411
 BR addr1 ... 362, 411
 BRA !addr1 ... 364, 411
 BR !addr ... 362, 411
 BR \$addr ... 362, 412
 BR \$addr1 ... 363, 412

BRCB ! caddr ... 364, 413
BR PCDE ... 363, 414
BR PCXA ... 363, 414
BR BCDE ... 363, 415
BR BCXA ... 363, 415
TBR addr ... 371, 415

[特殊命令]

SEL RBn ... 370, 425
SEL MBn ... 370, 425
GETI taddr ... 371, 425

[サブルーチン・スタック制御命令]

CALLA ! addr1 ... 364, 416
CALL ! addr ... 365, 416
CALLF ! faddr ... 366, 417
TCALL ! addr ... 366, 417
RET ... 367, 418
RETS ... 368, 418
RETI ... 369, 419
PUSH rp ... 369, 420
PUSH BS ... 369, 420
POP rp ... 369, 420
POP BS ... 369, 420

[割り込み制御命令]

EI ... 369, 421
EI IE x x x ... 369, 421
DI ... 369, 421
DI IE x x x ... 369, 421

[入出力命令]

IN A, PORTn ... 370, 422
IN XA, PORTn ... 370, 422
OUT PORTn, A ... 370, 423
OUT PORTn, XA ... 370, 423

[CPU制御命令]

HALT ... 370, 424
STOP ... 370, 424
NOP ... 370, 424

D.2 命令索引 (アルファベット順)

[A]

ADDC A, @HL ... 358, 394
 ADDC rp' 1, XA ... 358, 395
 ADDC XA, rp' ... 358, 394
 ADDS A, #n4 ... 358, 393
 ADDS A, @HL ... 358, 393
 ADDS rp' 1, XA ... 358, 394
 ADDS XA, rp' ... 358, 393
 ADDS XA, #n8 ... 358, 393
 AND A, #n4 ... 359, 397
 AND A, @HL ... 359, 397
 AND rp' 1, XA ... 359, 398
 AND XA, rp' ... 359, 397
 AND1 CY, fmem. bit ... 360, 409
 AND1 CY, pmem. @L ... 360, 409
 AND1 CY, @H+mem. bit ... 360, 409

[B]

BR addr ... 361, 411
 BR addr1 ... 362, 411
 BR BCDE ... 363, 415
 BR BCXA ... 363, 415
 BR PCDE ... 363, 414
 BR PCXA ... 363, 414
 BR !addr ... 362, 411
 BR \$addr ... 362, 412
 BR \$addr1 ... 363, 412
 BRA !addr1 ... 364, 411
 BR CB !caddr ... 364, 413

[C]

CALL !addr ... 365, 416
 CALLA !addr1 ... 364, 416
 CALLF !faddr ... 366, 417
 CLR1 CY ... 360, 406

CLR1 fmem. bit ... 360, 407
 CLR1 mem. bit ... 360, 407
 CLR1 pmem. @L ... 360, 407
 CLR1 @H+mem. bit ... 360, 407

[D]

DECS reg ... 359, 403
 DECS rp' ... 359, 403
 DI ... 369, 421
 DI IE x x x ... 369, 421

[E]

EI ... 369, 421
 EI IE x x x ... 369, 421

[G]

GETI taddr ... 371, 425

[H]

HALT ... 370, 424

[I]

IN A, PORTn ... 370, 422
 IN XA, PORTn ... 370, 422
 INCS mem ... 359, 402
 INCS reg ... 359, 402
 INCS rp1 ... 359, 402
 INCS @HL ... 359, 402

[M]

MOV A, mem ... 357, 383
 MOV A, reg ... 357, 384
 MOV A, #n4 ... 357, 380
 MOV A, @HL ... 357, 381
 MOV A, @HL+ ... 357, 381

MOV	A, @HL -	...	357, 381
MOV	A, @rpa1	...	357, 382
MOV	HL, #n8	...	357, 381
MOV	mem, A	...	357, 383
MOV	mem, XA	...	357, 384
MOV	reg1, A	...	357, 384
MOV	reg1, #n4	...	357, 380
MOV	rp'1, XA	...	357, 384
MOV	rp2, #n8	...	357, 381
MOV	XA, mem	...	357, 383
MOV	XA, rp'	...	357, 384
MOV	XA, #n8	...	357, 380
MOV	XA, @HL	...	357, 382
MOV	@HL, A	...	357, 382
MOV	@HL, XA	...	357, 383
MOVT	XA, @BCDE	...	358, 391
MOVT	XA, @BCXA	...	358, 391
MOVT	XA, @PCDE	...	358, 388
MOVT	XA, @PCXA	...	358, 390
MOV1	CY, fmem. bit	...	358, 392
MOV1	CY, pmem. @L	...	358, 392
MOV1	CY, @H + mem. bit	...	358, 392
MOV1	fmem. bit, CY	...	358, 392
MOV1	pmem. @L, CY	...	358, 392
MOV1	@H + mem. bit, CY	...	358, 392
[N]			
NOP	...	370, 424	
NOT	A	...	359, 401
NOT1	CY	...	360, 406
[O]			
OR	A, #n4	...	359, 398
OR	A, @HL	...	359, 398
OR	rp'1, XA	...	359, 399
OR	XA, rp'	...	359, 398
OR1	CY, fmem. bit	...	360, 409
OR1	CY, pmem. @L	...	360, 409
OR1	CY, @H + mem. bit	...	360, 409
OUT	PORTn, A	...	370, 423
OUT	PORTn, XA	...	370, 423
[P]			
POP	BS	...	369, 420
POP	rp	...	369, 420
PUSH	BS	...	369, 420
PUSH	rp	...	369, 420
[R]			
RET	...	367, 418	
RETI	...	369, 419	
RETS	...	368, 418	
RORC	A	...	359, 401
[S]			
SEL	MBn	...	370, 425
SEL	RBn	...	370, 425
SET1	CY	...	360, 406
SET1	fmem. bit	...	360, 407
SET1	mem. bit	...	360, 407
SET1	pmem. @L	...	360, 407
SET1	@H + mem. bit	...	360, 407
SKE	A, reg	...	359, 405
SKE	A, @HL	...	359, 404
SKE	reg, #n4	...	359, 404
SKE	XA, rp'	...	359, 405
SKE	XA, @HL	...	359, 404
SKE	@HL, #n4	...	359, 404
SKF	fmem. bit	...	360, 408
SKF	mem. bit	...	360, 408
SKF	pmem. @L	...	360, 408
SKF	@H + mem. bit	...	360, 408
SKT	CY	...	360, 406
SKT	fmem. bit	...	360, 408

SKT mem. bit ... 360, 408
SKT pmem. @L ... 360, 408
SKT @H + mem. bit ... 360, 408
SKTCLR fmem. bit ... 360, 409
SKTCLR pmem. @L ... 360, 409
SKTCLR @H + mem. bit ... 360, 409
STOP ... 370, 424
SUBC A, @HL ... 359, 396
SUBC rp'1, XA ... 359, 397
SUBC XA, rp' ... 359, 396
SUBS A, @HL ... 359, 395
SUBS rp'1, XA ... 359, 396
SUBS XA, rp' ... 359, 395

[T]

TBR addr ... 371, 415
TCALL !addr ... 366, 417

[X]

XCH A, mem ... 357, 386
XCH A, reg1 ... 357, 387
XCH A, @HL ... 357, 385
XCH A, @HL + ... 357, 385
XCH A, @HL - ... 357, 385
XCH A, @rpa1 ... 357, 385
XCH XA, mem ... 357, 386
XCH XA, rp' ... 357, 387
XCH XA, @HL ... 357, 386
XOR A, #n4 ... 359, 399
XOR A, @HL ... 359, 399
XOR rp'1, XA ... 359, 400
XOR XA, rp' ... 359, 399
XOR1 CY, fmem. bit ... 360, 410
XOR1 CY, pmem. @L ... 360, 410
XOR1 CY, @H + mem. bit ... 360, 410

付録E ハードウェア索引

E.1 ハードウェア索引 (50音順)

[あ行]

アクノリッジ許可ビット (ACKE) ...	206
アクノリッジ検出フラグ (ACKD) ...	206
アクノリッジ・トリガ・ビット (ACKT) ...	206
アドレス・コンパレータからの信号 (COI) ...	204
INTAレジスタ (INTA) ...	72
INTCレジスタ (INTC) ...	72
INTEレジスタ (INTE) ...	72
INTFレジスタ (INTF) ...	72
INTGレジスタ (INTG) ...	72
INTHレジスタ (INTH) ...	72
INT0エッジ検出モード・レジスタ (IM0) ...	305
INT2エッジ検出モード・レジスタ (IM2) ...	325
INT0割り込み許可フラグ (IE0) ...	299
INT2割り込み許可フラグ (IE2) ...	322
INT4割り込み許可フラグ (IE4) ...	299
INT0割り込み要求フラグ (IRQ0) ...	299
INT2割り込み要求フラグ (IRQ2) ...	322
INT4割り込み要求フラグ (IRQ4) ...	299
ウエイク・アップ機能指定ビット (WUP) ...	204
ウォッチドッグ・タイマ許可フラグ (WDTM) ...	131

[か行]

キー割り込み0-3 (KR0-KR3) ...	323
キャリー・フラグ (CY) ...	93
クロック出力モード・レジスタ (CLOM) ...	126
ゲート制御許可フラグ (TGCE) ...	149
コマンド検出フラグ (CMDD) ...	207
コマンド・トリガ・ビット (CMDT) ...	207

【さ行】

- シフト・レジスタ (SIO) ... 208
- シリアル・インタフェース動作許可/禁止指定ビット (CSIE) ... 203
- シリアル・インタフェース割り込み許可フラグ (IECSI) ... 299
- シリアル・インタフェース割り込み要求フラグ (IRQCSI) ... 299
- シリアル動作モード・レジスタ (CSIM) ... 202
- シリアル・バス・インタフェース・コントロール・レジスタ (SBIC) ... 205
- スキップ・フラグ0-2 (SK0-SK2) ... 94
- スタック・バンク選択レジスタ (SBS) ... 89
- スタック・ポインタ (SP) ... 89
- スレーブ・アドレス・レジスタ (SVA) ... 209

【た行】

- タイマ/イベント・カウンタ0カウント・レジスタ (T0) ... 71
- タイマ・カウンタ1カウント・レジスタ (T1) ... 71
- タイマ・カウンタ2カウント・レジスタ (T2) ... 71
- タイマ・カウンタ・コントロール・レジスタ (TC2) ... 149
- タイマ/イベント・カウンタ0出力許可フラグ (TOE0) ... 148
- タイマ・カウンタ1出力許可フラグ (TOE1) ... 148
- タイマ・カウンタ2出力許可フラグ (TOE2) ... 149
- タイマ・カウンタ・ハイ・レベル期間設定用モジュロ・レジスタ (TMOD2H) ... 165
- タイマ/イベント・カウンタ0モード・レジスタ (TM0) ... 144
- タイマ・カウンタ1モード・レジスタ (TM1) ... 144
- タイマ・カウンタ2モード・レジスタ (TM2) ... 144
- タイマ/イベント・カウンタ0モジュロ・レジスタ (TMOD0) ... 156
- タイマ・カウンタ1モジュロ・レジスタ (TMOD1) ... 156
- タイマ・カウンタ2モジュロ・レジスタ (TMOD2) ... 156
- タイマ/イベント・カウンタ0割り込み許可フラグ (IET0) ... 299
- タイマ・カウンタ1割り込み許可フラグ (IET1) ... 299
- タイマ・カウンタ2割り込み許可フラグ (IET2) ... 299
- タイマ/イベント・カウンタ0割り込み要求フラグ (IRQT0) ... 299
- タイマ・カウンタ1割り込み要求フラグ (IRQT1) ... 299
- タイマ・カウンタ2割り込み要求フラグ (IRQT2) ... 299
- 時計モード・レジスタ (WM) ... 139

【な行】

- ノー・リターン・ゼロ・バッファ・フラグ (NRZB) ... 149
- ノー・リターン・ゼロ・フラグ (NRZ) ... 149

【は行】

- バス・リリース検出フラグ (RELD) ... 207
- バス・リリース・トリガ・ビット (RELT) ... 207
- バンク選択レジスタ (BS) ... 97
- BT割り込み許可フラグ (IEBT) ... 299
- BT割り込み要求フラグ (IRQBT) ... 299
- ビジィ許可ビット (BSYE) ... 206
- ビット・シーケンシャル・バッファ0-3 (BSB0-BSB3) ... 292
- 表示制御レジスタ (LCDC) ... 267
- 表示モード・レジスタ (LCDM) ... 265
- プルアップ抵抗指定レジスタ・グループA, B (POGA, POGB) ... 113
- プログラム・カウンタ (PC) ... 77
- プログラム・ステータス・ワード (PSW) ... 93
- プロセッサ・クロック・コントロール・レジスタ (PCC) ... 118
- ベーシック・インターバル・タイマ (BT) ... 132
- ベーシック・インターバル・タイマ・モード・レジスタ (BTM) ... 129
- ポート (PORT0-3, 5, 6, 8, 9) ... 99
- ポート・モード・レジスタ・グループA, B, C (PMGA, PMGB, PMGC) ... 106

【ま行】

- メモリ・バンク許可フラグ (MBE) ... 96
- メモリ・バンク選択レジスタ (MBS) ... 97

【ら行】

- レジスタ・バンク許可フラグ (RBE) ... 96
- レジスタ・バンク選択レジスタ (RBS) ... 97
- リモコン出力許可フラグ (REMC) ... 149

【わ行】

- 割り込みステータス・フラグ0, 1 (IST0, IST1) ... 306
- 割り込みプライオリティ選択レジスタ (IPS) ... 300
- 割り込みマスタ許可フラグ (IME) ... 301

E.2 ハードウェア索引 (アルファベット順)

[A]

ACKD ... 206
ACKE ... 206
ACKT ... 206

[B]

BS ... 97
BSB0-BSB3 ... 292
BSYE ... 206
BT ... 132
BTM ... 129

[C]

CLOM ... 126
CMDD ... 207
CMDT ... 207
COI ... 204
CSIE ... 203
CSIM ... 202
CY ... 93

[I]

IE0 ... 299
IE2 ... 322
IE4 ... 299
IEBT ... 299
IECSI ... 299
IET0 ... 299
IET1 ... 299
IET2 ... 299
IEW ... 322
IM0 ... 305
IM2 ... 325
IME ... 301
INTA ... 72

INTC ... 72
INTE ... 72
INTF ... 72
INTG ... 72
INTH ... 72
IPS ... 300
IRQ0 ... 299
IRQ2 ... 322
IRQ4 ... 299
IRQBT ... 299
IRQCSI ... 299
IRQT0 ... 299
IRQT1 ... 299
IRQT2 ... 299
IRQW ... 322
IST0, IST1 ... 306

[K]

KR0-KR3 ... 323

[L]

LCDC ... 267
LCDM ... 265

[M]

MBE ... 96
MBS ... 97

[N]

NRZ ... 149
NRZB ... 149

[P]

PC ... 77
PCC ... 118

PMGA, PMGB, PMBC ... 106
POGA, POGB ... 113
PORT0-PORT9 ... 99
PSW ... 93

[R]

RBE ... 96
RBS ... 97
RELD ... 207
RELT ... 207
REMC ... 149

[S]

SBIC ... 205
SBS ... 89
SIO ... 208
SK0-SK2 ... 94
SP ... 89
SVA ... 209

[T]

T0 ... 71
TGCE ... 149
TM0, TM1, TM2 ... 144
TMOD0, TMOD1, TMOD2 ... 156
TMOD2H ... 165
TOE0 ... 148
TOE1 ... 148
TOE2 ... 149

[W]

WDTM ... 131
WM ... 139
WUP ... 204

〔メモ〕

付録 F 改版履歴

これまでの改版履歴を次に示します。なお、適用箇所は各版での章を示します。

版 数	前版からの主な改版内容	適用箇所
第 2 版	μ PD753204, 753206, 753208, 75P3216を開発中から開発済みに変更	全 般
	ポート 4 , 5 のN-chオープン・ドレイン時, 入力耐圧を12 Vから13 Vに変更	
	データ・バス端子 (D0-D7) を追加	
	未使用端子の処理一覧を変更	第 2 章 端子機能
	Mk モードとMk モードの違いの注意を変更	第 4 章 内部CPU機能
	パス・リリース信号 (REL) とコマンド信号 (CMD) に注意を追加	第 5 章 周辺ハードウェア機能
	マスク・オプションの選択を追加	第 7 章 スタンバイ機能
	プログラム・メモリの書き込みの手順を変更	第 9 章 PROM (プログラム・メモリ) の書き込みとベリファイ
	プログラム・メモリの読み出しの手順を変更	
	各命令の命令コードの命令一覧表の一部を修正	第11章 命令セット
	マスクROM発注の手順の発注用媒体を変更	付録 C マスクROM発注の手順

— 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] μPD753208 ユーザーズ・マニュアル

(U10158JJ2V1UM00 (第2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してください。

NEC半導体テクニカルホットライン

FAX: (044) 548-7900