

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（暫定

V30MZTM

16 ビット・マイクロプロセッサ・コア

ハードウェア編

資料番号 A13761JJ1V2UM00 (第1版)

発行年月 January 2002 NS CP(N)

© NEC Corporation 1998

[メモ]

目次要約

第 1 章	概 説	... 15
第 2 章	端子機能	... 19
第 3 章	CPU 機能	... 27
第 4 章	バス制御機能	... 49
第 5 章	割り込み機能	... 57
第 6 章	スタンバイ機能	... 67
第 7 章	リセット機能	... 69
第 8 章	テスト機能	... 71
付録 A	命令実行クロック数一覧	... 73
付録 B	総合索引	... 85

CMOSデバイスの一般的注意事項

静電気対策（MOS全般）

注意 MOSデバイス取り扱いの際は静電気防止を心がけてください。

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

また、MOSデバイスを実装したボードについても同様の扱いをしてください。

未使用入力の処理（CMOS特有）

注意 CMOSデバイスの入力レベルは固定してください。

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

V30MZ, V30HL, V30MX, V シリーズは日本電気株式会社の商標です。

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（仕様を決定した者）が判定してください。

- 本資料は、この製品の企画段階で作成していますので、予告なしに内容を変更することがあります。また本資料で扱う製品の製品化を中止することがあります。
- 文書による当社の承諾なしに本資料の転載複製を禁じます。
- 本資料に記載された製品の使用もしくは本資料に記載の情報の使用に際して、当社は当社もしくは第三者の知的財産権その他の権利に対する保証または実施権の許諾を行うものではありません。上記使用に起因する第三者所有の権利にかかわる問題が発生した場合、当社はその責を負うものではありませんのでご了承ください。
- 本資料に記載された回路、ソフトウェア、及びこれらに付随する情報は、半導体製品の動作例、応用例を説明するためのものです。従って、これら回路・ソフトウェア・情報をお客様の機器に使用される場合には、お客様の責任において機器設計をしてください。これらの使用に起因するお客様もしくは第三者の損害に対して、当社は一切その責を負いません。
- 当社は品質、信頼性の向上に努めていますが、半導体製品はある確率で故障が発生します。当社半導体製品の故障により結果として、人身事故、火災事故、社会的な損害等を生じさせない冗長設計、延焼対策設計、誤動作防止設計等安全設計に十分ご注意願います。
- 当社は、当社製品の品質水準を「標準水準」、「特別水準」およびお客様に品質保証プログラムを指定して頂く「特定水準」に分類しております。また、各品質水準は以下に示す用途に製品が使われることを意図しておりますので、当社製品の品質水準をご確認の上ご使用願います。

標準水準：コンピュータ、OA機器、通信機器、計測機器、AV機器、家電、工作機械、パーソナル機器、産業用ロボット

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

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

当社製品のデータ・シート／データ・ブック等の資料で、特に品質水準の表示がない場合は標準水準製品であることを表します。当社製品を上記の「標準水準」の用途以外でご使用をお考えのお客様は、必ず事前に当社販売窓口までご相談頂きますようお願い致します。

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

はじめに

- 対象者** このマニュアルは CBIC の CPU コアである V30MZ のハードウェア機能を理解し、それを用いた応用システムを設計するユーザを対象とします。
- 目的** このマニュアルは、次の構成に示す V30MZ 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います。

- 概説
- 端子機能
- CPU 機能
- バス制御機能
- 割り込み機能
- スタンバイ機能
- リセット機能
- テスト機能

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

一通り V30MZ の機能を理解しようとするとき
→目次に従ってお読みください。

V30HL™, V30MX™ との違いを知りたいとき
→1.3 V30MZ と V30HL, V30MX の違いを参照してください。

命令機能の詳細を知りたいとき
→別冊の **16 ビット V シリーズ™ ユーザーズ・マニュアル 命令編**を参照してください。

- 凡例**
- | | |
|-------------|---|
| データ表記の重み | : 左が上位桁, 右が下位桁 |
| アクティブ・ロウの表記 | : xxxB (端子, 信号名称のあとに B) |
| 注 | : 本文中につけた注の説明 |
| 注意 | : 気をつけて読んでいただきたい内容 |
| 備考 | : 本文の補足説明 |
| 数の表記 | : 2 進数 ...xxxx または xxxxB
10 進数...xxxx
16 進数...xxxxH |

2 のべき数を示す接頭語 (アドレス空間, メモリ容量) :

K (キロ)	: $2^{10} = 1024$
M (メガ)	: $2^{20} = 1024^2$
G (ギガ)	: $2^{30} = 1024^3$

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

- 16 ビット V シリーズ ユーザーズ・マニュアル 命令編 (U11301J)
- CB-C9 ファミリ VX/VM タイプ ユーザーズ・マニュアル 設計編 (A12745J)
- CB-C9 ファミリ VX/VM タイプ コア・ライブラリ
ユーザーズ・マニュアル CPU コア編 (A13195J)

目 次

第 1 章 概 説 ... 15

- 1.1 特 徴 ... 15
- 1.2 シンボル図 ... 15
- 1.3 V30MZ と V30HL, V30MX の違い ... 16

第 2 章 端子機能 ... 19

- 2.1 端子一覧 ... 19
- 2.2 端子状態 ... 20
- 2.3 端子機能の説明 ... 21
 - 2.3.1 通常端子 ... 21
 - 2.3.2 テスト用端子 ... 24
 - 2.3.3 予約端子 ... 24
- 2.4 端子の未使用時の処理 ... 25

第 3 章 CPU 機能 ... 27

- 3.1 レジスタ構成 ... 27
 - 3.1.1 汎用レジスタ (AW, BW, CW, DW) ... 27
 - 3.1.2 セグメント・レジスタ (PS, SS, DS0, DS1) ... 27
 - 3.1.3 ポインタ (SP, BP) ... 27
 - 3.1.4 プログラム・カウンタ (PC) ... 28
 - 3.1.5 プログラム・ステータス・ワード (PSW) ... 28
 - 3.1.6 インデクス・レジスタ (IX, IY) ... 32
- 3.2 アドレス空間 ... 33
 - 3.2.1 メモリ空間 ... 33
 - 3.2.2 I/O 空間 ... 34
- 3.3 命令プリフェッチ ... 35
- 3.4 論理アドレスと物理アドレス ... 36
 - 3.4.1 セグメント方式 ... 36
 - 3.4.2 セグメントの構成 ... 37
 - 3.4.3 ダイナミック・リロケーション ... 39
- 3.5 実効アドレス ... 41
- 3.6 命令セット ... 42
 - 3.6.1 機能別命令セット一覧 ... 42

3.6.2	命令語の形式 ...	43
3.7	アドレッシング・モード ...	44
3.7.1	命令アドレス ...	44
3.7.2	データ・アドレス ...	45
第4章	バス制御機能 ...	49
4.1	メモリとのインタフェース ...	49
4.1.1	ワード・データをアクセスする場合の注意事項 ...	50
4.2	I/O とのインタフェース ...	51
4.3	メモリ, I/O のリード/ライト・タイミング ...	52
4.3.1	メモリ, I/O のリード・タイミング ...	52
4.3.2	メモリ, I/O のライト・タイミング ...	54
4.4	バス・ホールド機能 ...	56
第5章	割り込み機能 ...	57
5.1	ハードウェア割り込み ...	60
5.1.1	ノンマスカブル割り込み (NMI) ...	60
5.1.2	マスカブル割り込み (INT) ...	60
5.2	ソフトウェア割り込み ...	62
5.3	割り込みが受け付けられないタイミング ...	63
5.4	ブロック処理命令実行中の割り込み処理 ...	64
第6章	スタンバイ機能 ...	67
6.1	スタンバイ・モードの設定 ...	67
6.2	スタンバイ・モード ...	67
6.3	スタンバイ・モードの解除 ...	68
6.3.1	ハードウェア割り込み要求による解除 ...	68
6.3.2	RESET 入力による解除 ...	68
第7章	リセット機能 ...	69
第8章	テスト機能 ...	71
8.1	テスト端子 ...	71
8.1.1	テスト・バス端子 (TBI22-TBI0, TBO42-TBO0) ...	71
8.1.2	BUNRI, TEST 端子 ...	71
8.2	ノーマル・モード ...	72

8.3 単体テスト・モードとスタンバイ・テスト・モード ... 72

8.3.1 単体テスト・モード ... 72

8.3.2 スタンバイ・テスト・モード ... 72

付録 A 命令実行クロック数一覧 ... 73

付録 B 総合索引 ... 85

B.1 50音で始まる語句の索引 ... 85

B.2 アルファベットで始まる語句の索引 ... 87

図の目次

図番号	タイトル, ページ
3 - 1	プログラム・ステータス・ワード (PSW) ... 29
3 - 2	メモリ・マップ ... 33
3 - 3	ワード・データとダブル・ワード・データの構成 ... 34
3 - 4	I/O マップ ... 34
3 - 5	セグメント方式の概念図 ... 36
3 - 6	セグメント・レジスタ, オフセット・アドレスと物理アドレスの関係 ... 37
3 - 7	各セグメント・レジスタとセグメント, およびメモリ空間の関係 ... 38
3 - 8	ダイナミック・リロケーション ... 40
3 - 9	メモリ・アドレスの計算 ... 41
3 - 10	命令語形式 ... 43
4 - 1	V30MZ とメモリのインタフェース ... 49
4 - 2	メモリ, I/O のリード・タイミング ... 52
4 - 3	メモリ, I/O のライト・タイミング ... 54
4 - 4	バス・ホールド・タイミング ... 56
5 - 1	割り込みベクタ・テーブルの構成 ... 58
5 - 2	割り込みアクリッジ・サイクル ... 61
6 - 1	スタンバイ・モードへの移行のタイミング ... 67

表の目次

表番号	タイトル, ページ
2 - 1	オペランドと UBEB, A0, および各バス・サイクルの関係 ... 21
2 - 2	BS3-BS0 信号とバス・サイクルの関係 ... 22
3 - 1	各メモリ要素のアドレス, データ構成 ... 33
3 - 2	セグメント・レジスタとオフセット・アドレス指定の対応 ... 38
3 - 3	機能別命令セット一覧 ... 42
4 - 1	V30MZ のデータ・アクセス ... 50
5 - 1	割り込み要因一覧 ... 57
5 - 2	割り込み要求を受け付けるまでに要するバス・サイクル数 ... 64
7 - 1	リセット時の出力端子状態 ... 69
7 - 2	リセット後の各レジスタの初期値 ... 69
8 - 1	テスト・モードの設定一覧 ... 71
A - 1	命令実行クロック数一覧 ... 74

[メモ]

第1章 概 説

V30MZ は、NEC のオリジナル・マイクロプロセッサ「 μ PD70116H」（別名称：V30HL）のバス効率を改善した CPU コア「V30MX」を、さらに改良した CPU コアです。

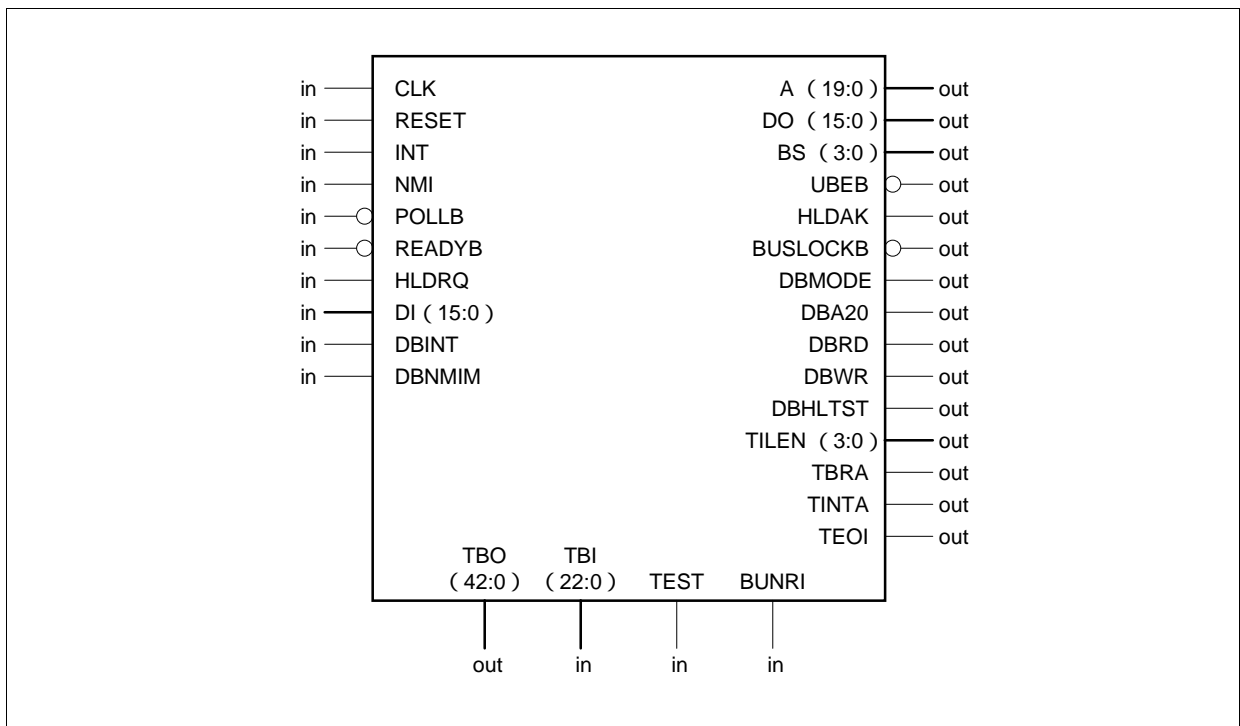
1 クロック / 1 バス・サイクルを実現してバス効率を向上、また内部パイプライン化により命令実行速度を大幅に向上させ、RISC 系マイクロプロセッサ並みの高速処理が可能となっています。

V30MX の 4.3MIPS（33MHz 動作，ノー・ウエイト時）に対し、V30MZ では 35MIPS（66MHz 動作，ノー・ウエイト時）の処理性能を実現しました。

1.1 特 徴

- (1) 処理性能：35MIPS（66MHz 動作，ノー・ウエイト時）
- (2) CMOS スタティック設計（内部システム・クロックの完全停止が可能）
- (3) 1 バス・サイクル：1 クロック
- (4) 外部バス・インタフェース
 - アドレス・バス：20 ビット
 - データ・バス：16 ビット（入力 / 出力分離バス）
- (5) バス・ホールド機能
- (6) スタンバイ機能（HALT モード）

1.2 シンボル図



1.3 V30MZ と V30HL, V30MX の違い

(1/3)

項 目	V30MZ	V30HL	V30MX
アドレス/データ・バス	A19-A0, DI15-DI0, DO15-DO0	A19-A16, AD15-AD0	A23-A0, D15-D0
大規模モード/小規模モード	なし	あり	なし
端子機能	V30HL より次の端子を削除 ^準 ASTB, PS3-PS0, BUFEN, QS1, QS0, BUF \bar{R} W, \bar{R} D, IC, \bar{R} Q/AK1, \bar{R} Q/AK0, INTAK, S/LG, LBS0, \bar{W} R, NC		-
μ PD8080AF エミュレーション機能	なし	あり	なし
数値演算コプロセッサとの接続	不可	可	
LIM EMS4.0 機能	なし		あり
CBIC コアとしてのテスト機能	あり (TBI22-TBI0, TBO42-TBO0, BUNRI, TEST)	なし	あり
HALT 命令の前に BUSLOCK 命令を実行した場合の BUSLOCKB 端子の状態	ハイ・レベル出力	ロウ・レベル出力	
割り込みアクリッジ・サイクルの UBEB 端子の状態	ハイ・レベル出力	ロウ・レベル出力	
バス・ホールド時の出力端子の状態	2.2 端子状態参照	ハイ・インピーダンス	
BUSLOCK 命令とバス・ホールド要求の関係	メモリまたは I/O へのアクセスを行わない命令の直前に BUSLOCK 命令を実行しても、BUSLOCK 命令は無視され、バス・ホールド要求を受け付けます。BUSLOCKB 出力もハイ・レベルのままです。	すべての命令で BUSLOCK 命令が有効です。	
奇数アドレスのワード・データにアクセスする場合の、1 回目と 2 回目のバス・サイクルの間でのバス・ホールド要求受け付け	不可	可	
バス・ホールド状態からスタンバイ・モードへの復帰時のバス・ステータス出力	なし (アイドル状態のままです)	あり	
命令実行時間	V30MZ は、各命令の実行クロック数や CPU 動作周波数が改善されているので、命令実行時間が大幅に短縮されています。連続した I/O アクセスなど、命令の実行クロック数に依存するプログラムは正常に動作しない場合があるので注意してください。		
割り込み応答時間	V30MZ は、内部でパイプライン処理を行っており、複数個の命令を並列に実行しています。そのため、あるバス・サイクルに同期してハードウェア割り込みを要求するような場合、V30HL, V30MX に比較して、より多くの命令を実行してから割り込み要求に応答する場合があります。ただし、I/O アクセスについては、このようなことはありません。		
未定義フラグの変化	フラグの変化が不定と定義されている演算命令を実行した場合、実行後のフラグの内容が V30HL, V30MX と異なる場合があります。特に乗除算命令で発生します。		
割り込み要求受け付け禁止タイミング	割り込み要求が受け付けられないタイミングが異なります (5.3 割り込みが受け付けられないタイミング を参照してください) 。		

注 V30MZ の BS3 端子は、出力タイミングを除き V30HL の \bar{I} O/M 端子と同一の機能を持ちます。

備考 アクティブ・ロウ端子の名称を V30HL では xxx (上線を付記) と表記していますが、V30MZ では xxxB ("B" を付記) と表記しています。

項 目	V30MZ	V30HL	V30MX
サポートする命令	V30HL, V30MX がサポートする命令のうち、次の命令はサポートしていません。これらの命令を実行した場合の結果は、不定となります。 ADD4S, BRKEM, CALLN, CLR1 ^{注1} , CMP4S, EXT, FPO2, INS, NOT1 ^{注2} , REPC, REPNC, RETEM, ROL4, ROR4, SET1 ^{注3} , SUB4S, TEST1 また、FPO1 命令は、NOP 命令として扱われます。		
命令プリフィクスの個数	使用できる命令プリフィクスは7個までです（全命令が対象）。命令プリフィクスが冗長に使用されていても、7個以内であれば正常に動作します。1つの命令に7個以上の命令プリフィクスが存在した場合、（命令プリフィクスが付けられた）命令の実行結果は保証できません。また、割り込み処理からの正常な復帰もできません。	リピート・ストリング命令（REP, MOVBK など）には、3種類までの命令プリフィクスしか使用できません（REP も1種類として計算）。冗長な命令プリフィクスが存在すると、割り込み処理終了後に、正常にリピート・ストリング命令が実行できません。ストリング命令以外の命令では、命令プリフィクスの個数に制限はありません。	
10進補正命令	CVTDB, CVTBD 命令の第2バイト目を基数として補正演算を実行します。	CVTDB, CVTBD 命令の第2バイト目が任意の値であっても、10進補正演算を実行します。	
複数ビットのシフト、ローテート命令	シフト数の下位5ビットだけが有効です。	シフト数（イミーディエト、または CL レジスタによる指定）の8ビットすべてが有効です。	
PREPARE 命令	第2オペランドの下位5ビットだけが有効です。	第2オペランドは8ビットすべてが有効です。	
POP R 命令	8回のメモリ・リード・サイクルを実行します。ただし、SP に相当するデータは使用されません。	SP を除く7回のメモリ・リード・サイクルを実行します。	
リピート・プリフィクス付きの CMPBK, CMPBKB, CMPBKW 命令	メモリ・リードを IX→IY の順で実行します。	メモリ・リードを IY→IX の順で実行します。	
CALL memptr32 命令	現在の PC, PS の値をスタックに退避してから新しい PC, PS の値を読み出します。	新しい PC, PS の値を読み出してから、現在の PC, PS の値をスタックに退避します。	
シフト、ローテート命令でシフト数が0の場合	メモリ・オペランドのライト・サイクルも実行します。SHL, SHR, SARA 命令において、Z フラグ、P フラグ、S フラグが変化します。これらのフラグはシフト命令の実行結果によりセット/クリアされません。	オペランドがメモリの場合は、リード・サイクルだけを実行し、シフト結果のライト・サイクルは実行しません。SHL, SHR, SARA 命令において、Z フラグ、P フラグ、S フラグは変化しません。これらのフラグは命令実行前の状態を保ちます。	

注1. 「CLR1 CY」, 「CLR1 DIR」を除く。

2. 「NOT1 CY」を除く。

3. 「SET1 CY」, 「SET1 DIR」を除く。

項 目	V30MZ	V30HL	V30MX
BUSLOCK 命令	メモリ，または I/O アクセスを行う命令でだけ有効で，その他の命令では無視されます。また，バス・ロック期間中には，コード・フェッチ・バス・サイクルは実行しません。	すべての命令に対して有効です。	BUSLOCK 命令の次の命令の実行中は $\overline{\text{BUSLOCK}}$ 出力がロウ・レベルになり，この間は，バス・ホールド要求を受け付けません。また，バス・ロック期間中にコード・フェッチ・バス・サイクルを実行することがあります。

第2章 端子機能

2.1 端子一覧

端子名称	入出力	機能
A19-A0	出力	アドレス信号出力
DI15-DI0	入力	データ信号入力
DO15-DO0	出力	データ信号出力
UBEB	出力	データ・バス上位バイト・イネーブル信号出力
BS3-BS0	出力	バス・ステータス信号出力
READYB	入力	ウェイト・ステート発生信号入力
BUSLOCKB	出力	バス・ロック信号出力
POLLB	入力	外部システム同期センス信号入力
RESET	入力	システム・リセット信号入力
HLDRQ	入力	バス・ホールド要求信号入力
HLDK	出力	バス・ホールド応答信号出力
NMI	入力	ノンマスカブル割り込み要求信号入力
INT	入力	マスカブル割り込み要求信号入力
CLK	入力	システム・クロック入力
BUNRI	入力	テスト・バスを用いたテストを行うための端子
TEST	入力	
TBI22-TBI0	入力	
TBO42-TBO0	出力	
DBINT	入力	NECの予約端子
DBMODE	出力	
DBA20	出力	
DBRD	出力	
DBWR	出力	
DBNMIM	入力	
DBHLTST	出力	
TEOI	出力	
TILEN3-TILEN0	出力	
TBRA	出力	
TINTA	出力	

2.2 端子状態

動作モードによる各出力端子の動作状態は、次のとおりです。

端子名称	端子状態				
	ノーマル・モード			テスト・モード	
	バス・ホールド	スタンバイ (HALT)モード	リセット	スタンバイ・ テスト・モード	単体テスト・ モード
A19-A0	H	H	H	不定	不定
DO15-DO0	不定	不定	不定	不定	不定
UBEB	H	H	H	不定	不定
BS3-BS0	H	H	H	不定	不定
BUSLOCKB	H	H	H	不定	不定
HLDKAK	H	L	L	不定	不定
TBO42-TBO0	Hi-Z	Hi-Z	Hi-Z	Hi-Z	動作

備考 H : ハイ・レベル出力
 L : ロウ・レベル出力
 Hi-Z : ハイ・インピーダンス
 動作 : 有効な信号を出力

2.3 端子機能の説明

2.3.1 通常端子

(1) A19-A0 (Address) ...出力

20 ビットのアドレスを出力するバスです。

各端子はハイ・インピーダンスになることはありません。

(2) DI15-DI0 (Data input) ...入力

16 ビットのデータを入力する入力専用のバスです。

必ずハイ・レベル, またはロウ・レベルの信号を入力してください (ハイ・インピーダンスにしないでください)。

(3) DO15-DO0 (Data output) ...出力

16 ビットのデータを出力する出力専用のバスです。

各端子は, ハイ・インピーダンスになることはありません。

(4) UBEB (Upper byte enable) ...出力

メモリ, または I/O アクセス・サイクルで 16 ビットのデータ・バスの上位 8 ビットを使用することを示すロウ・アクティブの信号を出力します。ハイ・インピーダンスになることはありません。

この信号がアクティブになるバス・サイクルは次のとおりです。

- 奇数アドレスに対するバイト・アクセスによるバス・サイクル
- 奇数アドレスに対するワード・データのための 1 回目のバイト・アクセスによるバス・サイクル
- 偶数アドレスに対するワード・データのアクセスによるバス・サイクル

A0 信号との組み合わせで, バス・サイクルを次のように識別できます。

表2-1 オペランドとUBEB, A0, および各バス・サイクルの関係

オペランド		UBEB 端子 出力レベル	A0 端子 出力レベル	バス・ サイクル数
偶数アドレスのワード		L	L	1
奇数アドレスのワード	第 1 バス・サイクル	L	H	2
	第 2 バス・サイクル	H	L	
偶数アドレスのバイト		H	L	1
奇数アドレスのバイト		L	H	1

備考 L: ロウ・レベル

H: ハイ・レベル

(5) BS3-BS0 (Bus status) ...出力

バス・サイクルの状態を, 外部に知らせるためのステータス信号を出力します。リセット中とバス・ホールド応答中は, アイドル状態 (ハイ・レベル出力) になります。

ハイ・インピーダンスになることはありません。

なお、BS3 端子は、出力タイミングを除き V30HL の \overline{IO}/M 端子と同一の機能を持ちます（名称が異なるだけです）。

表2 - 2 BS3-BS0信号とバス・サイクルの関係

各端子の出力レベル				バス・サイクル（ステータス）
BS3	BS2	BS1	BS0	
L	L	L	L	割り込みアクノリッジ
L	H	L	H	I/O リード
L	H	H	L	I/O ライト
H	L	L	L	スタンバイ（HALT）モード
H	L	L	H	メモリ・データ・リード
H	L	H	L	メモリ・データ・ライト
H	H	L	H	コード・フェッチ
H	H	H	H	アイドル状態

備考 1. L：ロウ・レベル

H：ハイ・レベル

2. 上記以外の組み合わせでの出力はありません。

(6) READYB (Ready) ...入力

ウエイト制御を行います。

基本バス・サイクル（1クロック）以内にメモリやI/Oがデータのリード/ライト動作を終了できないとき、この端子にインアクティブ・レベル（ハイ・レベル）を入力することにより、バス・サイクルを延長できます。

(7) BUSLOCKB (Bus lock) ...出力

BUSLOCK 命令に続く 1 命令を実行中、ほかのバス・マスタに対して、システム・バスを使用しないように要求するロウ・アクティブの信号を出力します。割り込みアクノリッジ中にも出力します。

ハイ・インピーダンスになることはありません。

(8) POLLB (Poll) ...入力

V30MZ のプログラム実行と外部システムとの同期をとるために使用します。この端子への入力は POLL 命令によりチェックされ、ロウ・レベルが入力されていれば次の命令に移り、ハイ・レベルが入力されていれば、ロウ・レベルになるまでプログラムの実行を停止します。

なお、この端子へのロウ・レベルの入力は、最低 9 クロック・サイクル以上、継続してください。

(9) RESET (Reset) ...入力

リセット信号を入力します。リセット解除後、V30MZ はメモリの FFFF0H（セグメント値：FFFFH、オフセット値：0000H）番地からプログラム実行を開始します。

(10) HLDRQ (Hold request) ...入力

V30MZ に対し、外部のバス・マスタがアドレス・バス、データ・バス、制御バスの解放（バス・ホー

ルド)を要求する信号を入力します。

この端子にハイ・レベルを入力すると、現在実行中のバス・サイクルの終了後にバス・ホールド応答状態に入り、ハイ・レベルを入力する間、バス・ホールド応答状態を継続します。

最低3クロック・サイクル以上の期間、ハイ・レベルを入力してください。

(11) HLDACK (Hold acknowledge) ...出力

HLDQRQ 信号を受け付け、バス・ホールド応答状態であることを示す信号を出力します。

(12) NMI (Non-maskable interrupt) ...入力

ソフトウェアによるマスクが不可能な割り込み要求信号を入力します。

NMI 信号は、立ち上がりエッジ・アクティブで、どのクロック・サイクルでも検出されますが、実際に割り込み処理に入るのは、そのとき実行中の命令終了後になります。

この割り込みに対する割り込み開始アドレスは、割り込みベクタ2によって決定されます。

この端子へは、立ち上がりエッジのあと、最低5クロック・サイクル以上のアクティブ・レベル(ハイ・レベル)を入力してください。

なお、NMI 要求を連続して入力する場合、最低1クロック・サイクルのロウ・レベルを保持してください。

割り込み要求信号の優先順位は次のとおりです。

INT < NMI < HLDQRQ

備考 NMI 信号によりスタンバイ・モードを解除することもできます。

(13) INT (Interrupt request) ...入力

ソフトウェアによるマスクが可能な割り込み要求信号を入力します。

この端子へは、BS3-BS0 端子から割り込みアクノリッジ・ステータスが出力されるまで、アクティブ・レベル(ハイ・レベル)を入力してください。

(14) CLK (Clock) ...入力

クロック信号を入力します。CLK 端子への入力周波数と V30MZ の内部システム・クロック周波数は同一です。

この端子への入力を停止することにより、電源電流は0Aになります。

2.3.2 テスト用端子

(1) TBI22-TBI0 (Test bus input) ...入力

入力テスト・バス端子です。

(2) TBO42-TBO0 (Test bus output) ...出力

出力テスト・バス端子です。

(3) TEST (Test bus control) ...入力

テスト・バス・コントロール入力端子です。

(4) BUNRI (Test bus control) ...入力

ノーマル・モード / テスト・モードを選択する入力端子です。

備考 各端子の機能の詳細については、**第8章 テスト機能**を参照してください。

2.3.3 予約端子

次の各端子は、NECの予約端子です。

2.4 端子の未使用時の処理に示す内容に従って、各端子の処理を行ってください。

- DBINT
- DBMODE
- DBA20
- DBRD
- DBWR
- DBNMIM
- DBHLTST
- TEOI
- TILN3-TILEN0
- TBRA
- TINTA

2.4 端子の未使用時の処理

端子名称	入出力	推奨接続方法
A19-A0	出力	オープンにしてください。
DO15-DO0	出力	
UBEB	出力	
BS3-BS0	出力	
READYB	入力	ロウ・レベルを入力してください。
BUSLOCKB	出力	オープンにしてください。
POLLB	入力	ロウ・レベルを入力してください。
HLDRQ	入力	
HLDAK	出力	オープンにしてください。
NMI	入力	ロウ・レベルを入力してください。
INT	入力	
DBINT	入力	
DBMODE	出力	オープンにしてください。
DBA20	出力	
DBRD	出力	
DBWR	出力	
DBNMIM	入力	ロウ・レベルを入力してください。
DBHLTST	出力	オープンにしてください。
TEOI	出力	
TILEN3-TILEN0	出力	
TBRA	出力	
TINTA	出力	

[メモ]

第 3 章 CPU 機能

3.1 レジスタ構成

3.1.1 汎用レジスタ (AW, BW, CW, DW)

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

したがって、これらのレジスタは転送命令、算術演算命令、論理演算命令など広範な命令に対して 8 ビットまたは 16 ビット・レジスタとして利用されます。

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

- AW : ワード乗除算, ワード入出力, データ変換
- AL : バイト乗除算, バイト入出力, BCD ローテート, データ変換
- AH : バイト乗除算
- BW : データ変換 (テーブル参照)
- CW : ループ制御ブランチ, リピート・プリフィクス
- CL : シフト, ローテート
- DW : ワード乗除算, 間接アドレッシング入出力

3.1.2 セグメント・レジスタ (PS, SS, DS0, DS1)

V30MZ は、メモリ空間を 64K バイト単位の論理セグメントに分割し、同時に 4 つのセグメントまで管理できます (セグメント方式)。各セグメントは、それぞれ次の 4 つのセグメント・レジスタで先頭アドレスが指定されます。

- プログラム・セグメント・レジスタ (PS) : 命令を格納しているセグメントのベース・アドレスを指定
- スタック・セグメント・レジスタ (SS) : スタック動作を行うセグメントのベース・アドレスを指定
- データ・セグメント 0 レジスタ (DS0) : データを格納しているセグメントのベース・アドレスを指定
- データ・セグメント 1 レジスタ (DS1) : データ転送命令で、データの転送先などに使用するセグメントのベース・アドレスを指定

セグメント方式と各セグメント・レジスタの詳細については、3.4 論理アドレスと物理アドレスを参照してください。

3.1.3 ポインタ (SP, BP)

ポインタは 2 本の 16 ビット・レジスタ (スタック・ポインタ (SP), ベース・ポインタ (BP)) から構成されます。

各レジスタは、メモリのアドレスを指定するためのポインタとして使用され、命令中で参照することもできますが、メモリ・データ参照時にはインデックス・レジスタとして用いられます。

SPはスタック・セグメント内の最新のデータが格納されているアドレスを示し、スタック操作時にはデフォルト・レジスタとして使用されます。

BPはスタック上に格納されたデータの取り出しに使用されます。

3.1.4 プログラム・カウンタ (PC)

PCは、現在エグゼキューション・ユニット (EXU) が実行しようとしているプログラム・メモリ・アドレスのオフセット情報を保持する16ビット・バイナリ・カウンタです。

PCの値は、マイクロプログラムが命令キューから命令コードをフェッチするごとに自動的にインクリメント (+1) されます。

また、ブランチ/条件付きブランチ命令、サブルーチン制御命令、および割り込み命令の実行時には、新たなロケーションがロードされ、このときPCの値はプリフェッチ・ポインタ (PFP) の値と同じになります。

3.1.5 プログラム・ステータス・ワード (PSW)

PSWは、6種類のステータス・フラグと4種類のコントロール・フラグで構成されます。

(1) ステータス・フラグ

- オーバフロー・フラグ (V)
- サイン・フラグ (S)
- ゼロ・フラグ (Z)
- 補助キャリー・フラグ (AC)
- パリティ・フラグ (P)
- キャリー・フラグ (CY)

(2) コントロール・フラグ

- モード・フラグ (MD)
- 方向フラグ (DIR)
- 割り込み許可フラグ (IE)
- ブレーク・フラグ (BRK)

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

コントロール・フラグは、命令によってセット/クリアされ、V30MZの動作を制御します。IEフラグとBRKフラグは、割り込み処理が起動されるとクリア (0) されます。

なお、RESET入力によって、MDフラグ以外の全フラグがクリア (0) されます。

また、PSWは、次に示すような処理により、バイトまたはワード単位で操作されます。バイト単位の処理は下位8ビット (Vフラグを除くステータス・フラグが含まれる) に対してだけ行われます。

図3-1 プログラム・ステータス・ワード (PSW)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MD	1	1	1	V	DIR	IE	BRK	S	Z	0	AC	0	P	1	CY

ビット7-ビット0は、MOV 命令によって AH にストアまたはリストアできます。

PSW の全ビットの内容は、割り込み発生時とコール命令 (CALL) 実行時にスタックに退避され、リターン命令 (RET, RETI) によって復帰されます。

また、PUSH PSW 命令、POP PSW 命令によって、単独で PSW の内容をスタックに退避、またはスタックから復帰できます。

フラグは各命令の実行後、次に示すような状態になります。

(a) キャリー・フラグ (CY)

2 進加減算

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

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

インクリメント、デクリメント命令では変化しません。

論理演算

演算結果にかかわらずクリアされます。

2 進乗算

符号なしバイト演算の結果、AH が0以外ならばクリアされます。

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

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

符号なしワード演算の結果、DW が AW のサイン拡張になっていればクリアされ、それ以外ではセットされます。

8ビット・イミューディエト演算の場合は、積が16ビット以内のときはクリアされ、16ビットを越える場合にはセットされます。

2 進除算

不定となります。

シフト/ローテート

CY フラグを含むシフトまたはローテートの場合は、CY フラグへシフトされるビットが1の場合はセットされ、0の場合はクリアされます。

(b) パリティ・フラグ(P)**2 進加減算, 論理演算, シフト**

演算の結果の下位 8 ビットのうち, 1 であるビットの数が偶数個のときセットされ, 奇数個のときクリアされます。

演算結果の各ビットがすべて 0 のときにはセットされます。

2 進乗除算

不定となります。

(c) 補助キャリー・フラグ(AC)**2 進加減算**

バイト演算の場合, 下位 4 ビットから上位 4 ビットへのキャリーまたは上位 4 ビットから下位 4 ビットへのボローがあるときセットされ, それ以外のときクリアされます。

ワード演算では, 下位バイトについてバイト演算の場合と同様の動作を行います。

論理演算, 2 進乗除算, シフト/ローテート

不定となります。

(d) ゼロ・フラグ(Z)**2 進加減算, 論理演算, シフト/ローテート**

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

2 進乗除算

不定となります。

(e) サイン・フラグ(S)**2 進加減算, 論理演算, シフト/ローテート**

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

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

2 進乗除算

不定となります。

(f) オーパフロー・フラグ(V)**2 進加減算**

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

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

2 進乗算

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

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

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

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

8 ビット・イミディエイト演算の場合は、積が 16 ビット以内のときにはクリアされ、16 ビットを超える場合にはセットされます。

2 進除算

クリアされます。

論理演算

クリアされます。

シフト/ローテート

左 1 ビット・シフト/ローテートの場合、演算結果により次のようになります。

- CY = 最上位ビットのとき：クリア
- CY 最上位ビットのとき：セット

右 1 ビット・シフト/ローテートの場合、演算結果により次のようになります。

- 最上位ビット = 最上位の次の下位ビットのとき：クリア
- 最上位ビット 最上位の次の下位ビットのとき：セット

多ビット・シフト/ローテートの場合は不定となります。

(g) ブレーク・フラグ (BRK)

PSW の一部としてスタックに退避されている状態でだけ、メモリ操作命令によってセットでき、セット後に PSW にリストアされると有効となります。

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

(h) 割り込み許可フラグ (IE)

EI 命令でセットされてマスクブル割り込み (INT) 許可状態にし、DI 命令でクリアされてマスクブル割り込み (INT) 禁止状態にします。

(i) 方向フラグ (DIR)

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

(j) モード・フラグ (MD)

V30HL が持つ μ PD8080AF エミュレーション機能に関するフラグです。V30MZ は、このエミュレーション機能をサポートしていないので、このフラグは無効です。

3.1.6 インデクス・レジスタ (IX, IY)

2本の16ビット・レジスタ (IX, IY) から構成されます。メモリ・データ参照時には、実効アドレス生成用のインデクス・レジスタとして用いられます (各レジスタは、命令中で参照することもできます)。

また、特定の命令処理では次のような特別な役割を持っています。

- IX : ブロック・データ操作命令でのソース・オペランド用アドレス・レジスタ
BCD 演算命令でのソース・オペランド用アドレス・レジスタ
- IY : ブロック・データ操作命令でのデスティネーション・オペランド用アドレス・レジスタ
BCD 演算命令でのデスティネーション・オペランド用アドレス・レジスタ

3.2 アドレス空間

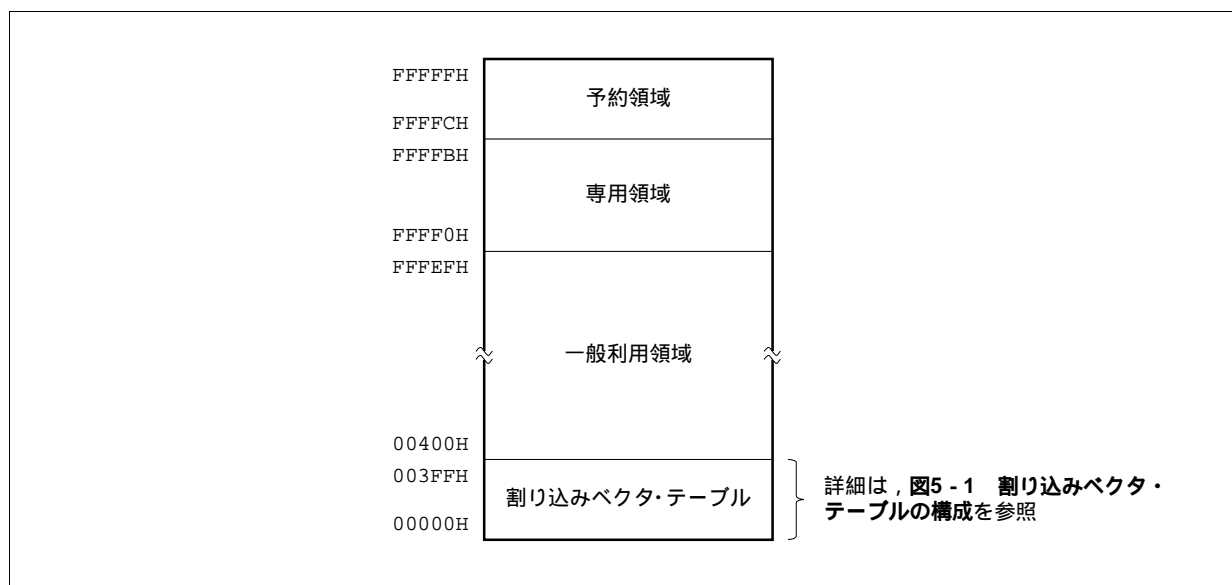
3.2.1 メモリ空間

V30MZ は 20 ビットのアドレス情報を持ち、1M バイト（512K ワード）のメモリをアクセスできます。

図 3 - 2 にメモリ・マップを示します。00000H-003FFH 番地の 1K バイトは、割り込みベクタ・テーブルに割り付けられています。ただし、システムによって使用しないテーブル・エリアはほかの目的に使用できます。

V30MZ のリセット後のスタート・アドレスは、FFFF0H 番地です。したがって FFFF0H-FFFFBH 番地の 12 バイトは、リセット・スタートなどで自動的に使用するので、ほかの目的に使用できません。また、FFFFCH-FFFFFH 番地の 4 バイトも、将来の使用のため予約されているので、ユーザは使用できません。

図3-2 メモリ・マップ



メモリ空間にストアされる要素としては命令コード、割り込み開始アドレス、スタック・データ、一般変数などがあり、バイト単位のものと同ワード単位のものがあります。

これらの要素に対して命令によって生成されるアドレスとしては、偶数（A0=0）と奇数（A0=1）の両方が考えられます。V30MZ のワード・データのアクセスは、そのアドレスが偶数でも奇数でも可能な設計となっており、命令の生成アドレスとして偶数、奇数の両方が可能です。

アクセス方法については 4.1 メモリとのインタフェースを参照してください。

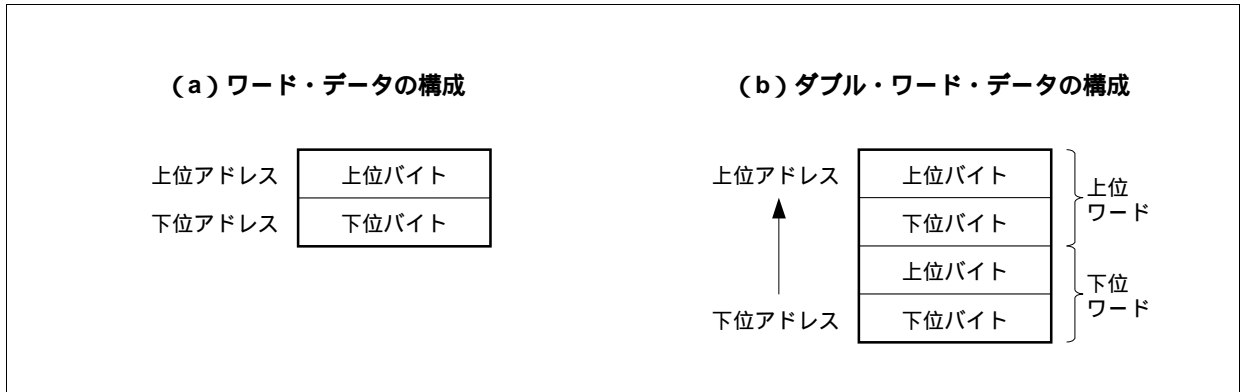
各メモリ要素のアドレス、データ構成を表 3 - 1 に示します。

表3-1 各メモリ要素のアドレス、データ構成

メモリ要素	アドレス	データ構成
命令コード	偶数 / 奇数	1-6 バイト
割り込みベクタ・テーブル	偶数	2 ワード / ベクタ
スタック	偶数 / 奇数	ワード
一般変数	偶数 / 奇数	バイト / ワード / ダブル・ワード

ワード・データとダブル・ワード・データの構成は次のとおりです。

図3-3 ワード・データとダブル・ワード・データの構成



3.2.2 I/O 空間

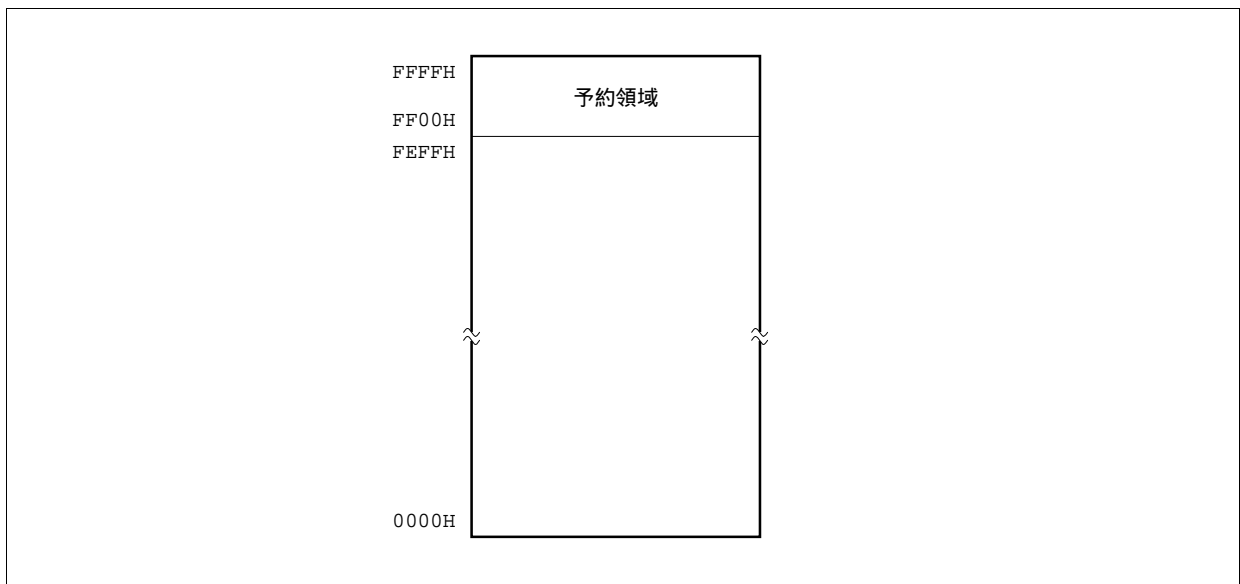
V30MZ は 64K バイト (32K ワード) までの I/O 空間を、メモリ空間とは独立した領域でアクセスできます。

I/O 空間はアドレス・バスの下位 16 ビットより出力される I/O アドレス情報によってアドレスされます。

図 3 - 4 に I/O マップを示します。なお、FF00H-FFFFH 番地の 256 バイトは、将来の使用のために予約されているので、ユーザは使用できません。

アクセス方法については、4.2 I/O とのインタフェースを参照してください。

図3-4 I/Oマップ



3.3 命令プリフェッチ

V30MZ は内部でパイプライン処理を行っており、命令フェッチ（プリフェッチ）と命令デコード、および命令実行を並列に行っています。そのため命令コード・フェッチのアドレス・バスの出力を観測して、現在、プログラムのどの部分を実行しているかを把握することは難しくなっています。

また条件分岐命令がある場合には、分岐不成立の場合でも、分岐先のアドレスをプリフェッチするので（1回のみ）、さらにプログラムの追跡は難しくなります。

なお、V30MZ は、プリフェッチ・キューを8個（16バイト）持っています。

3.4 論理アドレスと物理アドレス

メモリ空間のアドレスには論理アドレスと物理アドレスの2種類があります。

物理アドレスはハードウェアに直接対応したアドレスを意味します。

V30MZは1Mバイトのメモリ空間をアクセスすることができるので、物理アドレスの値の範囲は00000H-FFFFFFHです。物理アドレスの生成は命令をフェッチしたり、データを転送したりするバス・コントロール・ユニット(BCU)が起動するたびに行われます。

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

3.4.1 セグメント方式

セグメントとは、プログラム作成の際に直接依存しない小単位(最大64Kバイト)のアドレス空間のことです。

各セグメントは連続したメモリから構成され、それぞれ個別に指定できます。

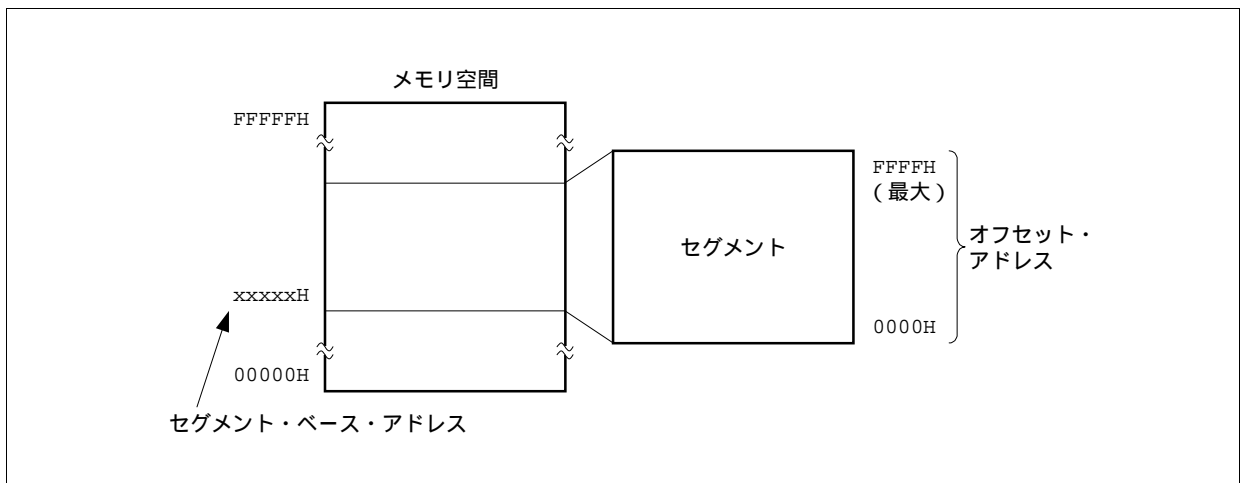
マシン語のプログラム作成では、物理アドレスを直接管理することはできません。V30MZでは、メモリのアドレスをセグメント方式で指定します。

セグメント方式のアドレス指定には、次の2種類のアドレスを使用します。

- セグメント・ベース・アドレス : セグメントの先頭アドレス(1Mバイト・メモリ空間内でのアドレス)
- オフセット・アドレス : 各セグメントごとに付けられたアドレス

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

図3-5 セグメント方式の概念図



セグメント・ベース・アドレスはセグメント・レジスタによって指定します。

物理アドレスはセグメント・ベース・アドレスにオフセット・アドレスを加えた値となります。図3-6にセグメント・レジスタ、オフセット・アドレスと物理アドレスの関係を示します。

図3-6 セグメント・レジスタ、オフセット・アドレスと物理アドレスの関係

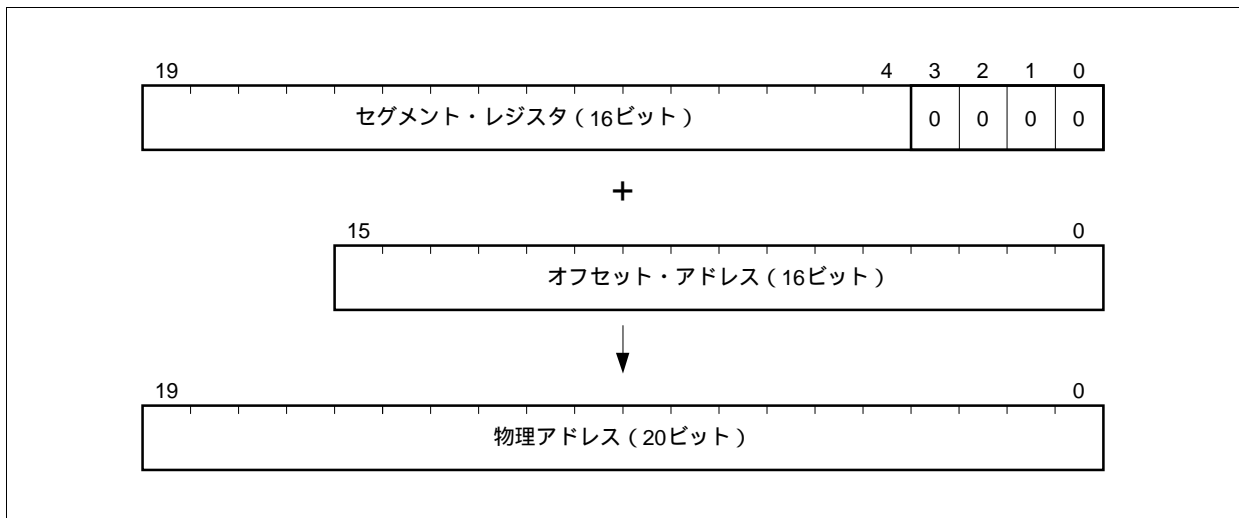


図3-6に示すように、セグメント・レジスタの内容を16倍(4ビット左へシフト)したものにオフセット値を加えたものが物理アドレスになります。このとき、セグメント・レジスタの内容とオフセット値は符号なしデータとして扱います。

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

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

3.4.2 セグメントの構成

V30MZは4種類のセグメント(プログラム、スタック、データ0、データ1)を区別して定義することができます。各セグメントは、それぞれ次の4つのセグメント・レジスタで開始アドレスが指定されます。

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

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

各セグメント内のオフセット・アドレスは、特定のレジスタが、実効アドレスによって指定されます。セグメント・レジスタとオフセット・アドレス指定の対応を表3-2に示します。

表3-2 セグメント・レジスタとオフセット・アドレス指定の対応

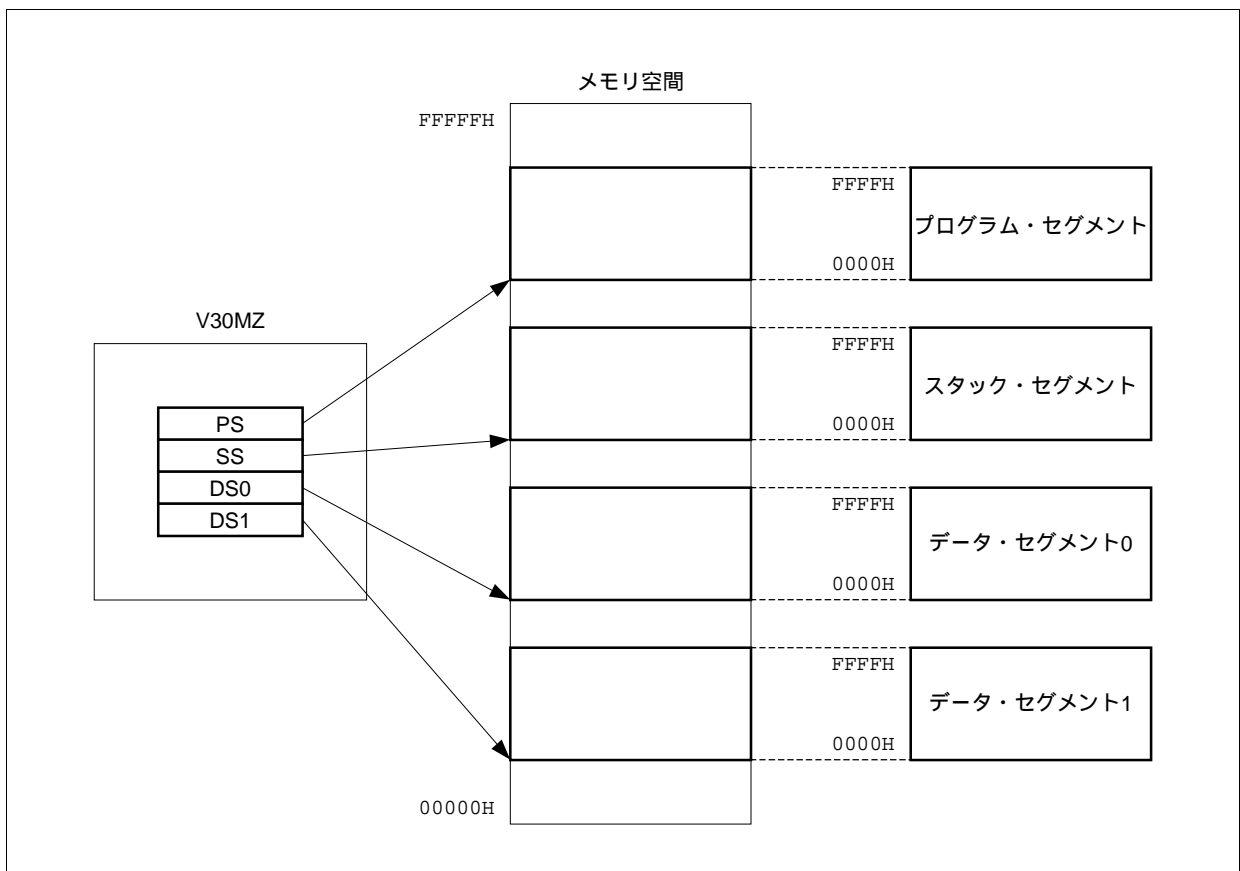
オフセット	セグメント・レジスタ	デフォルト	オーバーライド
PFP		PS	不可
SP		SS	不可
実効アドレス (BP ベース)			PS, DS0, DS1
実効アドレス (非BP ベース)		DS0	PS, SS, DS1
命令群 A (プリミティブ・ブロック転送命令, プリミティブ出力命令, BCD 演算命令) での IX			
命令群 B (プリミティブ・ブロック転送命令, プリミティブ入力命令, BCD 演算命令) での IY		DS1	不可

デフォルト・オフセットがプリフェッチ・ポインタ (PFP), スタック・ポインタ (SP), 命令群 B でのインデクス・レジスタ (IY) の場合は, 組み合わせることのできるセグメント・レジスタはそれぞれ PS, SS, DS1 に固定されており, ほかのセグメント・レジスタは使用できません。

そのほかのデフォルト・オフセットでは, デフォルトのセグメント・レジスタ以外に任意のセグメント・レジスタをセグメント・オーバーライド・プリフィクスによって指定できます。

図3-7に各セグメント・レジスタとセグメント, およびメモリ空間の関係を示します。

図3-7 各セグメント・レジスタとセグメント, およびメモリ空間の関係



各セグメントは, 次のような意味を持っています。

(1) プログラム・セグメント

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

このセグメントには、命令コード、テーブル・データなどが配置されます。

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

(2) スタック・セグメント

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

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

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

(3) データ・セグメント 0

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

このセグメントは、一般変数の格納エリアに利用されます。

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

ベース・アドレスとして BP を用いた場合の実効アドレスでは、デフォルトとしてスタック・セグメントが使用されますが、セグメント・オーバーライド・プリフィクス (DS0:) を用いればデータ・セグメント 0 を利用できます。

(4) データ・セグメント 1

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

命令群 B 実行時での、デスティネーション・データ領域として利用できます。この場合、インデックス・レジスタ (IX) の内容がオフセットとなります。

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

3.4.3 ダイナミック・リロケーション

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

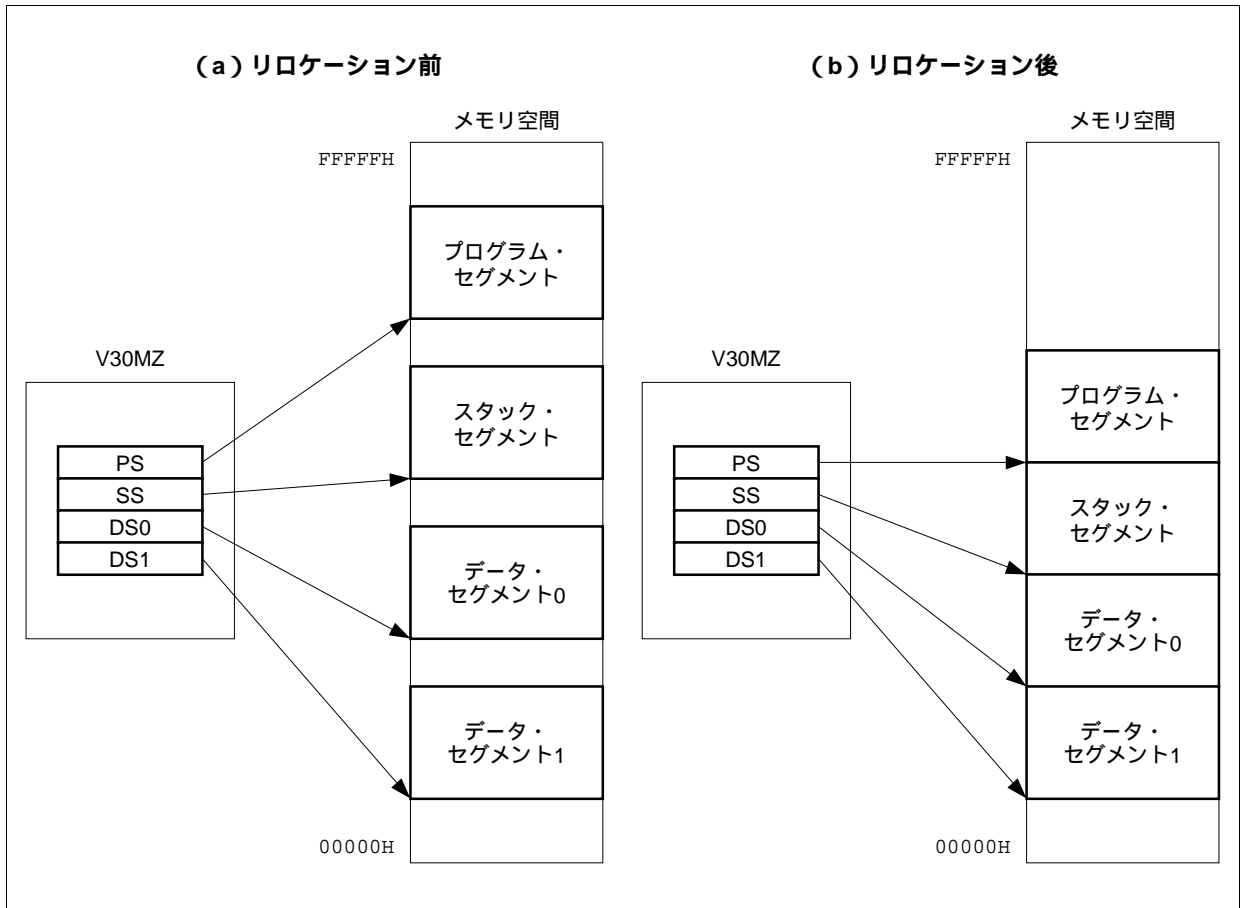
図 3-8 にダイナミック・リロケーションの概念図を示します。

V30MZ では、プログラムのメモリ・アドレス指定は、すべて各セグメントのベース・アドレス (各セグメント・レジスタにより指定) に対するオフセット値だけで決定できます。したがって、プログラムを配置する際、セグメント・レジスタの内容を、そのプログラムを配置しようとしているメモリの物理アドレスに合わせるだけでメモリの任意空間に配置することができます (ただし、プログラムの中で各セグメントのベース・ア

ドレスを変更をするような処理を行わない場合にかぎります)。このため、プログラムのメモリへの配置の自由度が高く（16バイト単位でのアドレス指定が可能）、メモリのより効果的な使用が可能になり、さらに複数のジョブやタスクを実行するシステムを容易に実現できます。

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

図3-8 ダイナミック・リロケーション



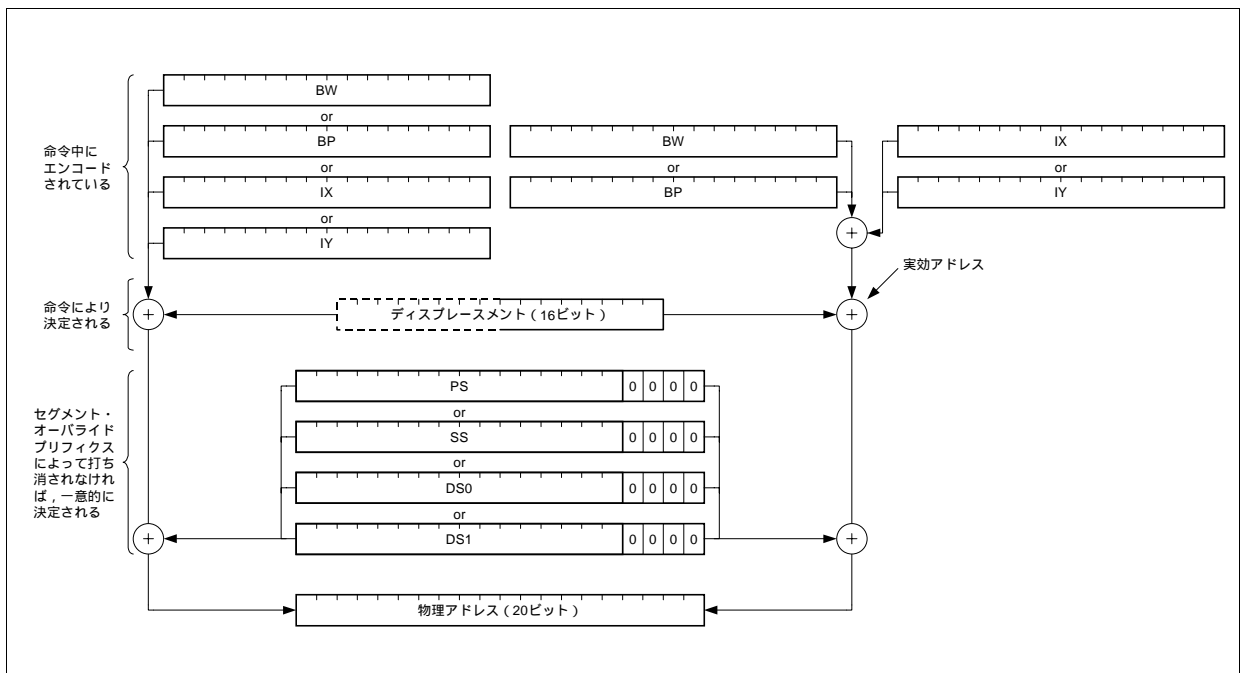
3.5 実効アドレス

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

EXU は EA を数種の異なる方法 (アドレッシング・モード) で計算しますが、方法の選択は命令の 2 バイト目のオペランドで指定します。命令の 2 バイト目にエンコードされた情報は、オペランドで示されるメモリの実効アドレスを EXU がどのように計算するかを表します。このオペランド・コードは、コンパイラまたはアセンブラによりプログラムのステートメントまたは命令記述から自動的に生成されます。アセンブリ言語では、すべてのアドレッシング・モードを使用できます (3.7 アドレッシング・モード参照)。

EA の計算方法を次に示します。図 3 - 9 は EXU がディスプレースメント、ベース・レジスタの内容、およびインデックス・レジスタの内容を加算することによって EA を計算することを示しています。どの命令でも、この 3 つの要素を任意に組み合わせることができます。ディスプレースメントは、オペランドで示される 8 ビット、または 16 ビットのイミディエト数です。

図3-9 メモリ・アドレスの計算



3.6 命令セット

3.6.1 機能別命令セット一覧

V30MZの命令セットを機能別に大別すると次のようになります。

表3-3 機能別命令セット一覧

命令群	二モニック
データ転送命令	LDEA, MOV, TRANS, TRANSB, XCH
リピート・プリフィクス	REP, REPE, REPNE, REPZ, REPZ
プリミティブ・ブロック転送命令	CMPBK, CMPBKB, CMPBKW, CMPM, CMPMB, CMPMW, LDM, LDMB, LDMW, MOVBK, MOVBKB, MOVBKW, STM, STMB, STMW
入出力命令	IN, OUT
プリミティブ入出力命令	INM, OUTM
加減算命令	ADD, ADDC, SUB, SUBC
増減命令	DEC, INC
乗算命令	MUL, MULU
除算命令	DIV, DIVU
BCD補正命令	ADJ4A, ADJ4S, ADJBA, ADJBS
データ変換命令	CVTBD, CVTBW, CVTDB, CVTDL
比較命令	CMP
補数演算命令	NEG, NOT
論理演算命令	AND, OR, TEST, XOR
ビット操作命令	CLR1 CY, CLR1 DIR, SET1 CY, SET1 DIR, NOT1 CY
シフト命令	SHL, SHR, SHRA
ローテート命令	ROL, ROLC, ROR, RORC
サブルーチン制御命令	CALL, RET
スタック操作命令	DISPOSE, POP, PREPARE, PUSH
ブランチ命令	BR
条件付きブランチ命令	BC, BCWZ, BE, BGE, BGT, BH, BL, BLE, BLT, BN, BNC, BNE, BNH, BNL, BNV, BNZ, BP, BPE, BPO, BZ, BV, DBNZ, DBNZE, DBNZNE
割り込み命令	BRK, BRKV, CHKIND, RETI
CPU制御命令	BUSLOCK, DI, EI, FPO1 ^注 , HALT, NOP, POLL
セグメント・オーバーライド・プリフィクス	DS0 : , DS1 : , PS : , SS :

注 NOP命令として扱われます。

備考1. V30HLがサポートする命令のうち、次の命令はサポートしていません。これらの命令を実行した場合の結果は、不定となります。

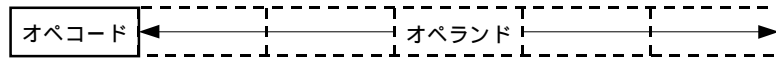
ADD4S, BRKEM, CALLN, CLR1 (CLR1 CY, CLR1 DIRを除く), CMP4S, EXT, FPO2, INS, NOT1 (NOT1 CYを除く), REPC, REPNC, RETEM, ROL4, ROR4, SET1 (SET1 CY, SET1 DIRを除く), SUB4S, TEST1

2. 各命令の詳細については、16ビットVシリーズ ユーザーズ・マニュアル 命令編を参照してください。

3.6.2 命令語の形式

命令語（オブジェクト・コード）は基本的に次の形式で表されます。

図3 - 10 命令語形式



備考 オペコード：命令の種類を表す8ビットのコードです。

オペランド：命令の処理の対象となるレジスタ，メモリ・アドレスを示すフィールドです。

0-5バイトのフィールドで表されます。

3.7 アドレッシング・モード

3.7.1 命令アドレス

命令アドレスとは、命令コードを読み出すアドレスのことで、通常は命令コードの読み出しごとに自動的にインクリメント(+1)されます。しかし、ジャンプ命令やサブルーチン・コール命令など、命令実行のシーケンスを制御する命令では、分岐先の命令アドレスをオペランドで指定します。

(1) ダイレクト・アドレッシング

命令コード中の4バイト・データが命令アドレスとなり、PSとPCの両レジスタにロードされます。このモードは次の命令で用いられます。

```
CALL far_proc
BR far_label
```

(2) PC 相対アドレッシング

命令コード中の1バイトまたは2バイト・データが、次の命令の先頭アドレス(PCの値)からのディスプレースメントとなり、PCに加算されます。このモードは次の命令で用いられます。

```
CALL near_proc
BR near_label
BR short_label
Bcondition short_label ;例 BZ short_label
BNC short_label
```

(3) レジスタ間接アドレッシング

命令コード中のレジスタ指定フィールドで指定される、任意の16ビット・レジスタの内容が命令アドレスとなり、PCにロードされます。このモードは次の命令で用いられます。

```
CALL regptr16 ;例 CALL AW
BR regptr16 ;例 BR IX
```

(4) メモリ間接アドレッシング

命令コード中のアドレッシング・モード指定フィールドが示す、メモリ・アドレッシング(3.7.2 データ・アドレス参照)で指定される、メモリ中の2バイトまたは4バイト・データが命令アドレスとなり、PCまたは、PSとPC両方に直接ロードされます。このモードは次の命令で用いられます。

```
CALL memptr16 ;例 CALL word_var [ BW ]
CALL memptr32 ;例 CALL dword_var [ BW + IX ]
BR memptr16 ;例 BR word_var [ BR + 2 ]
BR memptr32 ;例 BR dword_var [ BP + IY ]
```

3.7.2 データ・アドレス

データ・アドレスとは、各命令のオペランド・データの読み出し／書き込みをするためのアドレスのことです。通常、アドレスはメモリやI/Oに対しての概念ですが、ここでのオペランド・アドレスはレジスタ内データ、イミディエト・データおよびI/Oデータを含んでいます。

(1) 非メモリ・アドレッシング

非メモリ・アドレッシングは、レジスタ内データ、イミディエト・データおよびI/Oデータを指定します。

(a) レジスタ・アドレッシング

命令コード中のレジスタ指定フィールドが、オペランド・データを読み出し／書き込みするレジスタを指定します。レジスタ・アドレッシングは次の記述で示されます。

一般記述	記述可能なレジスタ
reg, reg'	AW, BW, CW, DW, SP, BP, IX, IY, AL, AH, BL, BH, CL, CH, DL, DH
reg8, reg8'	AL, AH, BL, BH, CL, CH, DL, DH
reg16, reg16'	AW, BW, CW, DW, SP, BP, IX, IY
sreg	PS, SS, DS0, DS1
acc	AW, AL

例 reg16 の場合 MOV AW, IX ; AW IX
reg8 の場合 ADD AL, CH ; AL AL + CH

(b) イミディエト・アドレッシング

命令コード中の1バイトまたは2バイト・データが、読み出し専用のオペランド・データとなります。イミディエト・アドレッシングは、命令のデスティネーション・オペランドには使用できません。イミディエト・アドレッシングは、次の記述で示されます。

一般記述	記述可能な値
imm8	0 ~ FFH (0 ~ 255 または -128 ~ +127)
imm16	0 ~ FFFFH (0 ~ 65535 または -32768 ~ +32767)
imm	0 ~ FFFFH (0 ~ 65535 または -32768 ~ +32767)
pop_value	0 ~ FFFFH (0 ~ 65535) ... 通常は偶数

例 imm16 の場合 MOV AW, 216 ; AW 216
imm8 の場合 SHL AL, 5 ; AL を5ビット左にシフトする
pop_value の場合 RET 16 ; スタック上の不要な16バイトを削除する

(c) I/O アドレッシング

I/O アドレッシングは、64KバイトのI/O空間内のデータを指定します。I/O アドレッシングには次の2種類の指定方法があり、入出力命令で用いられます。

imm8

命令コード中の8ビット・データが、I/Oアドレスを示します。この指定方法では、64KバイトのI/O空間のうち、下位側の256バイト空間の指定に限定されます。この指定方法を用いるのは次の2命令です。

```
IN    acc, imm8
OUT  imm8, acc
```

DW

DWレジスタの内容が、I/Oアドレスを示します。この指定方法では、64KバイトのI/O空間すべてにわたって指定できます。この指定方法を用いるのは次の4命令です。

```
IN    acc, DW
INM  dst_block, DW
OUT  DW, acc
OUTM DW, src_block
```

(2) メモリ・アドレッシング

メモリ・アドレッシングはメモリ中にあるオペランド・データを指定します。このメモリ・アドレッシングは、オペコードのあとに置かれる5ビットのメモリ・アドレッシング指定フィールドによって、さらにいくつかのモードに分類されます。メモリ・アドレッシングのすべてのモードは、デフォルトまたはセグメント・オーバーライドによって指定される、セグメント・ベースからの16ビット・オフセット・アドレスを指定します。メモリ・アドレッシングは次の記述で示されます。

記 述	データ長
dmem ^注	8/16ビット・データ
mem	8/16ビット・データ
mem8	8ビット・データ
mem16	16ビット・データ

注 メモリ・アドレッシング指定フィールドを持たない命令での記述。

(a) ダイレクト・アドレッシング

命令コード中の2バイト・データが、オペランド・データの読み出し/書き込み対象となるメモリ・アドレスを示します。

例 MOV byte_var, 216 ; bytemem (offset (byte_var)) 216

(b) レジスタ間接アドレッシング

命令コード中のメモリ・アドレッシング指定フィールドによって指定される16ビット・レジスタ (BWまたはIXまたはIY) が、オペランド・データの読み出し/書き込み対象となるメモリ・アドレスを示します。


```
例 MOV word ptr [ BW ], 10 ; wordmem ( BW ) 10
    ADD AL, byte ptr [ IX ] ; AL AL + bytemem ( IX )
```

(c) ベース・アドレッシング

命令コード中のメモリ・アドレッシング指定フィールドによって指定される 16 ビット・ベース・レジスタ (BW または BP) に、命令コード中の 1 バイトまたは 2 バイト・データで示されるディスプレースメントを符号拡張して加算した値が、オペランド・データの読み出し / 書き込み対象となるメモリ・アドレスを示します。

ベース・レジスタとして BP を選択した場合、デフォルトのセグメント・レジスタは SS となるので、手続き呼び出し時に引き数としてスタックにプッシュしたデータを、手続きの中からアクセスする場合に利用できます。

```
例 MOV word_var [ BW + 2 ], AW ; wordmem ( offset ( word_var ) + BW + 2 ) AW
    SUB AW, [ BP + 6 ] ; AW AW - wordmem ( BP + 6 )
```

(d) インデクスト・アドレッシング

命令コード中のメモリ・アドレッシング指定フィールドによって指定される 16 ビット・インデクス・レジスタ (IX または IY) に、命令コード中の 1 バイトまたは 2 バイト・データで示されるディスプレースメントを符号拡張して加算した値が、オペランド・データの読み出し / 書き込み対象となるメモリ・アドレスを示します。

```
例 MOV word_var [ IY + 2 ], 0 ; wordmem ( offset ( word_var ) + IY + 2 ) 0
    SUB AW, [ IX + 6 ] ; AW AW - wordmem ( IX + 6 )
```

(e) ベース・インデクス付きアドレッシング

命令コード中のメモリ・アドレッシング指定フィールドによって指定される 16 ビット・ベース・レジスタ (BW または BP) に、命令コード中の 1 バイトまたは 2 バイト・データで示されるディスプレースメントを符号拡張して加算した値に、さらに 16 ビット・インデクス・レジスタ (IX または IY) を加算した値が、オペランド・データの読み出し / 書き込み対象となるメモリ・アドレスを示します。すなわち、ベース・アドレッシングとインデクスト・アドレッシングを合わせたようなアドレッシングを行います。

このアドレッシングは、2 次元配列などの構造を持つデータのアクセスに利用できます。

```
例 MOV word_var [ BW + 6 ] [ IY + 2 ], 0 ; wordmem ( offset ( word_var ) + BW + 6 + IY + 2 ) 0
    SUB AW, [ BP + 6 + IX ] ; AW AW - wordmem ( BP + IX + 6 )
```

[メモ]

第 4 章 バス制御機能

V30MZ は、1 クロックで 1 回のバス・サイクルを実行します。

同一チップに集積したメモリに 1 クロックでアクセスすることが可能なので、ウエイト挿入機能を併用することにより、小容量で高速のオンチップ・メモリと、大容量で安価な外部メモリの両方の特性を生かしたシステムを構築できます。

4.1 メモリとのインタフェース

V30MZ はデータ・バス幅が 16 ビットのため、本来 16 ビットのワード・データを 1 バス・サイクルで転送する能力を持っていますが、これは命令によって生成されたアドレスが偶数 ($A0=0$) の場合だけで、奇数 ($A0=1$) の場合はワード・データの転送に 2 バス・サイクルを必要とします。

図 4 - 1 に V30MZ とメモリのインタフェースを示します。

図 4 - 1 V30MZ とメモリのインタフェース

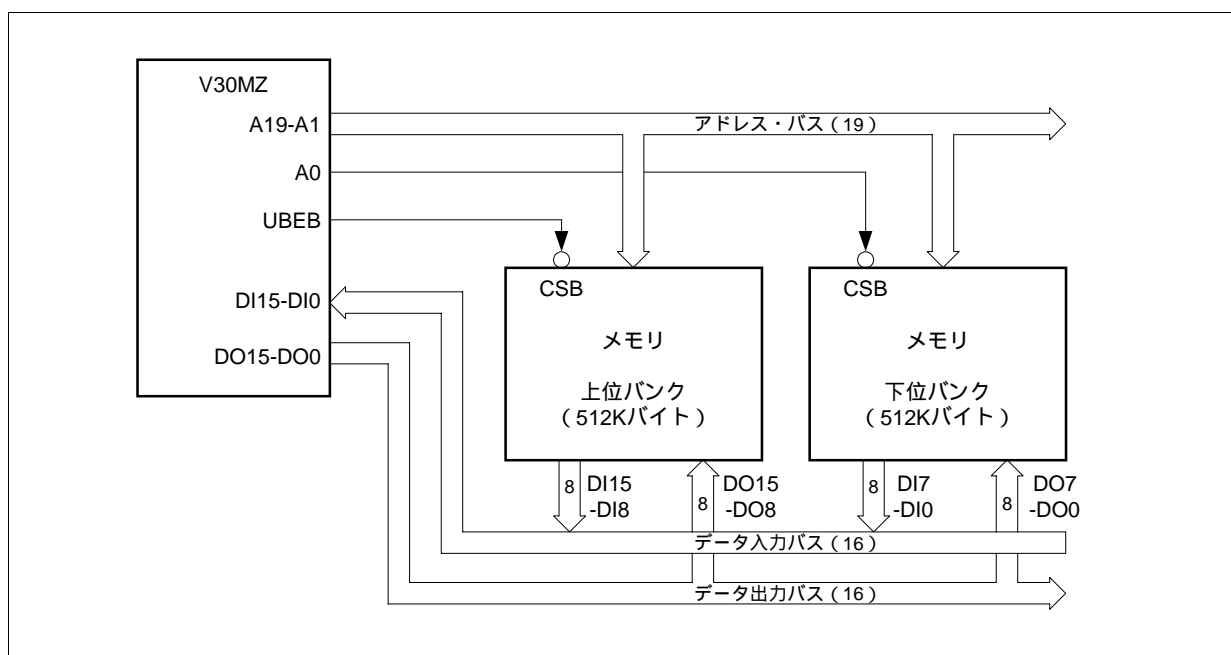


図 4 - 1 で、A0 はアクティブ・ロウでメモリの下位バンクのバイト・データを有効にします。また、アドレス・バスからの情報とは別に UBEB 信号が出力されて、アクティブ・ロウでメモリの上位バンクのバイト・データを有効にします。

(1) 奇数アドレスのワード・データにアクセスする場合

最初のバス・サイクルで $UBEB=0$, $A0=1$ となって上位バイトだけがアクセスされ、続いて自動的に $UBEB=1$ となり、アドレス情報の下位 16 ビット ($A15-A0$) がインクリメント (+1) され、すなわち $A0=0$ となって次のアドレスの下位バイトがアクセスされます。

(2) 偶数アドレスのワード・データにアクセスする場合

ワード・データは、UBEB=0, A0=0 によって1バス・サイクルでアクセスされます。

表4-1にオペランドのタイプとUBEB, A0端子およびバス・サイクル数の関係を示します。

表4-1 V30MZのデータ・アクセス

オペランド		UBEB 端子 出力レベル	A0 端子 出力レベル	バス・サイクル数
偶数アドレスのワード		L	L	1
奇数アドレスのワード	第1バス・サイクル	L	H	2
	第2バス・サイクル	H	L	
偶数アドレスのバイト		H	L	1
奇数アドレスのバイト		L	H	1

備考 L : ロウ・レベル

H : ハイ・レベル

通常、命令コードのアクセス（プリフェッチ）は、ワード単位で行います。ただし、奇数アドレスへのブランチが起こった場合、その奇数アドレスの1バイトだけがフェッチされ、それ以降は再びワード単位にフェッチされます。

割り込みベクタ・テーブルのアクセスは、ベクタ番号（0-225）からベクタ・テーブルのアドレスが生成されるとき、必ず偶数アドレスが生成されるので、常に偶数アドレスのワード・データとして行われます。したがって、1つの割り込みに対するベクタ・テーブルのアクセスは、セグメント・ベースとオフセットの2ワードに対して常に2バス・サイクルで行われます。

4.1.1 ワード・データをアクセスする場合の注意事項

メモリ・アクセスのための1つのバス・サイクルは1クロックを必要とします。したがって、奇数アドレスのワード・データを一度アクセスするごとに、偶数アドレスのワード・データをアクセスするのに比べて命令実行時間が1クロック余分に必要となります。ワード・データ・アクセス回数が1以上ある命令を実行する際に、このことが適用されます。

メモリからメモリへのワード・データ転送の場合、ソースからの読み出しのためとデスティネーションへの書き込みに2回のメモリ・アクセスが必要なため、両方とも奇数アドレスのとき実行時間は最大となります。

これらの奇数アドレスの問題は、スタック操作に対しても同様に起こります。割り込み処理でレジスタなどが自動的にスタックに退避されますが、これらはすべてワード・データになっているので、奇数アドレスで処理された場合、バス・サイクル数が2倍になり、割り込み応答時間を遅くするので考慮が必要です。

以上のことから、ワード・データをアクセスする場合、プログラマがチェックできるものはすべて偶数アドレスに配置できるように考慮してください。

例 MOV reg, mem 命令の実行クロック数

バイト・データのとき : 1

ワード・データのとき : 2 (奇数アドレスのとき)

: 1 (偶数アドレスのとき)

これはワード・データ・アクセスが1回行われる例です。

4.2 I/O とのインタフェース

I/O 空間に対しては、メモリ空間のようにセグメント方式は適用されていません。

I/O アドレス出力タイミングでは、アドレス・バスの上位 4 ビット (A19-A16) にはすべて 0 が出力されます。

V30MZ と I/O のデータ転送は、バイトまたはワード単位の両方が可能で、8 ビット I/O デバイス、または 16 ビット I/O デバイスの両方が接続可能です。ただし、メモリの場合と同様に、ワード・データ・アクセスのとき偶数アドレスなら 1 バス・サイクル、奇数アドレスなら 2 バス・サイクルとなっています。

8 ビット I/O デバイスをアクセスする際、I/O アドレス情報のうち A0 はデバイス選択にだけ使用し、A1 より上位の信号はデバイス選択と 1 つのデバイス内のいくつかのレジスタ選択に使用します。つまり、偶数アドレスの I/O デバイスは、その内部レジスタもすべて偶数で、奇数アドレスの I/O デバイスはその内部レジスタもすべて奇数で選択されるようにします。

なお、メモリ・マップト I/O 構成 (メモリの任意のエリアを I/O 用に割り付けて使用する) を利用すれば、I/O を I/O 空間にではなく 1M バイトのメモリ空間に配置できます。

このことを利用すれば、メモリ空間に対する各種のアドレッシング・モードや演算処理などを I/O デバイスに対して直接行えます。

注意 メモリ・マップト I/O では、V30MZ から出力される制御信号は、すべてメモリに対するものとなるので、I/O デバイスはアドレス情報でだけ区別されます。したがって、変数やスタティック・データなどのアドレスと I/O に割り付けられたアドレスが競合しないように注意してください。

4.3 メモリ、I/O のリード/ライト・タイミング

V30MZ は最低 1 クロックで 1 回のバス・サイクルを実行します。

4.3.1 メモリ、I/O のリード・タイミング

クロック (CLK) の立ち上がりエッジに同期して、アドレス (A19-A0)、UBEB 信号、バス・ステータス (BS3-BS0) を出力します。

データ (DI15-DI0) は、次の CLK 信号の立ち上がりエッジで取り込まれ、同時に READYB 信号をサンプリングします。このとき、READYB 信号がロウ・レベルであれば次のバス・サイクルに移行し、ハイ・レベルであればウエイト・サイクル (TW) に移行して、現在のバス・サイクルを延長します。TW ステートでは、A19-A0 信号と UBEB 信号は出力値を継続しますが、BS3-BS0 信号はハイ・レベルになります。

図4-2 メモリ、I/Oのリード・タイミング (1/2)

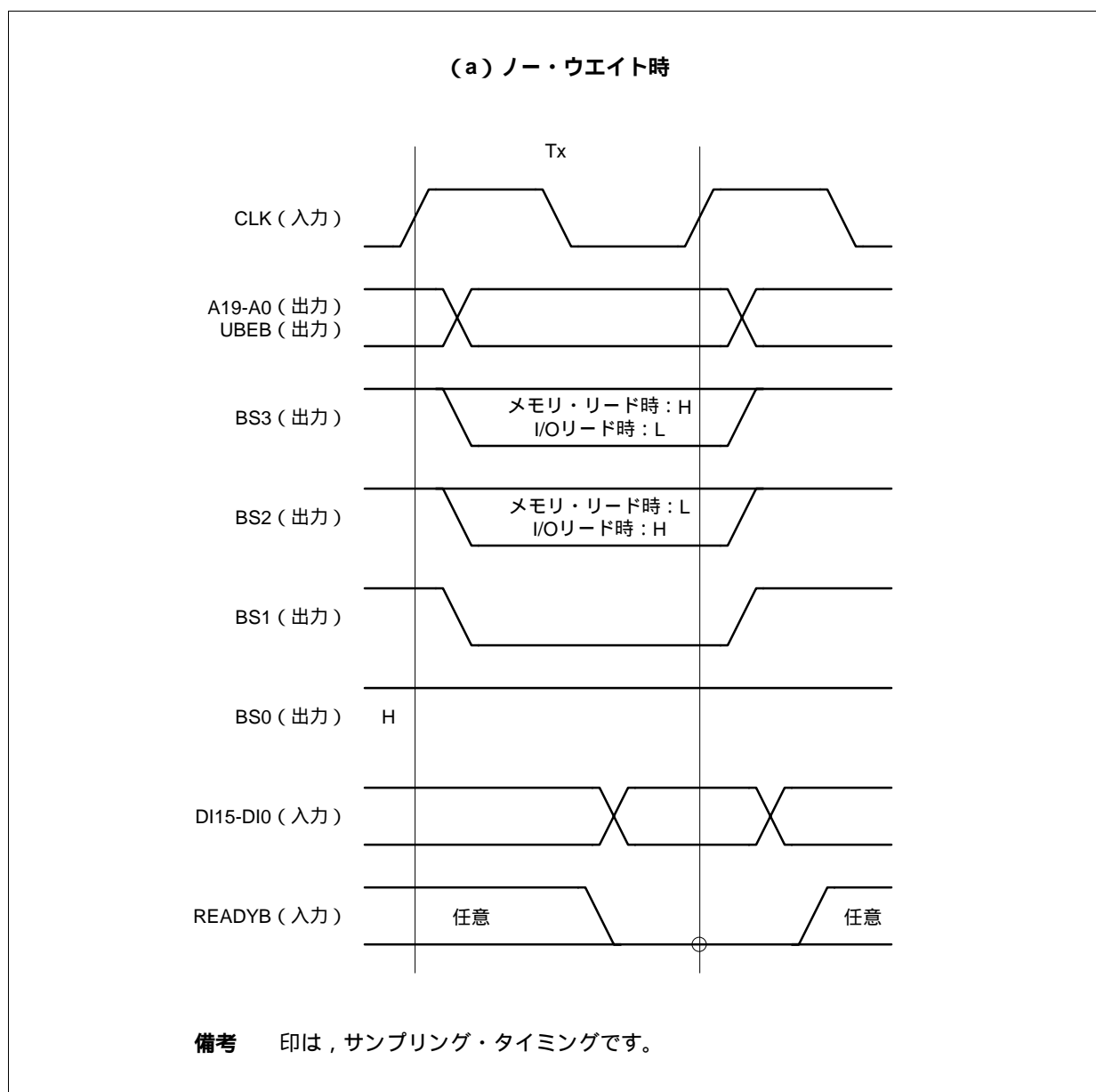
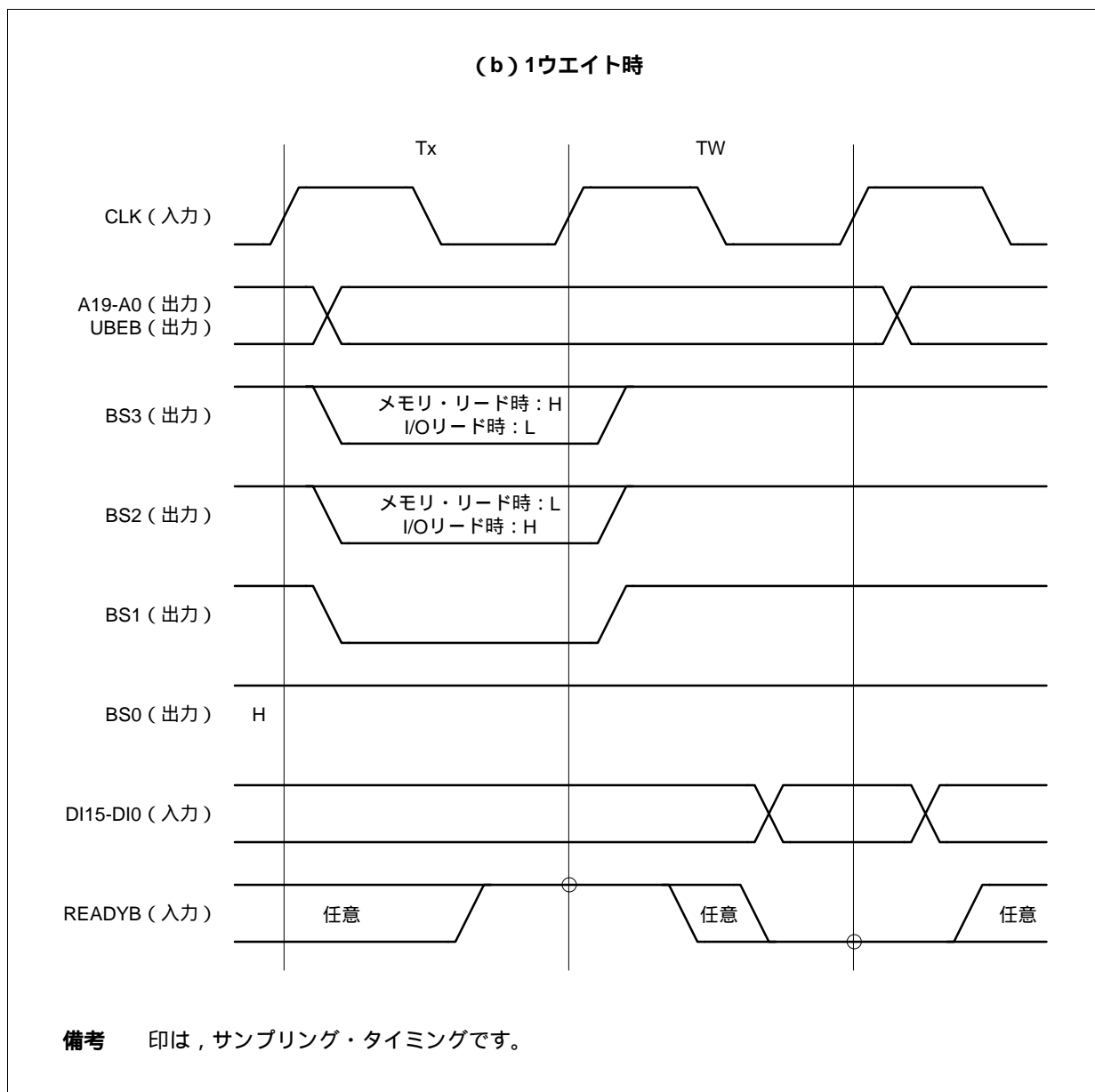


図4-2 メモリ、I/Oのリード・タイミング (2/2)



4.3.2 メモリ、I/Oのライト・タイミング

クロック（CLK）の立ち上がりエッジに同期して、アドレス（A19-A0）、UBEB 信号、バス・ステータス（BS3-BS0）、データ（DO15-DO0）を出力します。

続いて、次の CLK 信号の立ち上がりエッジで READYB 信号をサンプリングします。このとき、READYB 信号がロウ・レベルであれば次のバス・サイクルに移行し、ハイ・レベルであればウエイト・サイクル（TW）に移行して、現在のバス・サイクルを延長します。TW ステートでは、DO15-DO0 端子からは無効なデータを出します。そのため、TW ステートを挿入してバス・サイクルを延長する場合は、最初のバス・サイクルで出力される DO15-DO0 のデータを外部回路でラッチしてください。

また、TW ステートでは、A19-A0 信号と UBEB 信号は出力値を継続しますが、BS3-BS0 信号はハイ・レベルになります。

図4-3 メモリ、I/Oのライト・タイミング (1/2)

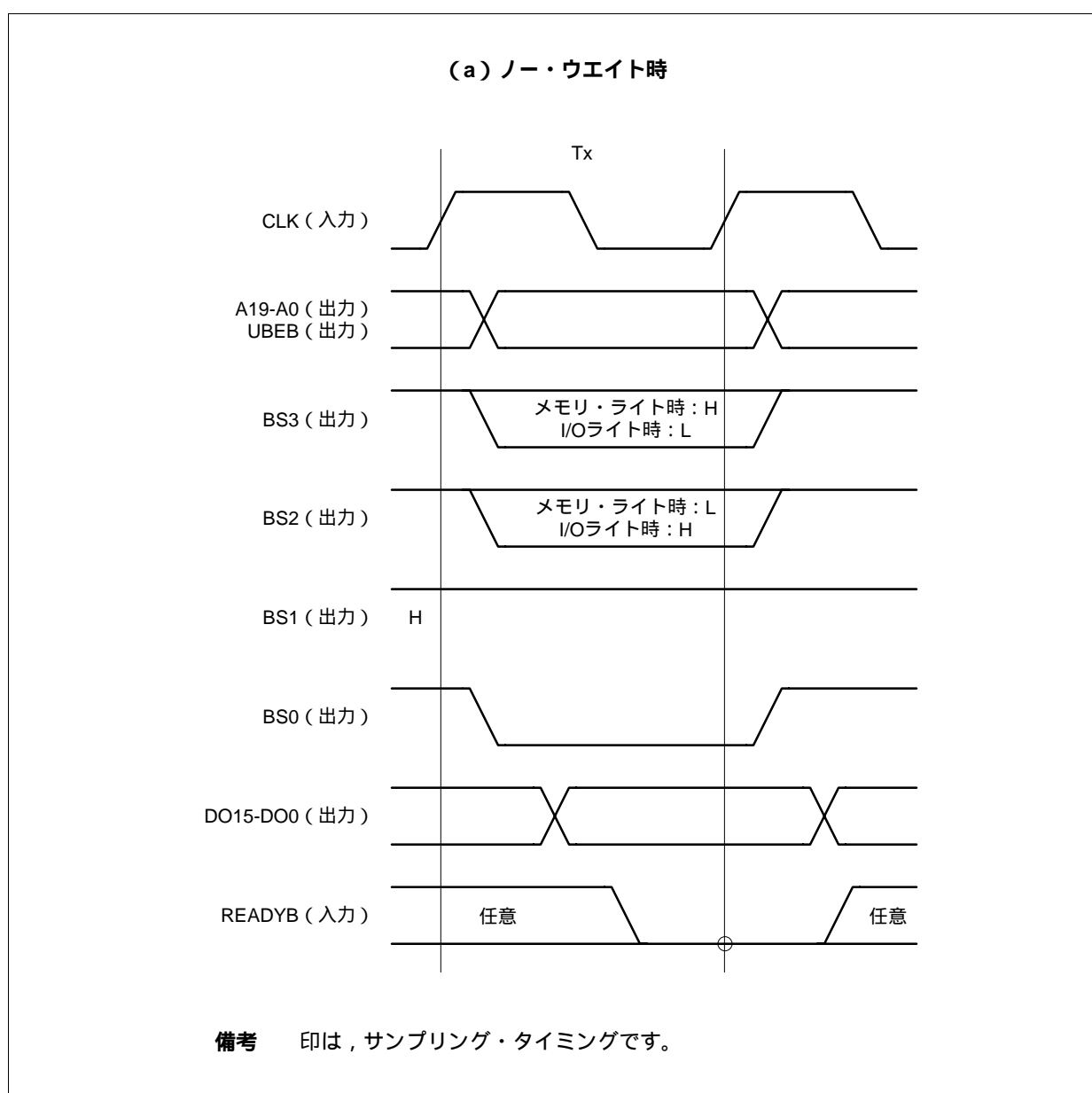
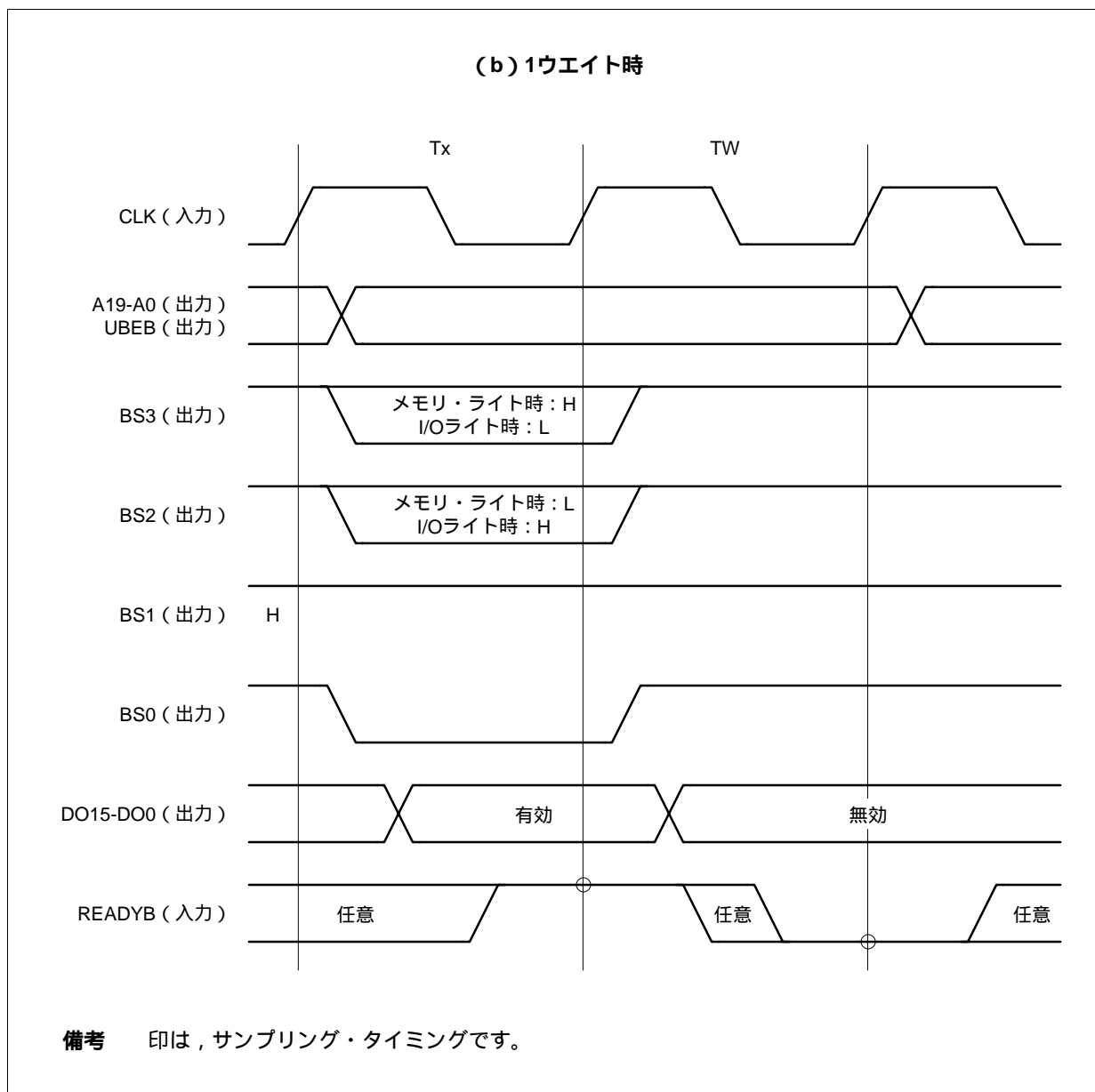


図4-3 メモリ、I/Oのライト・タイミング (2/2)



4.4 バス・ホールド機能

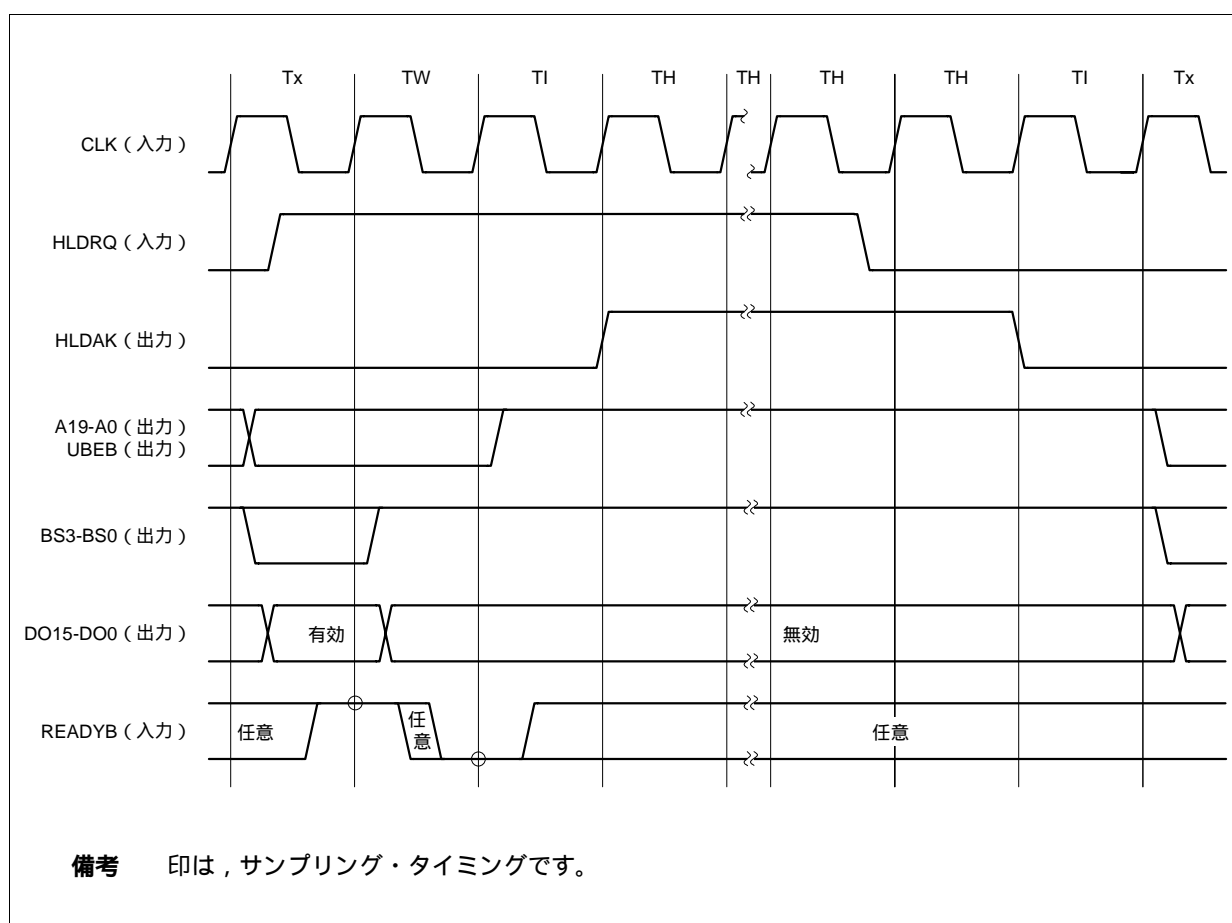
HLDQRQ 端子にハイ・レベルを入力すると、現在実行中のバス・サイクルの終了後に HLDARQ 信号がハイ・レベルになり、バス・ホールド状態 (TH) に移行します。ただし、TH ステートの直前には、1 クロック期間のアイドル・サイクル (TI) が必ず挿入されます。

TH ステートでは、A19-A0 信号、UBEB 信号、BS3-BS0 信号はハイ・レベルになりますが、DO15-DO0 信号は不定データになります。

続いて、HLDQRQ 端子にロウ・レベルを入力すると HLDARQ 信号がロウ・レベルになり、通常のパス・サイクルに戻ります。ただし、TH ステートの直後には、1 クロック期間の TI ステートが必ず挿入されます。

なお、TH ステートでは、コード・フェッチ・サイクルは、発生しません。

図4-4 バス・ホールド・タイミング



第5章 割り込み機能

割り込みには、ハードウェア割り込みとソフトウェア割り込みの2種類があります。

これらの割り込みは、すべて割り込みベクタ・テーブルを参照するベクタ割り込みになっています。

割り込みベクタ・テーブルは、割り込み処理ルーチンの開始アドレスを格納しています。

割り込みが発生すると、割り込み要因に対応したベクタ・テーブル中の固定の4バイト（固定ベクタ）、または、そのつど指定される任意の4バイト（可変ベクタ）を参照し、そこに格納されているアドレス（割り込み処理ルーチンの開始アドレス）に分岐します。

割り込みベクタ・テーブルはメモリ空間の000H-3FFHの1Kバイトの領域に割り当てられ、最大256のベクタを定義できます。

表5-1に各割り込み要因の処理クロック数、ベクタ番号、優先順位を示します。

また、図5-1に割り込みベクタ・テーブルの構成を示します。

表5-1 割り込み要因一覧

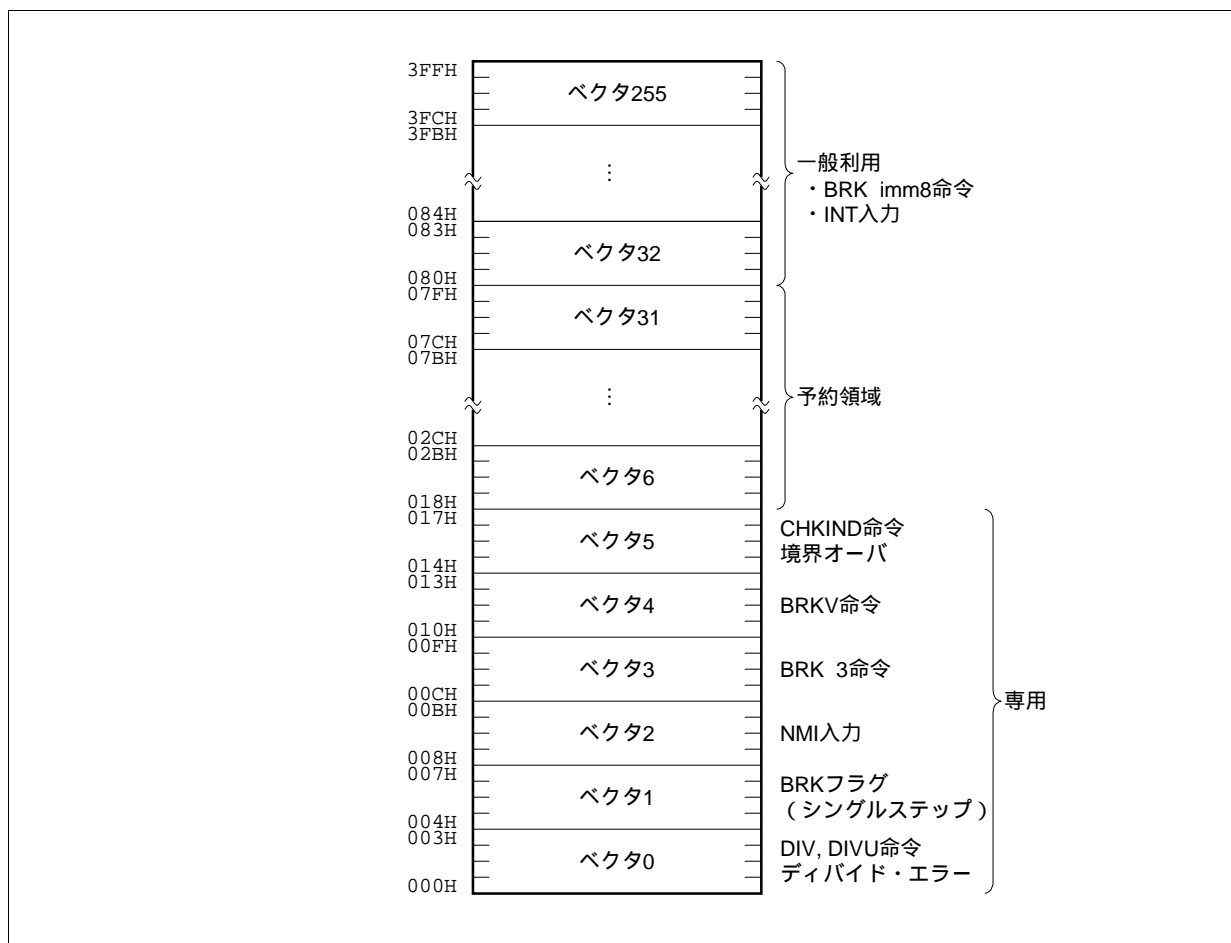
割り込み要因		処理クロック数 ^注	ベクタ番号	優先順位
ハードウェア 割り込み	NMI 入力（立ち上がりエッジ・アクティブ）	26	2	2
	INT 入力（ハイ・レベル・アクティブ）	32	32-255	3
ソフトウェア 割り込み	DIV, DIVU 命令のディバイド・エラー	25	0	1
	CHKIND 命令の境界オーバ	25	5	1
	BRKV 命令	25	4	1
	BRK 3 命令	25	3	1
	BRK imm8 命令	25	32-255	1
	BRK フラグ（シングルステップ）	25	1	4

注 割り込みにより命令実行が中断されてから、その割り込み処理ルーチンの先頭アドレスに分岐するまでのクロック数（実行中の命令のクロック数、バス・サイクルに対するウェイト・サイクルの挿入、バス・ホールド要求は考慮していません）。

備考 次の3つの命令は命令実行時間が比較的長い命令ですが、命令実行中に割り込み要求があっても途中で中断せずに、最後まで実行してから割り込み要求を受け付けます。割り込み応答時間が特に問題となるシステムでは注意してください。

命 令	実行クロック数	備 考
DIVU	24	DIVU 命令によるディバイド・エラーが発生しない場合
DIV	25	DIV 命令によるディバイド・エラーが発生しない場合
PREPARE	139	第2オペランドが31の場合

図5-1 割り込みベクタ・テーブルの構成



ベクタ0からベクタ5は使用する割り込み要因が指定されており、また、ベクタ6からベクタ31は予約領域のため、一般用途に使用できません。

ベクタ32からベクタ255は一般用途として、BRK imm8命令、INT入力を使用可能です。

1つの割り込みベクタは4バイトからなり、上位アドレスの2バイトがベース・アドレス・ポインタ（プログラム・セグメント値）としてプログラム・セグメント・レジスタ（PS）にロードされ、下位アドレスの2バイトがオフセット値としてプログラム・カウンタ（PC）にロードされます。

例 ベクタ0の場合

003H
002H
001H
000H

PS (003H, 002H)

PC (001H, 000H)

プログラム作成の際は上記の例に基づいて使用する各ベクタの内容を、プログラムの最初でイニシャライズしてください。

割り込み処理ルーチンにジャンプするときの基本ステップは次のとおりです。

TA ベクタ下位ワード・データ (オフセット値)
 TC ベクタ上位ワード・データ (プログラム・セグメント値)
 SP SP - 2, (SP + 1, SP) PSW
 IE 0, BRK 0, MD 1
 SP SP - 2, (SP + 1, SP) PS
 PS TC
 SP SP - 2, (SP + 1, SP) PC
 PC TA

注意 プログラム・ステータス・ワード (PSW) の割り込み許可フラグ (IE) とブレーク・フラグ (BRK) は割り込み処理が起動するとクリア (0) されるため、マスクブル割り込み (INT)、または BRK フラグ (シングルステップ) 割り込みは受け付けられなくなります。

5.1 ハードウェア割り込み

ハードウェア割り込みには次の2種類があります。

- ノンマスカブル割り込み (NMI)
- マスカブル割り込み (INT)

5.1.1 ノンマスカブル割り込み (NMI)

NMI は、マスク不可能な割り込みで、ソフトウェアによる禁止はできません。周辺デバイスから NMI 端子への入力があるといつでも受け付けられ、立ち上がりエッジで検出されます。

NMI は INT よりも高い優先順位を持ち、通常電源の急激な変動 (瞬時停電) やメモリ・エラー、バス・エラーなどの対応に使用します。

なお、NMI による割り込みアクノリッジ・サイクルは発行されず、割り込みアクノリッジも出力されません。

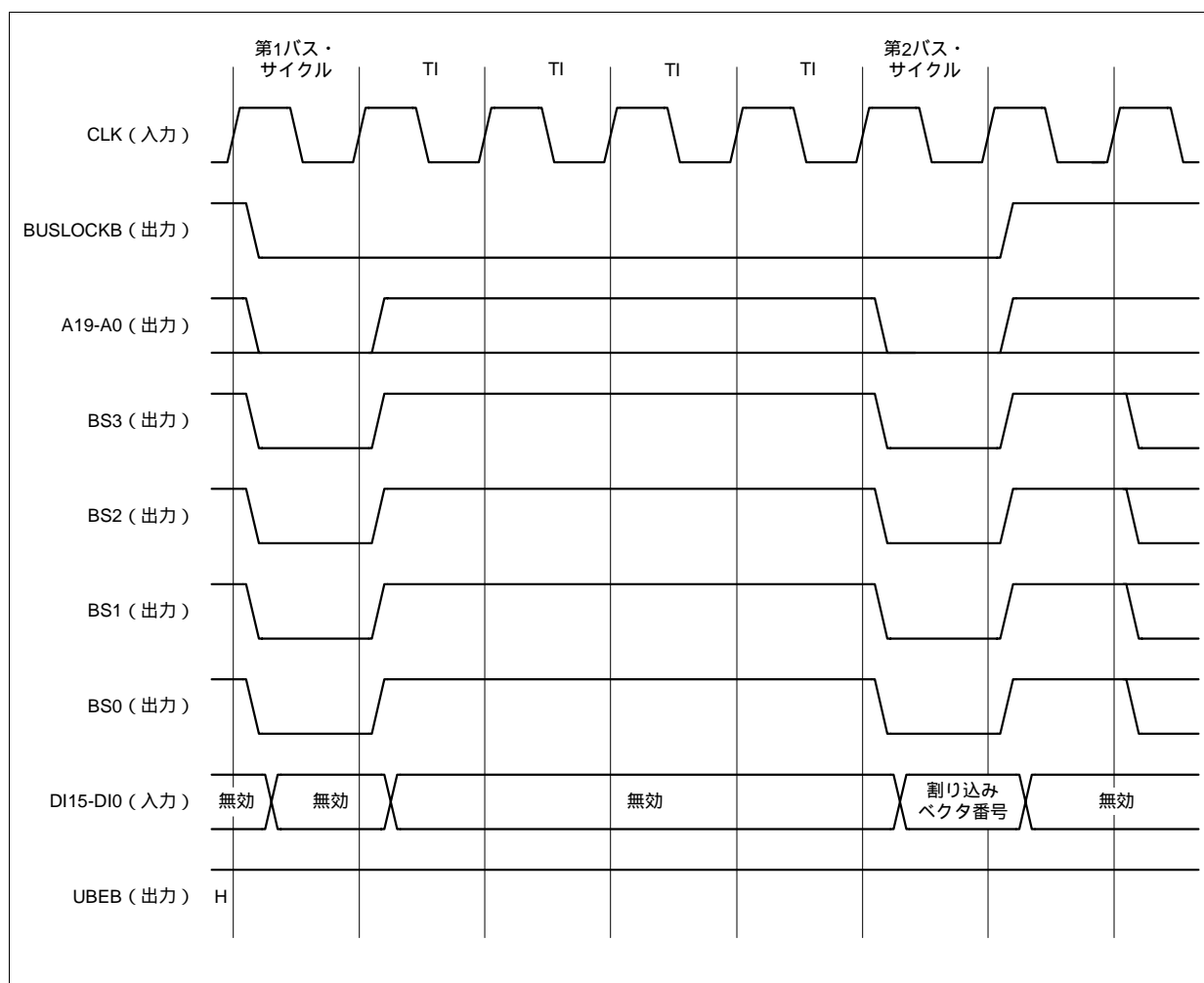
注意 リセット直後も NMI 要求は受け付けますが (INT 要求は受け付けません)、スタック・ポインタ (SS:SP) に正しい値をロードするまでは、正常な NMI 処理を実行できません。したがって、外部回路で NMI 入力をマスクするなどの対策を行ってください。

5.1.2 マスカブル割り込み (INT)

マスカブル割り込み (INT) は、次の応答シーケンスで受け付けられます。

なお、割り込みアクノリッジ信号は、BS3-BS0 信号をデコード (BS3-BS0 がすべてロウ・レベルのとき) して作成してください。

図5-2 割り込みアクノリッジ・サイクル



- (1) 第1バス・サイクルは、外部割り込みコントローラとの同期をとるために起動します。データ・バス (DI15-DI0) から読み込んだ値は使用しません。
このとき、アドレス・バス (A19-A0) には 00000H 番地が出力されますが、この値に意味はありません。また、READYB 信号を使用して、ウェイト・サイクルを挿入することができます。
- (2) 第1バス・サイクルと、第2バス・サイクルの間に、4クロックのアイドル・サイクル (TI) が挿入されます。この間、BUSLOCKB 信号は、ロウ・レベルのままです。なお、第1バス・サイクルと、第2バス・サイクルの間に、コード・フェッチ・サイクルが発生することはありません。
- (3) 第2バス・サイクルで、割り込みコントローラからベクタ番号を読み込みます (データ・バスの下位バイトだけが有効)。このとき、アドレス・バスには 00000H 番地が出力されますが、この値に意味はありません。
また、第1バス・サイクルと同様に READYB 信号を使用して、ウェイト・サイクルを挿入することができます。

備考 割り込みアクノリッジ・サイクルの期間、V30HL では \overline{UBE} 出力は常にロウ・レベルですが、V30MZ では UBEB 出力は常にハイ・レベルとなります (UBEB 出力がロウ・レベルになるのは、割り込み処理ルーチンの開始アドレスをリードするときだけです)。

5.2 ソフトウェア割り込み

ソフトウェア割り込みは BRK フラグ (シングルステップ) 割り込みを除き, ハードウェア割り込みより優先されます。次のように分類できます。

(1) 命令の結果による割り込み

- DIV 命令, DIVU 命令によるディバイド・エラー
- CHKIND 命令による境界オーバ検出

命令の処理結果が不正であったとき, 自動的に割り込み処理を行います。

(2) 条件付きブレーク (BRKV 命令の実行) による割り込み

BRKV 命令を実行するとき, V フラグがセット (1) されていると, 割り込み処理を行います。演算結果がオーバフローしたときの処理に使用します。

(3) 無条件ブレーク命令による割り込み

- 1 バイト・ブレーク命令 (BRK 3)
- 2 バイト・ブレーク命令 (BRK imm8 (3))

システム・コールやセグメント間コールで, 分岐先を意識しないでサブルーチンに分岐する場合に使用します。

(4) BRK フラグ (シングルステップ) 割り込み

プログラムのデバッグなどに有効な機能です。

この割り込みは, PSW の BRK フラグによって制御されます。ただし, BRK フラグを直接セット/クリアする命令はなく, PSW の内容をスタックに退避した状態で操作し, それを PSW に復帰することによって間接的にセット/クリア処理を行います。

BRK フラグがセット (1) されると, 次の 1 命令を実行したあと, ベクタ 1 で指定される割り込みルーチン (モニタ・プログラムなど) が起動され, そのとき IE フラグといっしょに BRK フラグもクリア (0) されます。

したがって, いったんベクタ 1 の割り込みが起動されると, 割り込みルーチンの命令は 1 つずつではなく, ほかの割り込みと同様に連続して実行されます。ここで内部レジスタやフラグの状態, メモリの内容などのチェックやダンプができます。

この割り込みルーチンの中で, シングルステップの回数などをチェックし, シングルステップ動作を終了してもよければ, メモリ操作命令でスタック内の BRK フラグをクリア (0) してリターン (復帰) します。こうするとメイン・ルーチンに戻ったあと, 連続して命令が実行されるようになります。

BRK フラグを操作せずに戻れば, スタックに退避していた BRK = 1 が PSW にリストアされ, メイン・ルーチンの 1 命令を実行したあと, 再びベクタ 1 の割り込みを起動します。

5.3 割り込みが受け付けられないタイミング

次の(1)から(4)で示されるタイミング、すなわち、セグメント・レジスタに直接データがセットされる命令または3種のプリフィクスと、それに続く1命令の間では、ハードウェア割り込み、およびBRKフラグ(シングルステップ)割り込みは受け付けられません。また、(5)で示されるタイミングではINT割り込みだけが受け付けられません。

割り込みが受け付けられないタイミングは、次の5つです。

- (1) 「MOV SS, reg16」, 「MOV SS, mem16」, 「POP SS」の各命令と次の命令の間
- (2) セグメント・オーバーライド・プリフィクス(「PS:」, 「SS:」, 「DS0:」, 「DS1:」)と次の命令の間
- (3) リピート・プリフィクス(REP, REPE, REPNE)と次の命令の間
- (4) BUSLOCK命令と次の命令の間
- (5) 「EI」, 「RETI」, 「POP PSW」の各命令(命令実行によりPSWレジスタのIEフラグがセット(1)される場合)と次の命令の間(INT割り込みのみ)。

ただし、(1)から(4)のタイミングで発生したNMIは、内部で保留され、続く1命令実行終了後に受け付けられます。

また、すでにIEフラグがセットされた状態で(5)のタイミングが発生した場合は、これらの命令の実行直後でもINT割り込みを受け付けます。

5.4 ブロック処理命令実行中の割り込み処理

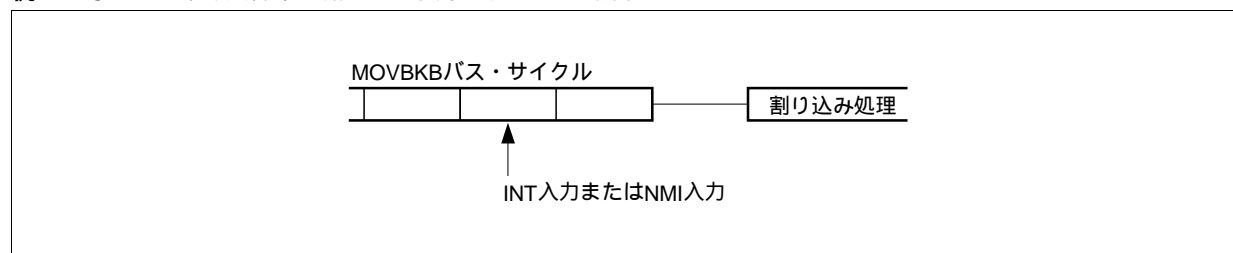
プリミティブ・ブロック転送/比較，入出力命令実行中にハードウェア割り込み要求が発生した場合，V30MZ はそれを受け付け，該当する割り込みアドレスに分岐します。

ただし，ブロック処理命令では，割り込み要求が発生したバス・サイクルの終了直後には，割り込み要求を受け付けられない場合があります。その場合，割り込み要求が発生してから，V30MZ がその割り込み要求を受け付けるまでに，数回のバス・サイクルを必要とします。表5-2に，そのバス・サイクル数を示します。この数は，割り込み要求が発生したバス・サイクルを1回目として数えています。

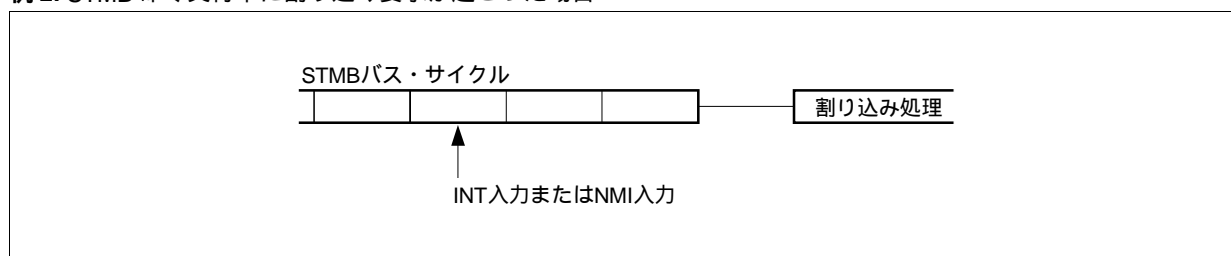
表5-2 割り込み要求を受け付けるまでに要するバス・サイクル数

命 令	IX レジスタ	IY レジスタ	割り込み要求を受け付けるまでに要する バス・サイクル数
MOVBKW	偶数	偶数	2-4
	偶数	奇数	3-6
	奇数	偶数	2-5
	奇数	奇数	3-7
MOVBKB	-	-	2-4
CMPBKW	偶数	偶数	1-2
	偶数	奇数	1-3
	奇数	偶数	1-3
	奇数	奇数	1-4
CMPBKB	-	-	1-2
CMPMW	-	偶数	1
	-	奇数	1-2
CMPMB	-	-	1
LDMW	偶数	-	1
	奇数	-	1-2
LDMB	-	-	1
STMW	-	偶数	3-4
	-	奇数	3-5
STMB	-	-	3-4

例 1. MOVBKB 命令実行中に割り込み要求が起こった場合



例2. STMB 命令実行中に割り込み要求が起こった場合



こうして起動された割り込み処理ルーチンの先頭で、ブロック・データのカウンタとして働いている CW レジスタの内容をスタックに退避しておき、割り込み処理ルーチンの終わりで、退避した内容を CW レジスタに復帰させたあと、RETI 命令で元のルーチンに戻れば、中断していたブロック処理を再開できます。

このとき、ブロック処理命令の前にプリフィクスが置かれていた場合、3種類までそれらを記憶し、割り込み処理ルーチンから戻ったときにプリフィクスの置かれているアドレスに戻れるように、リターン・アドレスを修正（1種類のプリフィクスに対して「-1」番地）して退避させるようになっています。

これらの機能を有効に利用するには、あるブロック処理命令の前に置かれているプリフィクスの合計を3個以下にしてください。

[メモ]

第6章 スタンバイ機能

6.1 スタンバイ・モードの設定

HALT 命令を実行することにより、スタンバイ (HALT) モードに設定されます。

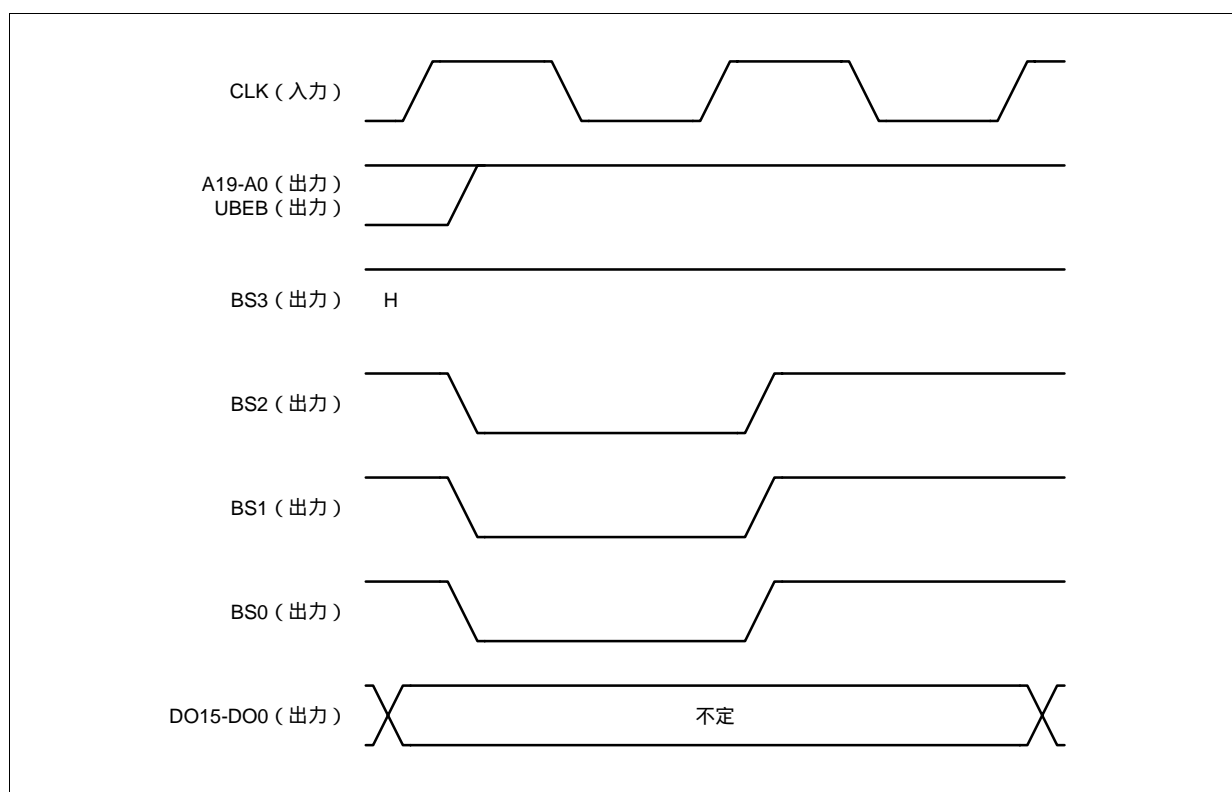
スタンバイ・モードでは、モードの解除に必要な機能とバス・ホールド機能に関する回路にだけクロックを供給し、そのほかの回路への供給を停止します。

この結果、システムの消費電力を大幅に削減できます。

6.2 スタンバイ・モード

スタンバイ・モードに移行する際、BS3-BS0 信号は 1 クロック期間、HALT ステータスを出力します。その後、A19-A0, UBEB, BS3-BS0 の各信号はハイ・レベルになります。なお、DO15-DO0 出力は不定となります。

図6-1 スタンバイ・モードへの移行のタイミング



スタンバイ・モード中もバス・ホールド機能は有効ですが、バス・ホールド・アクノリッジ期間が終了すると、再びスタンバイ・モードに戻ります。ただし、スタンバイ・モードに戻るときは、BS3-BS0 端子からの HALT ステータスの出力はありません (アイドル状態のままです)。

6.3 スタンバイ・モードの解除

スタンバイ・モードの解除には、大別してハードウェア割り込み要求（NMI 入力または INT 入力）による解除と RESET 入力による解除の 2 種類の方法があります。

なお、これらの入力が同時にアクティブになった場合は、通常の割り込み優先順位が適用されます。

6.3.1 ハードウェア割り込み要求による解除

注意 1. バス・ホールド要求とハードウェア割り込み要求が同時に行われた場合は、バス・ホールド要求が優先され、バス・ホールド・サイクル終了後にハードウェア割り込み要求によるスタンバイ・モードの解除が行われます。

2. HALT 命令の実行とハードウェア割り込み要求が同時に行われた場合は、BS3-BS0 端子から HALT ステータスが出力されますが、スタンバイ・モードへは移行せず、ただちに割り込み処理（または HALT 命令の次の命令）を実行します。

(1) NMI 入力による解除

NMI 入力の立ち上がりエッジを検出すると、スタンバイ・モードを解除し、割り込み処理（NMI ルーチン）を開始します。続いて、NMI ルーチンの終わりで RETI 命令が実行されると、HALT 命令の次の命令からプログラムが再開されます。

(2) INT 入力による解除

割り込み許可状態（PSW の IE フラグ = 1）か、割り込み禁止状態（PSW の IE フラグ = 0）かによって解除後の動作が異なります。

(a) 割り込み許可状態の場合

INT 入力があると、スタンバイ・モードを解除し、割り込み処理（INT ルーチン）を開始します。続いて、INT ルーチンの終わりで RETI 命令が実行されると、HALT 命令の次の命令からプログラムが再開されます。

注意 割り込みアクノリッジ・サイクルの第 1 バス・サイクルが起動されるまで、INT 端子にハイ・レベルを入力してください。

(b) 割り込み禁止状態の場合

INT 入力があると、スタンバイ・モードを解除し、HALT 命令の次の命令からプログラムが再開されます。

注意 1 クロック期間以上、INT 端子にハイ・レベルを入力してください。

6.3.2 RESET 入力による解除

スタンバイ・モード中に RESET 信号が入力されると、無条件にスタンバイ・モードは解除され、通常のリセット動作に入ります。したがって、スタンバイ・モード中に保持されていた状態は無効となり、スタンバイ・モードにより停止されていたプログラムは再開できません。

第7章 リセット機能

RESET 端子に4クロック以上のハイ・レベルが入力されると V30MZ の各出力端子は表7-1 に示す状態に変化します。ハイ・レベルが入力されている間は、この状態を保ちます。

なお、リセット入力、CLK 端子の立ち上がりエッジに対するセットアップ（設定）時間とホールド（保持）時間が規定されています。必ずこれらの規定時間を満足するようにリセット入力を行ってください。

表7-1 リセット時の出力端子状態

端 子	リセット時の状態
A19-A0	ハイ・レベル出力
DO15-DO0	不定
UBEB	ハイ・レベル出力
BUSLOCKB	
BS3-BS0	
HLDK	ロウ・レベル出力
TBO42-TBO0	ハイ・インピーダンス

また、各レジスタはリセット後、表7-2 に示す値に初期化されます。

表7-2 リセット後の各レジスタの初期値

対 象	初期値																																				
PC	0000H																																				
PFP																																					
PS	FFFFH																																				
SS, DS0, DS1	0000H																																				
AW, BW, CW, DW	不定																																				
SP, BP																																					
IX, IY																																					
PSW	<table border="1"> <tr> <td></td> <td>MD</td> <td></td> <td></td> <td></td> <td>V</td> <td>DIR</td> <td>IE</td> <td>BRK</td> </tr> <tr> <td>上位</td> <td>1</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td></td> <td>S</td> <td>Z</td> <td></td> <td>AC</td> <td></td> <td>P</td> <td></td> <td>CY</td> </tr> <tr> <td>下位</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>1</td> <td>0</td> </tr> </table>		MD				V	DIR	IE	BRK	上位	1	1	1	1	0	0	0	0		S	Z		AC		P		CY	下位	0	0	0	0	0	0	1	0
	MD				V	DIR	IE	BRK																													
上位	1	1	1	1	0	0	0	0																													
	S	Z		AC		P		CY																													
下位	0	0	0	0	0	0	1	0																													

RESET 端子に入力される信号がロウ・レベルに戻ると、V30MZ はメモリの FFFF0H（セグメント値：FFFFH，オフセット値：0000H）番地より命令のプリフェッチを開始します。

[メモ]

第8章 テスト機能

V30MZ は、他の CBIC コアと同様に、テスト・バスによる単体テスト機能を持っています。

8.1 テスト端子

V30MZ は、次のテスト用端子を持っています。

- TBI22-TBI0
- TBO42-TBO0
- BUNRI
- TEST

8.1.1 テスト・バス端子 (TBI22-TBI0, TBO42-TBO0)

テスト・バス端子は単体テスト・モード時に、ノーマル端子の代わりに使用します。

詳細は、各セルベース IC ファミリの **ユーザーズ・マニュアル 設計編** を参照してください。

8.1.2 BUNRI, TEST 端子

ノーマル/単体テスト/スタンバイ・テスト・モードの選択に使用します。

表8-1 テスト・モードの設定一覧

BUNRI 端子入力レベル	TEST 端子入力レベル	モード
ロウ・レベル	任意	ノーマル・モード
ハイ・レベル	ロウ・レベル	スタンバイ・テスト・モード
ハイ・レベル	ハイ・レベル	単体テスト・モード

8.2 ノーマル・モード

ユーザが通常、使用するモードです。

BUNRI 端子にロウ・レベルが入力されている場合、テスト用端子以外の端子が有効になり、ノーマル・モードになります。このとき、TBI22-TBI0 端子への入力は無視され、TBO42-TBO0 端子はハイ・インピーダンスになります。

8.3 単体テスト・モードとスタンバイ・テスト・モード

BUNRI 端子にハイ・レベルが入力されている場合、テスト用端子以外の端子への入力は無視され（無効）、テスト・モードになります。テスト・モードには単体テスト・モードとスタンバイ・テスト・モードの2つがあります。

単体テスト・モード、スタンバイ・テスト・モード中に、バス構成の端子（テスト用端子を除く）がフローティングやバス・ファイトを起こさないような回路設計を行ってください（各モードでの端子状態については、**2.2 端子状態**を参照してください）。

8.3.1 単体テスト・モード

BUNRI 端子と TEST 端子にハイ・レベルが入力されている場合、単体テスト・モードになります。単体テスト・モードでは、テスト用端子以外の端子からの入力は無視され、代わりに TBI22-TBI0 端子からの入力が有効となります。また、TBO42-TBO0 端子にテスト用端子以外の各端子に対応する値が出力されます。

注意 単体テスト・モードは、NEC 側でテストを行うためのモードです。

8.3.2 スタンバイ・テスト・モード

BUNRI 端子にハイ・レベルが入力され、TEST 端子にロウ・レベルが入力されている場合、スタンバイ・テスト・モードになります。

このモードは、テスト回路確認シミュレーション、ユーザ・ロジック分離シミュレーションのときに、テストしないコアに対して使用します。

TBI22-TBI0 端子への入力は無視され、TBO42-TBO0 端子はハイ・インピーダンスになります。

付録 A 命令実行クロック数一覧

この章では、次の(1)から(7)に示す条件での各命令の実行クロック数を示します。

各命令の機能の詳細については、16ビットVシリーズ ユーザーズ・マニュアル 命令編を参照してください。

- (1) 命令はデコードが完了している。
- (2) メモリ・アクセスやI/Oアクセスにおいて、ウエイト・ステートが発生しない。^{注1}
- (3) バス・ホールド要求がない。
- (4) ワード・データは、偶数アドレスに配置されている。^{注2}
- (5) 実効アドレス計算に必要なレジスタ(BW, BP, SP, IX, IYなど)が、直前の命令で変更されていない。^{注3}
- (6) 実効アドレス計算に必要なレジスタは1つだけである。^{注4}
- (7) 分岐命令の分岐先は偶数アドレスである。^{注5}

注1. ウエイト・ステートが発生する場合は、ウエイト・ステートのクロック数を命令実行クロック数に加算してください。

2. 奇数アドレスに配置されたワード・データをアクセスする場合は、1クロックを加算してください。

3. 直前の命令で変更したレジスタを実効アドレス計算に使用する場合は、1クロックを加算してください(ただし、LDEA命令とリピート・プリフィクス命令では増加しません)。

また、連続したPUSH, POP命令では、実行クロック数は増加しません(すべて1クロックで実行します)。

4. 実効アドレス計算に必要なレジスタが2つの場合(例 MOV AW, [BW+IX])、1クロックを加算してください。

5. 奇数アドレスに分岐する場合は、1クロックを加算してください。

表A - 1 命令実行クロック数一覧 (1/11)

二モニック,オペランド	オペレーション・コード														クロック数		
	7	6	5	4	3	2	1	0	7	6	5	4	3	2		1	0
ADD acc, imm	0	0	0	0	0	1	0	W	imm8 or imm16-low							1	
	imm16-high							-									
ADD mem, imm	1	0	0	0	0	0	s	W	mod	0	0	0	mem			3	
	(disp-low)							(disp-high)									
	imm8 or imm16-low							imm16-high									
ADD mem, reg	0	0	0	0	0	0	0	W	mod	reg			mem			3	
	(disp-low)							(disp-high)									
ADD reg, imm	1	0	0	0	0	0	s	W	1	1	0	0	0	reg		1	
	imm8 or imm16-low							imm16-high									
ADD reg, mem	0	0	0	0	0	0	1	W	mod	reg			mem			2	
	(disp-low)							(disp-high)									
ADD reg, reg'	0	0	0	0	0	0	1	W	1	1	reg			reg'			1
ADD reg, reg'	0	0	0	0	0	0	0	W	1	1	reg			reg'			1
ADDC acc, imm	0	0	0	1	0	1	0	W	imm8 or imm16-low							1	
	imm16-high							-									
ADDC mem, imm	1	0	0	0	0	0	s	W	mod	0	1	0	mem			3	
	(disp-low)							(disp-high)									
	imm8 or imm16-low							imm16-high									
ADDC mem, reg	0	0	0	1	0	0	0	W	mod	reg			mem			3	
	(disp-low)							(disp-high)									
ADDC reg, imm	1	0	0	0	0	0	s	W	1	1	0	1	0	reg		1	
	imm8 or imm16-low							imm16-high									
ADDC reg, mem	0	0	0	1	0	0	1	W	mod	reg			mem			2	
	(disp-low)							(disp-high)									
ADDC reg, reg'	0	0	0	1	0	0	1	W	1	1	reg			reg'			1
ADDC reg, reg'	0	0	0	1	0	0	0	W	1	1	reg			reg'			1
ADJ4A	0	0	1	0	0	1	1	1	-							10	
ADJ4S	0	0	1	0	1	1	1	1	-							10	
ADJBA	0	0	1	1	0	1	1	1	-							9	
ADJBS	0	0	1	1	1	1	1	1	-							9	
AND acc, imm	0	0	1	0	0	1	0	W	imm8 or imm16-low							1	
	imm16-high							-									
AND mem, imm	1	0	0	0	0	0	0	W	mod	1	0	0	mem			3	
	(disp-low)							(disp-high)									
	imm8 or imm16-low							imm16-high									
AND mem, reg	0	0	1	0	0	0	0	W	mod	reg			mem			3	
	(disp-low)							(disp-high)									
AND reg, imm	1	0	0	0	0	0	0	W	1	1	1	0	0	reg		1	
	imm8 or imm16-low							imm16-high									
AND reg, mem	0	0	1	0	0	0	1	W	mod	reg			mem			2	
	(disp-low)							(disp-high)									

表A - 1 命令実行クロック数一覧 (2/11)

二モニック,オペランド	オペレーション・コード																クロック数		
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0			
AND reg, reg'	0	0	1	0	0	0	1	W	1	1							reg	reg'	1
AND reg, reg'	0	0	1	0	0	0	0	W	1	1							reg	reg'	1
BC short-label	0	1	1	1	0	0	1	0									disp8		1 (CY=0のとき)
BC short-label	0	1	1	1	0	0	1	0									disp8		4 (CY=1のとき)
BCWZ short-label	1	1	1	0	0	0	1	1									disp8		1 (CW=0のとき)
BCWZ short-label	1	1	1	0	0	0	1	1									disp8		4 (CW=0のとき)
BE short-label	0	1	1	1	0	1	0	0									disp8		1 (Z=0のとき)
BE short-label	0	1	1	1	0	1	0	0									disp8		4 (Z=1のとき)
BGE short-label	0	1	1	1	1	1	0	1									disp8		1 (S V=1のとき)
BGE short-label	0	1	1	1	1	1	0	1									disp8		4 (S V=0のとき)
BGT short-label	0	1	1	1	1	1	1	1									disp8		1 ((S V) Z)=1のとき)
BGT short-label	0	1	1	1	1	1	1	1									disp8		4 ((S V) Z)=0のとき)
BH short-label	0	1	1	1	0	1	1	1									disp8		1 (CY Z=1のとき)
BH short-label	0	1	1	1	0	1	1	1									disp8		4 (CY Z=0のとき)
BL short-label	0	1	1	1	0	0	1	0									disp8		1 (CY=0のとき)
BL short-label	0	1	1	1	0	0	1	0									disp8		4 (CY=1のとき)
BLE short-label	0	1	1	1	1	1	1	0									disp8		1 ((S V) Z)=0のとき)
BLE short-label	0	1	1	1	1	1	1	0									disp8		4 ((S V) Z)=1のとき)
BLT short-label	0	1	1	1	1	1	0	0									disp8		1 (S V=0のとき)
BLT short-label	0	1	1	1	1	1	0	0									disp8		4 (S V=1のとき)
BN short-label	0	1	1	1	1	0	0	0									disp8		1 (S=0のとき)
BN short-label	0	1	1	1	1	0	0	0									disp8		4 (S=1のとき)
BNC short-label	0	1	1	1	0	0	1	1									disp8		1 (CY=1のとき)
BNC short-label	0	1	1	1	0	0	1	1									disp8		4 (CY=0のとき)
BNE short-label	0	1	1	1	0	1	0	1									disp8		1 (Z=1のとき)
BNE short-label	0	1	1	1	0	1	0	1									disp8		4 (Z=0のとき)
BNH short-label	0	1	1	1	0	1	1	0									disp8		1 (CY Z=0のとき)
BNH short-label	0	1	1	1	0	1	1	0									disp8		4 (CY Z=1のとき)
BNL short-label	0	1	1	1	0	0	1	1									disp8		1 (CY=1のとき)
BNL short-label	0	1	1	1	0	0	1	1									disp8		4 (CY=0のとき)
BNV short-label	0	1	1	1	0	0	0	1									disp8		1 (V=1のとき)
BNV short-label	0	1	1	1	0	0	0	1									disp8		4 (V=0のとき)
BNZ short-label	0	1	1	1	0	1	0	1									disp8		1 (Z=1のとき)
BNZ short-label	0	1	1	1	0	1	0	1									disp8		4 (Z=0のとき)
BP short-label	0	1	1	1	1	0	0	1									disp8		1 (S=1のとき)
BP short-label	0	1	1	1	1	0	0	1									disp8		4 (S=0のとき)
BPE short-label	0	1	1	1	1	0	1	0									disp8		1 (P=0のとき)
BPE short-label	0	1	1	1	1	0	1	0									disp8		4 (P=1のとき)
BPO short-label	0	1	1	1	1	0	1	1									disp8		1 (P=1のとき)
BPO short-label	0	1	1	1	1	0	1	1									disp8		4 (P=0のとき)

表A - 1 命令実行クロック数一覧 (3/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
BR far-label	1	1	1	0	1	0	1	0	offset-low								7
	offset-high								seg-low								
	seg-high								-								
BR memptr16	1	1	1	1	1	1	1	1	mod	1	0	0	mem			5	
	(disp-low)								(disp-high)								
BR memptr32	1	1	1	1	1	1	1	1	mod	1	0	1	mem			10	
	(disp-low)								(disp-high)								
BR near-label	1	1	1	0	1	0	0	1	disp-low								4
	disp-high								-								
BR regptr16	1	1	1	1	1	1	1	1	1	1	1	0	0	reg			4
BR short-label	1	1	1	0	1	0	1	1	disp8								4
BRK 3	1	1	0	0	1	1	0	0	-								9
BRK imm8 (3)	1	1	0	0	1	1	0	1	imm8								10
BRKV	1	1	0	0	1	1	1	0	-								6 (V = 0 のとき)
BRKV	1	1	0	0	1	1	1	0	-								13 (V = 1 のとき)
BUSLOCK	1	1	1	1	0	0	0	0	-								1
BV short-label	0	1	1	1	0	0	0	0	disp8								1 (V = 0 のとき)
BV short-label	0	1	1	1	0	0	0	0	disp8								4 (V = 1 のとき)
BZ short-label	0	1	1	1	0	1	0	0	disp8								1 (Z = 0 のとき)
BZ short-label	0	1	1	1	0	1	0	0	disp8								4 (Z = 1 のとき)
CALL far-proc	1	0	0	1	1	0	1	0	offset-low								10
	offset-high								seg-low								
	seg-high								-								
CALL memptr16	1	1	1	1	1	1	1	1	mod	0	1	0	mem			6	
	(disp-low)								(disp-high)								
CALL memptr32	1	1	1	1	1	1	1	1	mod	0	1	1	mem			12	
	(disp-low)								(disp-high)								
CALL near-proc	1	1	1	0	1	0	0	0	disp-low								5
	disp-high								-								
CALL regptr16	1	1	1	1	1	1	1	1	1	1	0	1	0	reg			5
CHKIND reg16, mem32	0	1	1	0	0	0	1	0	mod	reg			mem			13 (割り込み条件不成立時)	
	(disp-low)								(disp-high)								
CHKIND reg16, mem32	0	1	1	0	0	0	1	0	mod	reg			mem			20 (割り込み条件成立時)	
	(disp-low)								(disp-high)								
CLR1 CY	1	1	1	1	1	0	0	0	-								4
CLR1 DIR	1	1	1	1	1	1	0	0	-								4
CMP acc, imm	0	0	1	1	1	1	0	W	imm8 or imm16-low								1
	imm16-high								-								
CMP mem, imm	1	0	0	0	0	0	s	W	mod	1	1	1	mem			2	
	(disp-low)								(disp-high)								
	imm8 or imm16-low								imm16-high								

表A - 1 命令実行クロック数一覧 (4/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
CMP mem, reg	0	0	1	1	1	0	0	W	mod	reg			mem			2	
	(disp-low)						(disp-high)										
CMP reg, imm	1	0	0	0	0	0	s	W	1	1	1	1	1	reg			1
	imm8 or imm16-low						imm16-high										
CMP reg, mem	0	0	1	1	1	0	1	W	mod	reg			mem			2	
	(disp-low)						(disp-high)										
CMP reg, reg'	0	0	1	1	1	0	1	W	1	1	reg			reg'			1
CMP reg, reg'	0	0	1	1	1	0	0	W	1	1	reg			reg'			1
CMPBK [DS1-spec:] dst-block	1	0	1	0	0	1	1	W	-						6		
CMPBK [Seg-spec:] src-block, [DS1-spec:] dst-block	1	0	1	0	0	1	1	W	-						6		
CMPBKB	1	0	1	0	0	1	1	W	-						6		
CMPBKW	1	0	1	0	0	1	1	W	-						6		
CMPM [DS1-spec:] dst-block	1	0	1	0	1	1	1	W	-						4		
CMPMB	1	0	1	0	1	1	1	W	-						4		
CMPMW	1	0	1	0	1	1	1	W	-						4		
CVTBD	1	1	0	1	0	1	0	0	0	0	0	0	1	0	1	0	17
CVTBW	1	0	0	1	1	0	0	0	-						1		
CVTDB	1	1	0	1	0	1	0	1	0	0	0	0	1	0	1	0	6
CVTWL	1	0	0	1	1	0	0	1	-						1		
DBNZ short-label	1	1	1	0	0	0	1	0	disp8						2 (CW = 0のとき)		
DBNZ short-label	1	1	1	0	0	0	1	0	disp8						5 (CW = 0のとき)		
DBNZE short-label	1	1	1	0	0	0	0	1	disp8						6 (CW = 0でZ = 1のとき)		
DBNZE short-label	1	1	1	0	0	0	0	1	disp8						3 (上記以外のとき)		
DBNZNE short-label	1	1	1	0	0	0	0	0	disp8						6 (CW = 0でZ = 0のとき)		
DBNZNE short-label	1	1	1	0	0	0	0	0	disp8						3 (上記以外のとき)		
DEC mem	1	1	1	1	1	1	1	W	mod	0	0	1	mem			3	
	(disp-low)						(disp-high)										
DEC reg16	0	1	0	0	1	reg			-						1		
DEC reg8	1	1	1	1	1	1	1	W	1	1	0	0	1	reg			1
DI	1	1	1	1	1	0	1	0	-						4		
DISPOSE	1	1	0	0	1	0	0	1	-						2		
DIV mem16	1	1	1	1	0	1	1	W	mod	1	1	1	mem			25	
	(disp-low)						(disp-high)										
DIV mem8	1	1	1	1	0	1	1	W	mod	1	1	1	mem			18	
	(disp-low)						(disp-high)										
DIV reg16	1	1	1	1	0	1	1	W	1	1	1	1	1	reg			24
DIV reg8	1	1	1	1	0	1	1	W	1	1	1	1	1	reg			17
DIVU mem16	1	1	1	1	0	1	1	W	mod	1	1	0	mem			24	
	(disp-low)						(disp-high)										
DIVU mem8	1	1	1	1	0	1	1	W	mod	1	1	0	mem			16	
	(disp-low)						(disp-high)										

表A - 1 命令実行クロック数一覧 (5/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
DIVU reg16	1	1	1	1	0	1	1	W	1	1	1	1	0		reg	23	
DIVU reg8	1	1	1	1	0	1	1	W	1	1	1	1	0		reg	15	
DS0:	0	0	1	1	1	1	1	0								-	1
DS1:	0	0	1	0	0	1	1	0								-	1
EI	1	1	1	1	1	0	1	1								-	4
FPO1 fp-op, mem	1	1	0	1	1	X	X	X	mod	Y	Y	Y			mem	1	
	(disp-low)								(disp-high)								
HALT	1	1	1	1	0	1	0	0								-	9
IN acc, DW	1	1	1	0	1	1	0	W								-	6
IN acc, imm8	1	1	1	0	0	1	0	W							imm8	6	
INC mem	1	1	1	1	1	1	1	W	mod	0	0	0			mem	3	
	(disp-low)								(disp-high)								
INC reg16	0	1	0	0	0	reg										-	1
INC reg8	1	1	1	1	1	1	1	W	1	1	0	0	0		reg	1	
INM [DS1-spec:] dst-block, DW	0	1	1	0	1	1	0	W								-	6
LDEA reg16, mem16	1	0	0	0	1	1	0	1	mod		reg				mem	1	
	(disp-low)								(disp-high)								
LDM [Seg-spec:] src-block	1	0	1	0	1	1	0	W								-	3
LDMB	1	0	1	0	1	1	0	W								-	3
LDMW	1	0	1	0	1	1	0	W								-	3
MOV acc, dmem	1	0	1	0	0	0	0	W							addr-low	1	
	addr-high								-								
MOV AH, PSW	1	0	0	1	1	1	1	1								-	2
MOV dmem, acc	1	0	1	0	0	0	1	W							addr-low	1	
	addr-high								-								
MOV DS0, reg16, mem32	1	1	0	0	0	1	0	1	mod		reg				mem	6	
	(disp-low)								(disp-high)								
MOV DS1, reg16, mem32	1	1	0	0	0	1	0	0	mod		reg				mem	6	
	(disp-low)								(disp-high)								
MOV mem, imm	1	1	0	0	0	1	1	W	mod	0	0	0			mem	1	
	(disp-low)								(disp-high)								
MOV mem, reg	1	0	0	0	1	0	0	W	mod		reg				mem	1	
	(disp-low)								(disp-high)								
MOV mem16, sreg	1	0	0	0	1	1	0	0	mod	0	sreg				mem	3	
	(disp-low)								(disp-high)								
MOV PSW, AH	1	0	0	1	1	1	1	0								-	4
MOV reg, imm	1	0	1	1	W	reg									imm8 or imm16-low	1	
	imm16-high								-								
MOV reg, imm	1	1	0	0	0	1	1	W	1	1	0	0	0		reg	1	
	imm8 or imm16-low								imm16-high								

表A - 1 命令実行クロック数一覧 (6/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
MOV reg, mem	1	0	0	0	1	0	1	W	mod	reg			mem			1	
	(disp-low)								(disp-high)								
MOV reg, reg'	1	0	0	0	1	0	1	W	1	1	reg			reg'			1
MOV reg, reg'	1	0	0	0	1	0	0	W	1	1	reg			reg'			1
MOV reg16, sreg	1	0	0	0	1	1	0	0	1	1	0	sreg		reg		1	
MOV sreg, mem16	1	0	0	0	1	1	1	0	mod	0	sreg		mem			3	
	(disp-low)								(disp-high)								
MOV sreg, reg16	1	0	0	0	1	1	1	0	1	1	0	sreg		reg		2	
MOVBK [DS1-spec:] dst-block, [Seg-spec:] src-block	1	0	1	0	0	1	0	W	-								5
MOVBK [Seg-spec:] src-block	1	0	1	0	0	1	0	W	-								5
MOVBKB	1	0	1	0	0	1	0	W	-								5
MOVBKW	1	0	1	0	0	1	0	W	-								5
MUL mem16	1	1	1	1	0	1	1	W	mod	1	0	1	mem			4	
	(disp-low)								(disp-high)								
MUL mem8	1	1	1	1	0	1	1	W	mod	1	0	1	mem			4	
	(disp-low)								(disp-high)								
MUL reg16	1	1	1	1	0	1	1	W	1	1	1	0	1	reg		3	
MUL reg16, imm16	0	1	1	0	1	0	s	1	1	1	reg		reg'			3	
	imm16-low								imm16-high								
MUL reg16, imm8	0	1	1	0	1	0	s	1	1	1	reg		reg'			3	
	imm8								-								
MUL reg16, mem16, imm16	0	1	1	0	1	0	s	1	mod	reg		mem			4		
	(disp-low)								(disp-high)								
	imm16-low								imm16-high								
MUL reg16, mem16, imm8	0	1	1	0	1	0	s	1	mod	reg		mem			4		
	(disp-low)								(disp-high)								
	imm8								-								
MUL reg16, reg16', imm16	0	1	1	0	1	0	s	1	1	1	reg		reg'			3	
	imm16-low								imm16-high								
MUL reg16, reg16', imm8	0	1	1	0	1	0	s	1	1	1	reg		reg'			3	
	imm8								-								
MUL reg8	1	1	1	1	0	1	1	W	1	1	1	0	1	reg		3	
MULU mem16	1	1	1	1	0	1	1	W	mod	1	0	0	mem			4	
	(disp-low)								(disp-high)								
MULU mem8	1	1	1	1	0	1	1	W	mod	1	0	0	mem			4	
	(disp-low)								(disp-high)								
MULU reg16	1	1	1	1	0	1	1	W	1	1	1	0	0	reg		3	
MULU reg8	1	1	1	1	0	1	1	W	1	1	1	0	0	reg		3	
NEG mem	1	1	1	1	0	1	1	W	mod	0	1	1	mem			3	
	(disp-low)								(disp-high)								
NEG reg	1	1	1	1	0	1	1	W	1	1	0	1	1	reg		1	

表A - 1 命令実行クロック数一覧 (7/11)

二モニック,オペランド	オペレーション・コード																クロック数	
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
NOP	1	0	0	1	0	0	0	0	-								1	
NOT mem	1	1	1	1	0	1	1	W	mod	0	1	0	mem				3	
	(disp-low)								(disp-high)									
NOT reg	1	1	1	1	0	1	1	W	1	1	0	1	0	reg			1	
NOT1 CY	1	1	1	1	0	1	0	1	-								4	
OR acc, imm	0	0	0	0	1	1	0	W	imm8 or imm16-low								1	
	imm16-high								-									
OR mem, imm	1	0	0	0	0	0	0	W	mod	0	0	1	mem			3		
	(disp-low)								(disp-high)									
	imm8 or imm16-low								imm16-high									
OR mem, reg	0	0	0	0	1	0	0	W	mod	reg			mem			3		
	(disp-low)								(disp-high)									
OR reg, imm	1	0	0	0	0	0	0	W	1	1	0	0	1	reg			1	
	imm8 or imm16-low								imm16-high									
OR reg, mem	0	0	0	0	1	0	1	W	mod	reg			mem			2		
	(disp-low)								(disp-high)									
OR reg, reg'	0	0	0	0	1	0	1	W	1	1	reg			reg'			1	
OR reg, reg'	0	0	0	0	1	0	0	W	1	1	reg			reg'			1	
OUT DW, acc	1	1	1	0	1	1	1	W	-								6	
OUT imm8, acc	1	1	1	0	0	1	1	W	imm8								6	
OUTM DW, [Seg-spec:] src-block	0	1	1	0	1	1	1	W	-								7	
POLL	1	0	0	1	1	0	1	1	-								9+9×POLLB端子サブリンク数	
POP mem16	1	0	0	0	1	1	1	1	mod	0	0	0	mem			3		
	(disp-low)								(disp-high)									
POP PSW	1	0	0	1	1	1	0	1	-								3	
POP R	0	1	1	0	0	0	0	1	-								8	
POP reg16	0	1	0	1	1	reg			-								1	
POP reg16	1	0	0	0	1	1	1	1	1	1	0	0	0	reg			1	
POP sreg	0	0	0	sreg			1	1	1	-								3
PREPARE imm16, imm8 (imm8 = 0のとき)	1	1	0	0	1	0	0	0	imm16-low								8	
	imm16-high								imm8									
PREPARE imm16, imm8 (imm8 = 1のとき)	1	1	0	0	1	0	0	0	imm16-low								14	
	imm16-high								imm8									
PREPARE imm16, imm8 (imm8>1のとき)	1	1	0	0	1	0	0	0	imm16-low								15+4×imm8	
	imm16-high								imm8									
PS:	0	0	1	0	1	1	1	0	-								1	
PUSH imm16	0	1	1	0	1	0	s	0	imm16-low								1	
	imm16-high								-									
PUSH imm8	0	1	1	0	1	0	s	0	imm8								1	
PUSH mem16	1	1	1	1	1	1	1	1	mod	1	1	0	mem			2		
	(disp-low)								(disp-high)									

表A - 1 命令実行クロック数一覧 (8/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
PUSH PSW	1	0	0	1	1	1	0	0	-								2
PUSH R	0	1	1	0	0	0	0	0	-								9
PUSH reg16	0	1	0	1	0	reg			-								1
PUSH reg16	1	1	1	1	1	1	1	1	1	1	1	1	1	0	reg		1
PUSH sreg	0	0	0	sreg		1	1	0	-								2
REP INM	1	1	1	1	0	0	1	1	0	1	1	0	1	1	0	W	5+6xrep
REP LDM/LDMB/LDMW	1	1	1	1	0	0	1	1	1	0	1	0	1	1	0	W	5+6xrep
REP MOVBK	1	1	1	1	0	0	1	1	1	0	1	0	0	1	0	W	5+7xrep
REP OUTM	1	1	1	1	0	0	1	1	0	1	1	0	1	1	1	W	5+6xrep
REP STM/STMB/STMW	1	1	1	1	0	0	1	1	1	0	1	0	1	0	1	W	5+6xrep
REPE CMPM/CMPMB/CMPMW	1	1	1	1	0	0	1	1	1	0	1	0	1	1	1	W	5+9xrep
REPNE CMPM/CMPMB/CMPMW	1	1	1	1	0	0	1	0	1	0	1	0	1	1	1	W	5+9xrep
REPNZ CMPBK/CMPBKB/CMPBKW	1	1	1	1	0	0	1	0	1	0	1	0	0	1	1	W	5+9xrep
REPZ CMPBK/CMPBKB/CMPBKW	1	1	1	1	0	0	1	1	1	0	1	0	0	1	1	W	5+10xrep
RET pop-value (セグメント外コール)	1	1	0	0	1	0	1	0	pop-value-low								9
	pop-value-high								-								
RET pop-value (セグメント内コール)	1	1	0	0	0	0	1	0	pop-value-low								6
	pop-value-high								-								
RET (セグメント外コール)	1	1	0	0	1	0	1	1	-								8
RET (セグメント内コール)	1	1	0	0	0	0	1	1	-								6
RETI	1	1	0	0	1	1	1	1	-								10
ROL mem, 1	1	1	0	1	0	0	0	W	mod	0	0	0	mem			3	
	(disp-low)								(disp-high)								
ROL mem, CL	1	1	0	1	0	0	1	W	mod	0	0	0	mem			5	
	(disp-low)								(disp-high)								
ROL mem, imm8	1	1	0	0	0	0	0	W	mod	0	0	0	mem			5	
	(disp-low)								(disp-high)								
	imm8								-								
ROL reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	0	reg		1	
ROL reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	0	reg		3	
ROL reg, imm8	1	1	0	0	0	0	0	W	1	1	0	0	0	reg		3	
	imm8								-								
ROLC mem, 1	1	1	0	1	0	0	0	W	mod	0	1	0	mem			3	
	(disp-low)								(disp-high)								
ROLC mem, CL	1	1	0	1	0	0	1	W	mod	0	1	0	mem			5	
	(disp-low)								(disp-high)								
ROLC mem, imm8	1	1	0	0	0	0	0	W	mod	0	1	0	mem			5	
	(disp-low)								(disp-high)								
	imm8								-								
ROLC reg, 1	1	1	0	1	0	0	0	W	1	1	0	1	0	reg		1	
ROLC reg, CL	1	1	0	1	0	0	1	W	1	1	0	1	0	reg		3	

表A - 1 命令実行クロック数一覧 (9/11)

ニモニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
ROL reg, imm8	1	1	0	0	0	0	0	W	1	1	0	1	0			reg	3
	imm8								-								
ROR mem, 1	1	1	0	1	0	0	0	W	mod	0	0	1			mem	3	
	(disp-low)								(disp-high)								
ROR mem, CL	1	1	0	1	0	0	1	W	mod	0	0	1			mem	5	
	(disp-low)								(disp-high)								
ROR mem, imm8	1	1	0	0	0	0	0	W	mod	0	0	1			mem	5	
	(disp-low)								(disp-high)								
	imm8								-								
ROR reg, 1	1	1	0	1	0	0	0	W	1	1	0	0	1		reg	1	
ROR reg, CL	1	1	0	1	0	0	1	W	1	1	0	0	1		reg	3	
ROR reg, imm8	1	1	0	0	0	0	0	W	1	1	0	0	1		reg	3	
	imm8								-								
RORC mem, 1	1	1	0	1	0	0	0	W	mod	0	1	1			mem	3	
	(disp-low)								(disp-high)								
RORC mem, CL	1	1	0	1	0	0	1	W	mod	0	1	1			mem	5	
	(disp-low)								(disp-high)								
RORC mem, imm8	1	1	0	0	0	0	0	W	mod	0	1	1			mem	5	
	(disp-low)								(disp-high)								
	imm8								-								
RORC reg, 1	1	1	0	1	0	0	0	W	1	1	0	1	1		reg	1	
RORC reg, CL	1	1	0	1	0	0	1	W	1	1	0	1	1		reg	3	
RORC reg, imm8	1	1	0	0	0	0	0	W	1	1	0	1	1		reg	3	
	imm8								-								
SET1 CY	1	1	1	1	1	0	0	1								4	
SET1 DIR	1	1	1	1	1	1	0	1								4	
SHL mem, 1	1	1	0	1	0	0	0	W	mod	1	0	0			mem	3	
	(disp-low)								(disp-high)								
SHL mem, CL	1	1	0	1	0	0	1	W	mod	1	0	0			mem	5	
	(disp-low)								(disp-high)								
SHL mem, imm8	1	1	0	0	0	0	0	W	mod	1	0	0			mem	5	
	(disp-low)								(disp-high)								
	imm8								-								
SHL reg, 1	1	1	0	1	0	0	0	W	1	1	1	0	0		reg	1	
SHL reg, CL	1	1	0	1	0	0	1	W	1	1	1	0	0		reg	3	
SHL reg, imm8	1	1	0	0	0	0	0	W	1	1	1	0	0		reg	3	
	imm8								-								
SHR mem, 1	1	1	0	1	0	0	0	W	mod	1	0	1			mem	3	
	(disp-low)								(disp-high)								
SHR mem, CL	1	1	0	1	0	0	1	W	mod	1	0	1			mem	5	
	(disp-low)								(disp-high)								

表A - 1 命令実行クロック数一覧 (10/11)

ニモニック,オペランド	オペレーション・コード																クロック数	
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0		
SHR mem, imm8	1	1	0	0	0	0	0	W	mod	1	0	1	mem				5	
	(disp-low)								(disp-high)									
	imm8								-									
SHR reg, 1	1	1	0	1	0	0	0	W	1	1	1	0	1	reg				1
SHR reg, CL	1	1	0	1	0	0	1	W	1	1	1	0	1	reg				3
SHR reg, imm8	1	1	0	0	0	0	0	W	1	1	1	0	1	reg				3
	imm8								-									
SHRA mem, 1	1	1	0	1	0	0	0	W	mod	1	1	1	mem				3	
	(disp-low)								(disp-high)									
SHRA mem, CL	1	1	0	1	0	0	1	W	mod	1	1	1	mem				5	
	(disp-low)								(disp-high)									
SHRA mem, imm8	1	1	0	0	0	0	0	W	mod	1	1	1	mem				5	
	(disp-low)								(disp-high)									
	imm8								-									
SHRA reg, 1	1	1	0	1	0	0	0	W	1	1	1	1	1	reg				1
SHRA reg, CL	1	1	0	1	0	0	1	W	1	1	1	1	1	reg				3
SHRA reg, imm8	1	1	0	0	0	0	0	W	1	1	1	1	1	reg				3
	imm8								-									
SS:	0	0	1	1	0	1	1	0									-	1
STM [DS1-spec:] dst-block	1	0	1	0	1	0	1	W									-	3
STMB	1	0	1	0	1	0	1	W									-	3
STMW	1	0	1	0	1	0	1	W									-	3
SUB acc, imm	0	0	1	0	1	1	0	W	imm8 or imm16-low								1	
	imm16-high								-									
SUB mem, imm	1	0	0	0	0	0	s	W	mod	1	0	1	mem				3	
	(disp-low)								(disp-high)									
	imm8 or imm16-low								imm16-high									
SUB mem, reg	0	0	1	0	1	0	0	W	mod	reg		mem				3		
	(disp-low)								(disp-high)									
SUB reg, imm	1	0	0	0	0	0	s	W	1	1	1	0	1	reg				1
	imm8 or imm16-low								imm16-high									
SUB reg, mem	0	0	1	0	1	0	1	W	mod	reg		mem				2		
	(disp-low)								(disp-high)									
SUB reg, reg'	0	0	1	0	1	0	1	W	1	1	reg		reg'				1	
SUB reg, reg'	0	0	1	0	1	0	0	W	1	1	reg		reg'				1	
SUBC acc, imm	0	0	0	1	1	1	0	W	imm8 or imm16-low								1	
	imm16-high								-									
SUBC mem, imm	1	0	0	0	0	0	s	W	mod	0	1	1	mem				3	
	(disp-low)								(disp-high)									
	imm8 or imm16-low								imm16-high									
SUBC mem, reg	0	0	0	1	1	0	0	W	mod	reg		mem				3		
	(disp-low)								(disp-high)									

表A - 1 命令実行クロック数一覧 (11/11)

二モニック,オペランド	オペレーション・コード																クロック数
	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
SUBC reg, imm	1	0	0	0	0	0	s	W	1	1	0	1	1			reg	1
	imm8 or imm16-low								imm16-high								
SUBC reg, mem	0	0	0	1	1	0	1	W	mod			reg			mem	2	
	(disp-low)								(disp-high)								
SUBC reg, reg'	0	0	0	1	1	0	1	W	1	1		reg			reg'	1	
SUBC reg, reg'	0	0	0	1	1	0	0	W	1	1		reg			reg'	1	
TEST acc, imm	1	0	1	0	1	0	0	W	imm8 or imm16-low								1
	imm16-high								-								
TEST mem, imm	1	1	1	1	0	1	1	W	mod	0	0	0			mem	2	
	(disp-low)								(disp-high)								
	imm8 or imm16-low								imm16-high								
TEST mem,reg	1	0	0	0	0	1	0	W	mod		reg			mem	2		
	(disp-low)								(disp-high)								
TEST reg, imm	1	1	1	1	0	1	1	W	1	1	0	0	0		reg	1	
	imm8 or imm16-low								imm16-high								
TEST reg, mem	1	0	0	0	0	1	0	W	mod		reg			mem	2		
	(disp-low)								(disp-high)								
TEST reg, reg'	1	0	0	0	0	1	0	W	1	1		reg			reg'	1	
TRANS	1	1	0	1	0	1	1	1	-								5
TRANS src-table	1	1	0	1	0	1	1	1	-								5
TRANSB	1	1	0	1	0	1	1	1	-								5
XCH AW, reg16	1	0	0	1	0			reg	-								3
XCH mem, reg	1	0	0	0	0	1	1	W	mod		reg			mem	5		
	(disp-low)								(disp-high)								
XCH reg, mem	1	0	0	0	0	1	1	W	mod		reg			mem	5		
	(disp-low)								(disp-high)								
XCH reg, reg'	1	0	0	0	0	1	1	W	1	1		reg			reg'	3	
XCH reg16, AW	1	0	0	1	0			reg	-								3
XOR acc, imm	0	0	1	1	0	1	0	W	imm8 or imm16-low								1
	imm16-high								-								
XOR mem, imm	1	0	0	0	0	0	0	W	mod	1	1	0		mem	3		
	(disp-low)								(disp-high)								
	imm8 or imm16-low								imm16-high								
XOR mem, reg	0	0	1	1	0	0	0	W	mod		reg			mem	3		
	(disp-low)								(disp-high)								
XOR reg, imm	1	0	0	0	0	0	0	W	1	1	1	1	0		reg	1	
	imm8 or imm16-low								imm16-high								
XOR reg, mem	0	0	1	1	0	0	1	W	mod		reg			mem	2		
	(disp-low)								(disp-high)								
XOR reg, reg'	0	0	1	1	0	0	1	W	1	1		reg			reg'	1	
XOR reg, reg'	0	0	1	1	0	0	0	W	1	1		reg			reg'	1	

付録 B 総合索引

B.1 50音で始まる語句の索引

【あ行】

アドレッシング・モード ... 44
アドレス空間 ... 33
イミューディエト・アドレッシング ... 45
インデクスト・アドレッシング ... 47
インデクス・レジスタ ... 32
オーバフロー・フラグ ... 30

【か行】

キャリー・フラグ ... 29
コントロール・フラグ ... 28

【さ行】

サイン・フラグ ... 30
実効アドレス ... 41
シンボル図 ... 15
スタック・セグメント ... 39
スタンバイ機能 ... 67
スタンバイ・テスト・モード ... 72
スタンバイ・モード ... 67
スタンバイ・モードの解除 ... 68
スタンバイ・モードの設定 ... 67
ステータス・フラグ ... 28
セグメントの構成 ... 37
セグメント方式 ... 36
セグメント・レジスタ ... 27
ゼロ・フラグ ... 30
ソフトウェア割り込み ... 62

【た行】

ダイナミック・リロケーション ... 39
ダイレクト・アドレッシング ... 44, 46

ダブル・ワード・データの構成 ... 34

端子一覧 ... 19
端子機能 ... 19
端子機能の説明 ... 21
端子状態 ... 20
端子の未使用時の処理 ... 25
単体テスト・モード ... 72
データ・アドレス ... 45
データ・セグメント0 ... 39
データ・セグメント1 ... 39
テスト機能 ... 71
テスト・バス端子 ... 71

【な行】

ノーマル・モード ... 72
ノンマスカブル割り込み ... 60

【は行】

ハードウェア割り込み ... 60
バス制御機能 ... 49
バス・ホールド機能 ... 56
パリティ・フラグ ... 30
汎用レジスタ ... 27
非メモリ・アドレッシング ... 45
物理アドレス ... 36
ブレーク・フラグ ... 31
プログラム・カウンタ ... 28
プログラム・ステータス・ワード ... 28
プログラム・セグメント ... 39
ブロック処理命令実行中の割り込み処理 ... 64
ベースト・アドレッシング ... 47
ベースト・インデクス付きアドレッシング ... 47
ポインタ ... 27
方向フラグ ... 32

補助キャリア・フラグ ... 30

[ま行]

マスクブル割り込み ... 60

命令アドレス ... 44

命令語の形式 ... 43

命令実行クロック数一覧 ... 73

命令セット ... 42

命令プリフェッチ ... 35

メモリ, I/O のライト・タイミング ... 54

メモリ, I/O のリード・タイミング ... 52

メモリ・アドレッシング ... 46

メモリ・アドレスの計算 ... 41

メモリ間接アドレッシング ... 44

メモリ空間 ... 33

メモリとのインタフェース ... 49

メモリ・マップ ... 33

モード・フラグ ... 32

[ら行]

リセット機能 ... 69

リセット後の各レジスタの初期値 ... 69

リセット時の出力端子状態 ... 69

レジスタ・アドレッシング ... 45

レジスタ間接アドレッシング ... 44, 46

レジスタ構成 ... 27

論理アドレス ... 36

[わ行]

ワード・データの構成 ... 34

割り込みアクノリッジ・サイクル ... 61

割り込みが受け付けられないタイミング ... 63

割り込み機能 ... 57

割り込み許可フラグ ... 31

割り込みベクタ・テーブルの構成 ... 58

B.2 アルファベットで始まる語句の索引

[A]

A19-A0 ... 21

AC ... 30

AH ... 27

AL ... 27

AW ... 27

[B]

BH ... 27

BL ... 27

BP ... 27

BRK ... 31

BS3-BS0 ... 21

BUNRI ... 24,71

BUSLOCKB ... 22

BW ... 27

[C]

CH ... 27

CL ... 27

CLK ... 23

CPU 機能 ... 27

CW ... 27

CY ... 29

[D]

DBA20 ... 24

DBHLTST ... 24

DBINT ... 24

DBMODE ... 24

DBNMIM ... 24

DBRD ... 24

DBWR ... 24

DH ... 27

DI15-DI0 ... 21

DIR ... 32

DL ... 27

DO15-DO0 ... 21

DS0 ... 27

DS1 ... 27

DW ... 27

[H]

HLDK ... 23

HLDRQ ... 22

[I]

I/O アドレッシング ... 45

I/O 空間 ... 34

I/O とのインタフェース ... 51

I/O マップ ... 34

IE ... 31

INT ... 23, 60

IX ... 32

IY ... 32

[M]

MD ... 32

[N]

NMI ... 23, 60

[P]

P ... 30

PC ... 28

PC 相対アドレッシング ... 44

POLLB ... 22

PS ... 27

PSW ... 28

[R]

READYB ... 22

RESET ... 22

[S]

S ... 30

SP ... 27

SS ... 27

[T]

TBI22-TBI0 ... 24, 71

TBO42-TBO0 ... 24, 71

TBRA ... 24

TEOI ... 24

TEST ... 24, 71

TILEN3-TILEN0 ... 24

TINTA ... 24

[U]

UBEB ... 21

[V]

V ... 30

V30MZ と V30HL, V30MX の違い ... 16

[Z]

Z ... 30

[メモ]

— お問い合わせ先 —

【技術的なお問い合わせ先】

NEC半導体テクニカルホットライン
(電話：午前 9:00～12:00，午後 1:00～5:00)

電話 : 044-435-9494
FAX : 044-435-9608
E-mail : info@lsi.nec.co.jp

【営業関係お問い合わせ先】

第一販売事業部

東京 (03)3798-6106, 6107,
6108
大阪 (06)6945-3178, 3200,
3208, 3212
広島 (082)242-5504
仙台 (022)267-8740
郡山 (024)923-5591
千葉 (043)238-8116

第二販売事業部

東京 (03)3798-6110, 6111,
6112
立川 (042)526-5981, 6167
松本 (0263)35-1662
静岡 (054)254-4794
金沢 (076)232-7303
松山 (089)945-4149

第三販売事業部

東京 (03)3798-6151, 6155, 6586,
1622, 1623, 6156
水戸 (029)226-1702
前橋 (027)243-6060
鳥取 (0857)27-5313
太田 (0276)46-4014
名古屋 (052)222-2170, 2190
福岡 (092)261-2806

【資料の請求先】

上記営業関係お問い合わせ先またはNEC特約店へお申しつけください。

【NECエレクトロニクス デバイス ホームページ】

NECエレクトロニクスデバイスの情報がインターネットでご覧になれます。

URL(アドレス) <http://www.ic.nec.co.jp/>

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] V30MZ ユーザーズ・マニュアル(暫定) ハードウエア編
(A13761JJ1V2UM00(第1版))

[お名前など](さしつかえない範囲で)

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, その他 ()

ご協力ありがとうございました。

下記あてに FAX で送信いただくか, 最寄りの販売員にコピーをお渡しください。

日本電気(株) NEC エレクトロニクス
半導体テクニカルホットライン
FAX : (044) 435-9608

2000.6