

KAWATETSU



高速8ビットマイクロコントローラ

**KL5C8012CFP**

**KL5C80A12CFP**

**ハードウェアマニュアル**

# 高速 8 ビットマイクロコントローラ KL5C8012CFP KL5C80A12CFP ハードウェアマニュアル

1. 概要
  2. ブロック図
  3. 端子説明
  4. 外部バス・インターフェース・ユニット
  5. KC82 CPU
  6. 割込みコントローラ
  7. USART(同期/ 非同期シリアルインターフェース)
  8. タイマ/ カウンタ A
  9. タイマ/ カウンタ B
  10. パラレルポート A
  11. パラレルポート B
  12. 動作モード設定
  13. アドレスマッピング
  14. 発振回路
  15. その他
  16. 電気的特性
  17. 外形寸法図
- 付録 A インストラクション表

1995.2 Ver. 2.0  
川崎製鉄株式会社  
LSI 事業部

## 1. 概要

KL5C8012CFP, KL5C80A12CFP は共に最新の CMOS プロセスを駆使して開発された高速 8 ビットマイクロコントローラです。KL5C8012CFP と KL5C80A12CFP はコンパチブルな仕様になっており、以下本データシートでは両製品を総称して KL5C8012 と呼びます。CPU コアに Zilog 社の Z80 とバイナリ・レベルでコンパチブルで、アドレス空間を拡張する MMU 内蔵の KC82 を採用しています。KC82 は Z80 (10MHz) の約 4 倍の超高速性能を持ち、今までの 8 ビットマイコンの常識を越える性能を発揮する新世代の CPU コアです。KL5C8012 は CPU コア以外にも高速 RAM512 バイト、割り込みコントローラ、16 ビット高機能タイマ/カウンタ 5 本、同期/非同期シリアルコミュニケーション・インターフェース (USART)、パラレル・ポート 40 本を内蔵しており、高性能かつ小型化が要求されるシステムに最適です。また、KL5C8012 は低消費電力で、携帯装置への応用にも適しています。

### 特長

- 1) Z80 とバイナリ・コンパチブル
- 2) アドレス空間を 512K バイトに拡張する MMU
- 3) USART (同期/非同期シリアル・コミュニケーション・インターフェース)
- 4) 16 ビット高機能タイマ/カウンタ 2 本 (外部カウンタ・クロック入力可能)
- 5) 16 ビット高機能タイマ/カウンタ 3 本
- 6) 割り込み 内部 8 本 外部 8 本、優先順位設定可能  
    ノン・マスクابل割り込み 1 本
- 7) 40 本のパラレル・ポート
- 8) 高速 RAM 512 バイト内蔵
- 9) 外部メモリ・チップ・セレクト回路内蔵
- 10) 水晶発振パツファ搭載
- 11) 最高動作周波数 10 MHz
- 12) 低消費電力

KL5C8012 の CPU コア (KC82) のシステムバスはクロック同期型のバスを採用しています。KL5C8012 の内部 I/O や内部高速 RAM はこの KC82 のクロック同期型バスに対応したバス・インターフェースを持っており、そのまま接続されています。KL5C8012 の外部に I/O やメモリを接続する場合は、このままでは接続できませんので KC82 のクロック同期型バスを通常のメモリが接続できる非同期バスに変換する必要があります。KL5C8012 にはこの変換を行う回路 (外部バス・インターフェース・ユニット) が搭載されており、KL5C8012 外部に出力されているリード/ライト信号 (EMRD<sub>0</sub>, EMWR<sub>0</sub>, EIORD<sub>0</sub>, EIOWR<sub>0</sub>) は外部非同期バスに変換されたリード/ライト信号です。この外部バスサイクルのリード/ライト信号は内部バスサイクルのクロック同期バスのリード/ライト信号と異なり、外部の通常のメモリを直接接続できるリード/ライト用のストローブ型の信号となっているため使いやすくなっています。本データシートでは KC82 のクロック同期型バスを内部バスサイクル、外部バス・インターフェース・ユニットにより変換された外部非同期バスを外部バスサイクルと呼んでいます。

CPU の章の図はすべて内部バスサイクルを基本として描かれています。すなわち、KL5C8012 内部の KC82 の動作説明という形で記述されています。あらかじめ御了承ください。したがって、CPU の章にある説明の図と同様な動作で外部メモリや外部 I/O をアクセスする場合には外部ウェイト入力あるいはウェイト・ステート・コントローラによるウェイトが挿入されることがあります。外部バスサイクルに関しては外部バス・インターフェース・ユニットの章を参照下さい。

また、外部のメモリを効率良く接続するため外部バス・インターフェース・ユニットはウェイト・ステート・コントローラを内蔵しています。これは SCR1 (システム・コントロール・レジスタ 1) の 2 ビットにより、外部メモリアクセスや外部 I/O アクセス時に自動的に設定に応じたウェイトを挿入しながら、外部バス・タイミング・チャートの 0 ウェイト・外部メモリアクセス、1 ウェイト・外部メモリアクセス、1 ウェイト・外部 I/O アクセス、2 ウェイト・外部 I/O アクセスを発生させるものです。このウェイト・ステート・コントローラは本チップの内部物理アドレス空間 1M byte を 2 つに分割して制御しており、高速 SRAM と EPROM といったアクセス・タイムの違う 2 つのメモリを効率良く接続できるように工夫されています。内部メモリアクセスは SCR1 の設定に関係なく常に 0 ウェイト・内部メモリサイクルで実行されます。また、0 ウェイト・外部メモリアクセスでは外部ウェイト信号 (ERDY) は無視されますので注意が必要です。

SCR0, SCR1 の他にモード設定用の外部入力端子 (MODE[1:0]) が 2 本あり、MODE[1] = 0、MODE[0] = 1 を選択し、外部端子 BFSIO にバグ・ファインダ・アダプタを接続することで当社の簡易型デバッグツール (バグ・ファインダ) がリセット直後に立ち上がります。このとき外部の ROM の代わりに RAM を接続しますと ICE のようにパソコンから RAM ヘブプログラムのダウンロードができ、この RAM 上でシングル・ステップ実行等のデバッグ作業が可能です。

### 2. ブロック図

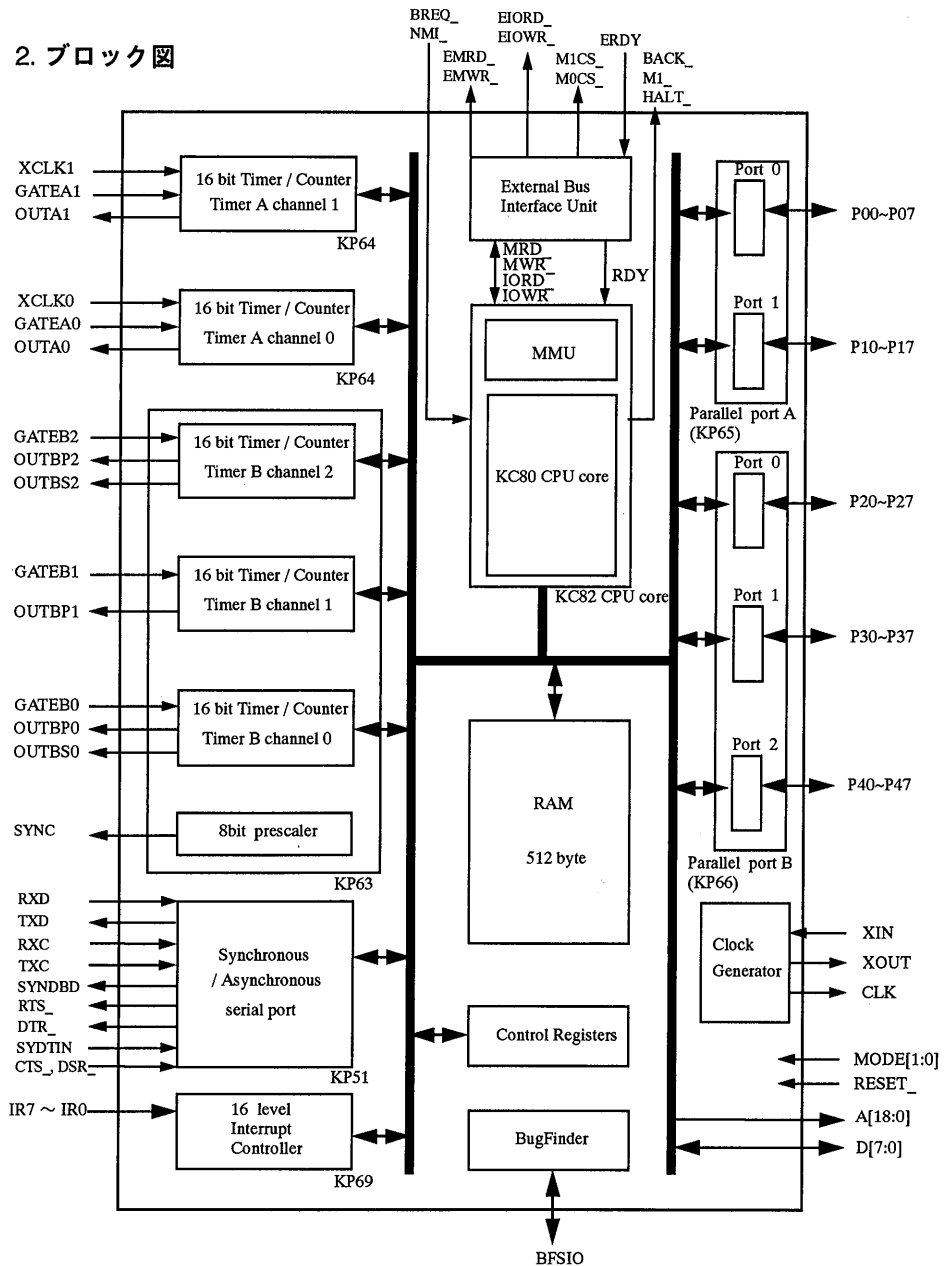


図 2-1 KL5C8012 のブロック図

### 3. 端子説明

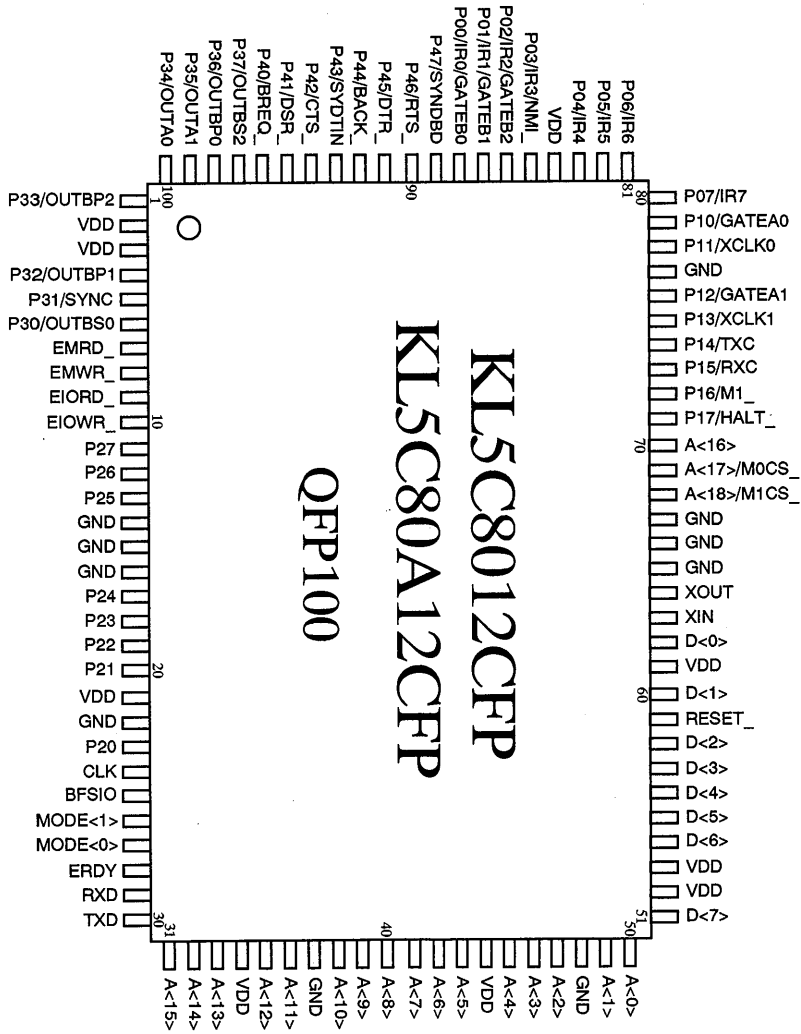


図 3-1 ピン配置図(上面図)

KL5C8012 の端子は専用端子とマルチプレクスされる端子に分けられます。専用端子については表 3-1 を参照して下さい。マルチプレクスされる端子は表 3-2 の端子が表 3-3 のパラレルポート端子とマルチプレクスされています。どの端子とどの端子がマルチプレクスされるかはピン配置図を参照して下さい。また、すべての入出力端子とすべてのトライステート出力端子 (A[15:0], EMRD\_, EMWR\_, EIORD\_, EIWWR\_) と入力端子の一部 (MODE[1:0], RXD) にプルアップ抵抗がついています。

表 3-1 専用端子

端子名	入力/出力	機能説明
GND	電源	0V を接続して下さい。
VDD	電源	5V の電源を接続して下さい。
EMRD_	出力	外部メモリアドレス信号。KC80 の内部バス信号 MRD_ を外部バス・インターフェース・ユニットで外部バス用に変換した信号です。内部 RAM アクセスのときは“H”のままです。
EMWR_	出力	外部メモリアドレス信号。KC80 の内部バス信号 MWR_ を外部バス・インターフェース・ユニットで外部バス用に変換した信号です。内部 RAM アクセスのときは“H”のままです。
EIORD_	出力	外部 I/O デバイスリード信号。KC80 の内部バス信号 IORD_ を外部バス・インターフェース・ユニットで外部バス用に変換した信号です。内部 I/O アクセスのときは“H”のままです。
EIWWR_	出力	外部 I/O デバイスライト信号。KC80 の内部バス信号 IWWR_ を外部バス・インターフェース・ユニットで外部バス用に変換した信号です。内部 I/O アクセスのときは“H”のままです。
CLK	出力	水晶発振子で発生した内部クロックをチップ外の回路へ出力します。
BFSIO	入出力	デバッグ・ツールのバグ・ファインダ専用ポートです。バグ・ファインダを利用しないときはオープンにして下さい。
MODE[1:0]	入力	モード設定用の入力端子。KL5C8012 の動作モードを設定します。
ERDY	入力	外部からのウェイト要求入力。
A[16:0]	出力	アドレス出力。CPU の A[16:0] が接続されています。
D[7:0]	入出力	外部データバス。
RESET_	入力	リセット入力。”L” で本回路はリセットされます。
XIN	水晶発振子用端子	内蔵のクロック発生回路に外部の水晶発振子を接続します。周波数はシステムクロックの 2 倍のものを使用して下さい。水晶発振子を使用しない場合はシステムクロックの 2 倍の周波数のクロックを入力して下さい。
XOUT	水晶発振子用端子	内蔵のクロック発生回路に外部の水晶発振子を接続します。周波数はシステムクロックの 2 倍のものを使用して下さい。
A[17]/M0CS_	出力	CPU のアドレス出力 A[17] あるいは外部 ROM チップセレクト出力 (アドレス = 00000H~1FFFFH) として動作します。
A[18]/M1CS_	出力	CPU のアドレス出力 A[18] あるいは外部 RAM/ROM チップセレクト出力 (アドレス = E0000H~FFDFFH) として動作します。
RXD	入力	USART の受信データ入力端子です。シリアルな受信データが入力される端子です。受信データは RXC の立ち上がりでサンプルされ、設定されたポーレートに応じて、内部に取り込まれます。

端子名	入力/出力	機能説明
TXD	出力	USARTの送信データ出力端子です。送信データがシリアルに出力される端子です。送信データは設定されたボーレートに応じて、TXCの立ち下がりに同期して出力されます。送信ブレイクコマンド (sbrk= '1') をライトすると、TXD 出力は強制的に "L" の状態にマスクされます (ブレイク状態)。ブレイク解除コマンド (sbrk= '1') するまでブレイク状態は続き、その間も受信バッファおよび TXRDYPIN、TXEMPTY 出力は、最初に設定されたキャラクタ長に従って、通常のキャラクタを送信しているのと同じタイミングで動作します。また CTS_ = "L" で、かつ送信イネーブル (txen= '1') で、送信データは出力可能となります。現在データ送信中に、CTS_ = "H" または送信ディセーブル (txen= '0') になると、送信すべきデータがなくなった時点でマーキング状態になります。

表 3-2 パラレルポートとマルチプレクスされる端子

端子名	入力/出力	機能説明
IR7~0	入力	外部割り込み入力レベル 7~0 です。詳細は 6 章を参照して下さい。
XCLK1	入力	カウンタ A チャンネル 1 への外部カウンタ・クロック入力です。詳細は 8 章を参照して下さい。
XCLK0	入力	カウンタ A チャンネル 0 への外部カウンタ・クロック入力です。詳細は 8 章を参照して下さい。
GATEA1	入力	カウンタ A チャンネル 1 への GATE 入力です。詳細は 8 章を参照して下さい。
GATEA0	入力	カウンタ A チャンネル 0 への GATE 入力です。詳細は 8 章を参照して下さい。
GATEB2	入力	カウンタ B チャンネル 2 への GATE 入力です。詳細は 9 章を参照して下さい。
GATEB1	入力	カウンタ B チャンネル 1 への GATE 入力です。詳細は 9 章を参照して下さい。
GATEB0	入力	カウンタ B チャンネル 0 への GATE 入力です。詳細は 9 章を参照して下さい。
OUTA1	出力	カウンタ A チャンネル 1 の OUT 出力です。詳細は 8 章を参照して下さい。
OUTA0	出力	カウンタ A チャンネル 0 の OUT 出力です。詳細は 8 章を参照して下さい。
OUTBS2	出力	カウンタ B チャンネル 2 のストロブ出力です。詳細は 9 章を参照して下さい。
OUTBS0	出力	カウンタ B チャンネル 0 のストロブ出力です。詳細は 9 章を参照して下さい。
OUTBP2	出力	カウンタ B チャンネル 2 のパルス出力です。詳細は 9 章を参照して下さい。
OUTBP1	出力	カウンタ B チャンネル 1 のパルス出力です。詳細は 9 章を参照して下さい。
OUTBP0	出力	カウンタ B チャンネル 0 のパルス出力です。詳細は 9 章を参照して下さい。
SYNC	出力	カウンタ B のプリスケアラの同期出力です。詳細は 9 章を参照して下さい。
HALT_	出力	CPU のホルト信号出力端子です。 アクティブ "L"。CPU が HALT 命令を実行し、HALT 状態にあることを示します。HALT 状態は、ノンマスクابل割り込みやマスクابل割り込み、あるいはリセット入力により解除されます。HALT 中 CPU は NOP 命令を実行していません。

端子名	入力/出力	機能説明
M1_	出力	CPUのM1サイクル出力端子です。 アクティブ“L”。現在のマシンサイクルが命令実行中のオペコードフェッチサイクルであることを示します。(2バイト命令実行時は連続して“L”を出力します)。
BACK_	出力	CPUのバスアクノリッジ出力端子です。 他のバスマスタに対し、バスを解放したことを通知する信号です。
BREQ_	入力	CPUのバスリクエスト入力端子です。 アクティブ“L”。バスリクエストがアクティブになると、現在実行中の命令終了後、アドレスバス、データバスはハイ・インピーダンス状態になります。BREQ_はNMI_、INT_よりも上位の優先順位を持っています。
NMI_	入力	CPUのノンマスクブル割り込み入力端子です。 立ち下がりがエッジ検知。このノンマスクブル割り込み要求の優先順位は、INT_よりも高く、BREQ_よりも低くなっています。現在実行中の命令処理終了後、割り込み許可フラグの状態に関係なく、0066H番地から始まる割り込み処理プログラムを実行します。
RXC	入力	USARTの受信クロック入力端子です。RXD入力端子から、データを取り込んだり、受信する際のボーレートを制御するクロックです。
TXC	入力	USARTの送信クロック入力端子です。TXD出力端子からデータを送信する際のボーレートを制御するクロックです。
SYNDBD	出力	USARTの同期検出信号/ブレイク検出信号出力端子です。この端子はリセット後は“L”を出力しますが、モード設定により異なった機能を持ちます。syncモード内部同期検出にプログラムされた場合、エンタハントコマンド発行後、syncキャラクタを受信検出すると“H”を出力します。タイミングとしてはsyncキャラクタ最終ビットの中央(bi-syncモードでは2番目のsyncキャラクタの最終ビットの中央/パリティイネーブル時はパリティビットの中央)で“H”にセットされます。そしてステータスリードにより“L”にリセットされます。syncモード外部同期検出にプログラムされた場合はRXC=“H”のときにSYDTIN=“H”を検出すると“H”を出力します。そして内部同期検出の場合同様ステータスリードにより“L”にリセットされます。次にasyncモードの場合、ブレイク信号を検出したことを示す出力端子として用いられます。RXD端子から2キャラクタデータブロック(スタートビット、パリティビット、ストップビットを含む)以上“L”を受信するとブレイク信号であると認識してこの端子は“H”を出力します。そしてRXD端子から“H”を受信すると“L”を出力します。
RTS_	出力	USARTのリクエストトゥゼンド信号出力端子です。通常モデムに対するリクエストトゥゼンド信号として用いますが、汎用の出力ポートとしても利用できます。

端子名	入力/出力	機能説明
DTR_	出力	USART のデータターミナルレディ信号出力端子です。通常モデムに対するデータターミナルレディ信号として用いますが、汎用の出力ポートとしても利用できます。
SYDTIN	入力	USART の外部同期検出信号入力端子です。sync モード外部同期検出にプログラムされた場合に RXC=“H” のときこの端子に“H”が入力されていると、エンタハントモードを抜けて、次の RXC の立ち上がりから受信データのサンプリングを開始します。事前にエンタハントコマンドの実行がなければ、データの受信は行いません。“H”の入力は、RXC の 1 周期以上の入力が必要です。また内部同期検出モードではこの端子は機能しません。
CTS_	入力	USART のクリアトゥセンド信号入力端子です。USART への送信要求信号が入力されます。“L”で送信要求中です。通常モデムのクリアトゥセンド信号として用います。送信イネーブル状態 (txen= ‘1’ <コマンドレジスタのビット 0 を ‘1’ にセット>) で、送信すべきデータがある場合、この端子の入力が“L”だと、TXD 端子からデータが送出されます。また入力が“H”になるとバッファ内のデータを全て送信し終わってから、TXD 端子はマーキング状態になります。
DSR_	入力	データセットレディ信号入力端子です。通常モデムのステータスをチェックするデータセットレディ (“L” = モデムは送信可能、“H” = 送信不可能) として用いますが、汎用の入力ポートとしても使用できます。

表 3-3 パラレルポート端子

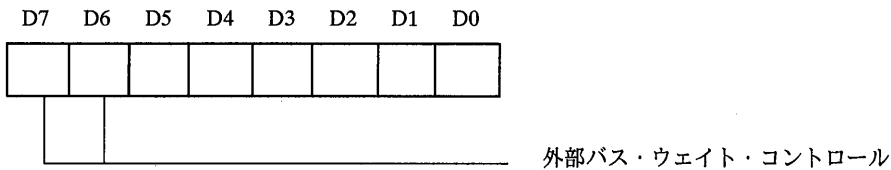
端子名	入力/出力	機能説明
P00~P07	入出力	パラレルポートブロック A の P0 ポートです。詳細は 10 章を参照して下さい。
P10~P17	入出力	パラレルポートブロック A の P1 ポートです。詳細は 10 章を参照して下さい。
P20~P27	入出力	パラレルポートブロック B の P0 ポートです。詳細は 11 章を参照して下さい。
P30~P37	入出力	パラレルポートブロック B の P1 ポートです。詳細は 11 章を参照して下さい。
P40~P47	入出力	パラレルポートブロック B の P2 ポートです。詳細は 11 章を参照して下さい。

### 4. 外部バス・インターフェース・ユニット

#### 4.1 概要

外部バス・インターフェース・ユニットはウェイト・ステート・コントローラを内蔵しており、指定されたウェイトを挿入しながら CPU の内部バスサイクルを外部バスサイクルに変換する回路です。ウェイト数の指定は SCR1 のビット 7、6 で行ないます。このユニットは外部 I/O や外部メモリのアクセスのときのみ動作し、外部バスのリード/ライト信号は内部 I/O と内部高速 RAM のアクセス時には“H”のままです。

SCR1



	外部メモリ (00000 ~ 7FFFF)	外部メモリ (80000 ~ FFDFF)	外部 I/O
00	1 ウェイト	1 ウェイト	2 ウェイト
01	1 " *	1 " *	2 "
10	1 "	0 "	1 "
11	0 "	0 "	1 "

\*ワイド・ライト・ストロブ・オプション

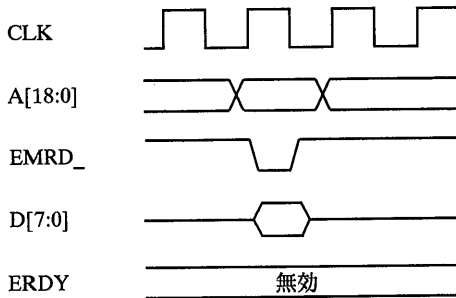
KL5C8012 の CPU コア, KC82 のシステムバスはクロック同期型のバスを採用しています。KL5C8012 の内部 I/O や内部高速 RAM はこの KC82 のクロック同期型バスに対応したバス・インターフェースを持っており、そのまま接続されています。KL5C8012 の外部に I/O やメモリを接続する場合は、外部バス・インターフェース・ユニットが内部バスサイクルのクロック同期バスのリード/ライト信号 (MRD\_, MWR\_, IORD\_, IOWR\_) を次ページ以降のバス・タイミング・チャートにある外部非同期バスのリード/ライト信号 (EMRD\_, EMWR\_, EIORD\_, EIOWR\_) に変換します。外部バス・インターフェース・ユニットはウェイト・ステート・コントローラを内蔵しており、指定されたウェイトを挿入しながら CPU の内部バスサイクルを外部バスサイクルに変換します。ウェイト数の指定は SCR1 のビット 7、6 で行ないま

す。外部バス・インターフェース・ユニットは外部ウェイト入力 (ERDY) から入力された外部ウェイト要求に内蔵のウェイト・ステート・コントローラからのウェイト要求を加えて CPU に渡します。

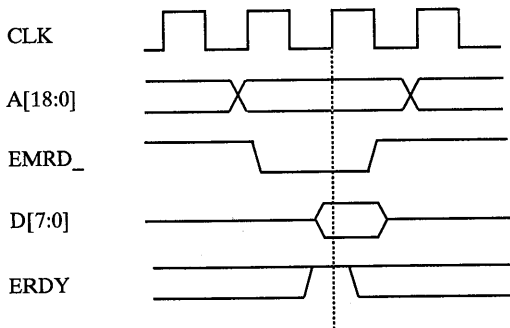
ウェイト・ステート・コントローラは本チップの内部物理アドレス空間 1M byte を 2 つに分割して制御しており、高速 SRAM と中速 EPROM といったアクセス・タイムの違う 2 つのメモリを効率良く接続できるように工夫されています。内部メモリアクセスは SCR1 の設定に関係なく常に 0 ウェイト・内部メモリサイクルで実行されます。また、0 ウェイト・外部メモリアクセスでは外部ウェイト信号 (ERDY) は無視されますので注意が必要です。

### 外部バスサイクル・タイミング図

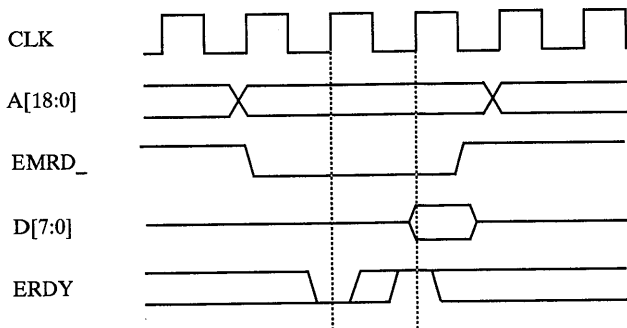
外部メモリアドレス (ウェイト・ステート・コントローラを0ウェイトに設定した時)



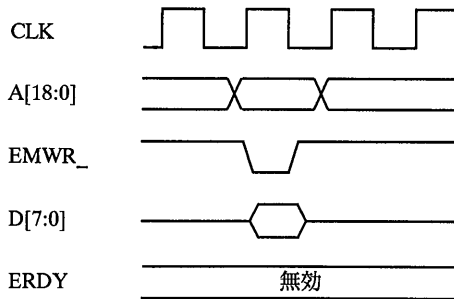
外部メモリアドレス (ウェイト・ステート・コントローラを1ウェイトに設定した時)



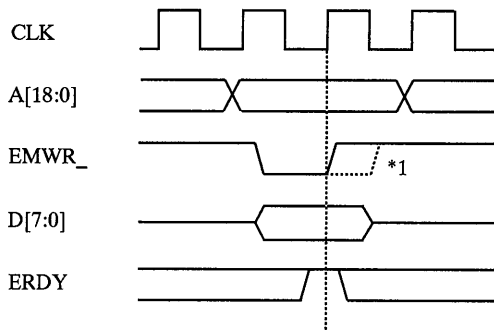
外部メモリアドレス (ウェイト・ステート・コントローラを1ウェイトに設定し、1ウェイト追加した時)



### 外部メモリライト (ウェイト・ステート・コントローラを0ウェイトに設定した時)

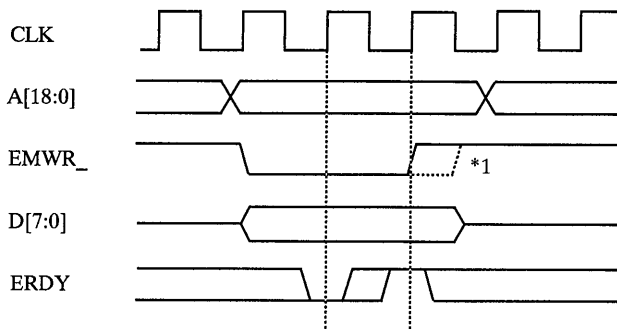


### 外部メモリライト (ウェイト・ステート・コントローラを1ウェイトに設定した時)

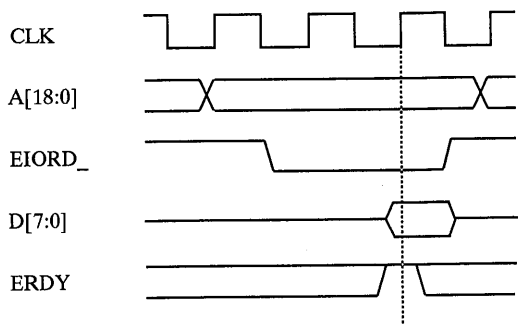


\*1 ワイド・ライト・ストローブ・オプション時の EMWR\_ 信号のタイミング

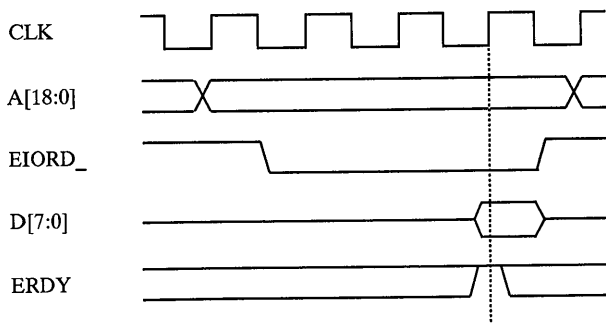
### 外部メモリライト (ウェイト・ステート・コントローラを1ウェイトに設定し、1ウェイト追加した時)



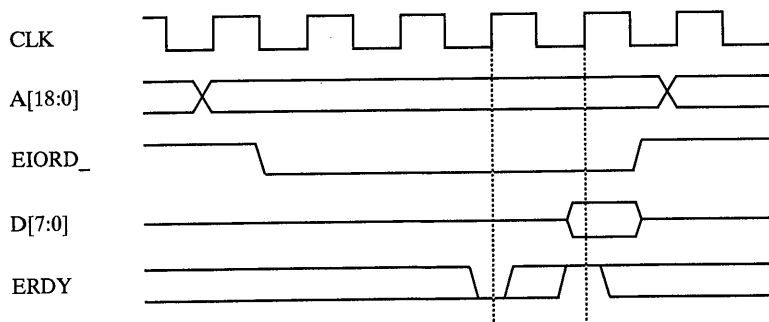
外部 I/O リード (ウェイト・ステート・コントローラを1ウェイトに設定した時)



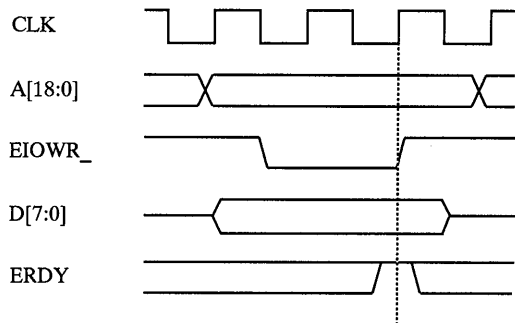
外部 I/O リード (ウェイト・ステート・コントローラを2ウェイトに設定した時)



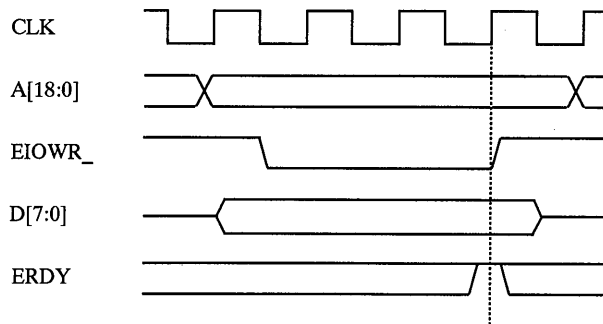
外部 I/O リード (ウェイト・ステート・コントローラを2ウェイトに設定し、1ウェイト追加した時)



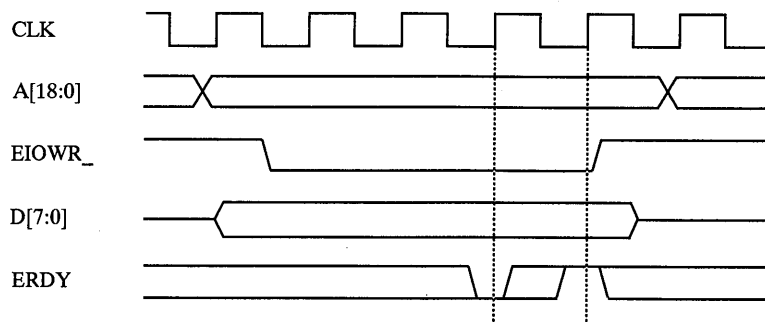
外部 I/O ライト (ウェイト・ステート・コントローラを 1 ウェイトに設定した時)



外部 I/O ライト (ウェイト・ステート・コントローラを 2 ウェイトに設定した時)



外部 I/O ライト (ウェイト・ステート・コントローラを 2 ウェイトに設定し、1 ウェイト追加した時)



### 5. KC82 CPU

#### 5.1 概要

KC82 は KC80 CPU コアに MMU を搭載してアドレス空間を 1M バイトに拡張した新開発の CPU コアです。KC80 は Zilog 社 Z80 とオブジェクトレベルでコンパチブルな CPU コアです。本 CPU コアは同期バスを採用しており、同ファミリである KP6X シリーズや KP55、KP59、KP54、KP37、KP51 と組み合わせて使用することで、従来の非同期バス接続に比べてシステム性能を飛躍的に向上させることが可能です。

#### 特長

KC82 は以下の特長を持っています。

- 1) Zilog 社 Z80 に比較し、KC82 は平均で約 1/4 のクロックサイクルで処理します。

例：命令	Z80	KC82
LD r, r'	4 クロック	1 クロック
ADD HL, ss	11 クロック	1 クロック

- 2) インターフェース用マクロを使用することにより、従来の Z80 ペリフェラル・ファミリとの接続も可能です。
- 3) インストラクション・セットは 158 種類で、Zilog 社 Z80 とオブジェクトコードレベルでフルコンパチブルです。

#### 5.2 ブロック図

図 5-1 に KC82 の全体ブロック図を示します。

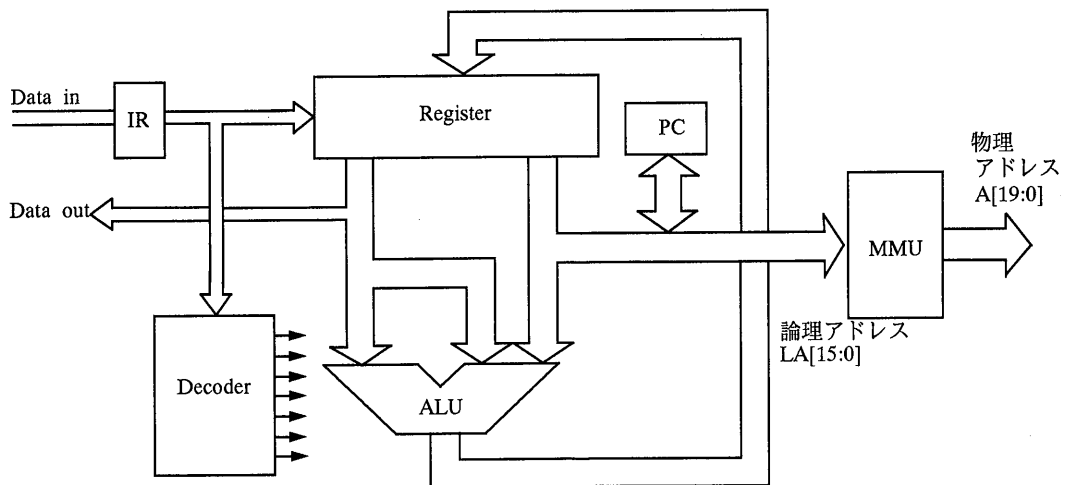


図 5-1 KC82 全体ブロック図

### 5.3 CPU 内部端子機能説明

端子記号	I/O	機能説明
MRD_	O	内部バスメモリリード出力端子 アクティブ“L”。メモリからのデータを読み込む際に出力します。
MWR_	O	内部バスメモリライト出力端子 アクティブ“L”。メモリへデータを書き込む際に出力します。
IORD_	O	内部バス I/O リード出力端子 アクティブ“L”。I/O デバイスからのデータを読み込む際に出力します。 最小 2 クロック期間アクティブとなります。
IOWR_	O	内部バス I/O ライト出力端子 アクティブ“L”。I/O デバイスへデータを書き込む際に出力します。 最小 2 クロック期間アクティブとなります。
WAIT_	I	ウェイト信号入力端子 アクティブ“L”。メモリや I/O デバイスがデータ送用の用意ができていないことを CPU に通知する為の信号です。この信号がアクティブの期間、CPU はウェイト状態を続けます。この入力は外部バス・インターフェース・ユニットのウェイト・ステート・コントローラを介して外部入力端子 ERDY に接続されています。
INT_	I	マスカブル割り込み入力端子 アクティブ“L”。I/O デバイスからの割り込み要求信号を入力します。CPU 内の割り込み許可フラグがセットされていて、BREQ_入力がインアクティブならば、現在実行中の命令処理終了後、割り込み処理プログラムを実行します。この入力端子は割り込みコントローラの INT_出力に接続されています。
IACK_	O	インタラプトアクノリッジ出力端子 アクティブ“L”。マスカブルインタラプトを認識した時、割り込みベクタや命令を I/O デバイスに要求する為の信号です。最小 2 クロック期間アクティブになります。この出力端子は割り込みコントローラに接続されています。この信号に同期して割り込みコントローラから割り込みベクタを読み込みます。
EOI_	O	エンド・オブ・インタラプト信号出力端子 アクティブ“L”。RETI 命令 (コード ED 4D) を再フェッチする際出力する信号です。割り込みコントローラに接続されています。
LA [15:0]	O	論理アドレス出力。KC82 内部の MMU により物理アドレス A[19:0]に拡張されて出力されます。

### 5.4 レジスタ構成

#### 5.4.1 専用レジスタ

##### プログラム・カウンタ (PC)

次に実行すべきプログラムのアドレス情報 16 ビットを保持するレジスタです。この PC で示されるアドレスから命令をフェッチします。

##### スタック・ポインタ (SP)

RAM 上のスタック領域のその時点での先頭アドレス情報 16 ビットを保持するレジスタです。

##### インデックス・レジスタ (IX, IY)

インデックス・アドレッシングモード用として用いられる 16 ビットのベースアドレスを保持するレジスタで、IX レジスタ、IY レジスタの 2 つがあります。

##### インタラプトページ・アドレス・レジスタ (I)

割り込みモード 2 において、間接サブルーチン・ジャンプする為の間接アドレスの上位 8 ビットの値を保持するレジスタです。

##### メモリ・リフレッシュ・レジスタ (R)

8 ビットレジスタで、その内容は命令フェッチごとに自動的にインクリメントされます。ただし最上位ビットはインクリメントされません。

#### アキュムレータ (A, A') とフラグ・レジスタ (F, F')

2 組の 8 ビットアキュムレータ (A, A') と、各々対になった 2 組のフラグ・レジスタがあります。アキュムレータは 8 ビットの算術・論理演算の結果を保持するレジスタです。フラグ・レジスタは 8 ビットまたは 16 ビットの演算結果の状態を保持するレジスタです。A, F と A', F' の交換は EX AF, AF' で行ないます。

#### 汎用レジスタ

2 組の汎用レジスタ群があり、それぞれ単独で 8 ビットレジスタ (B, C, D, E, H, L または B', C', D', E', H', L') として使用可能で、またこれらは 16 ビットのレジスタペア (BC, DE, HL または BC', DE', HL') としても使用できます。

B, C, D, E, H, L と B', C', D', E', H', L' の交換は EXX 命令で行ないます。

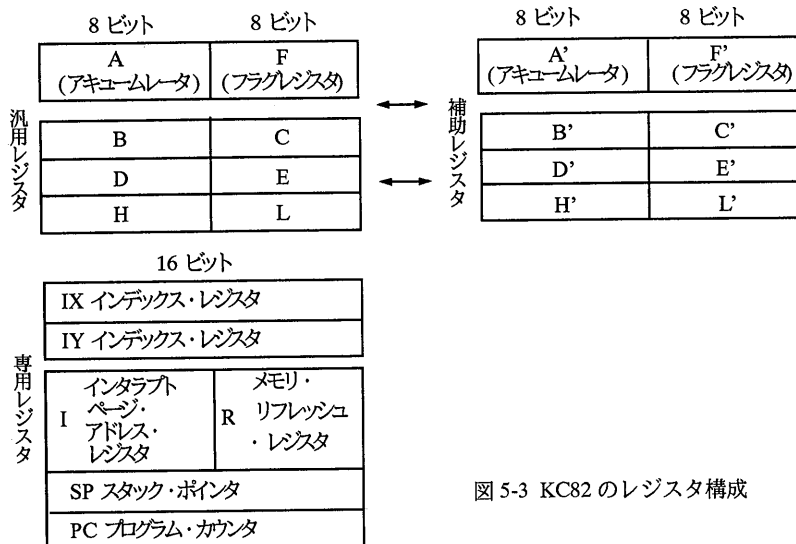


図 5-3 KC82 のレジスタ構成

### 5.5 フラグについて

フラグ・レジスタのビット構成

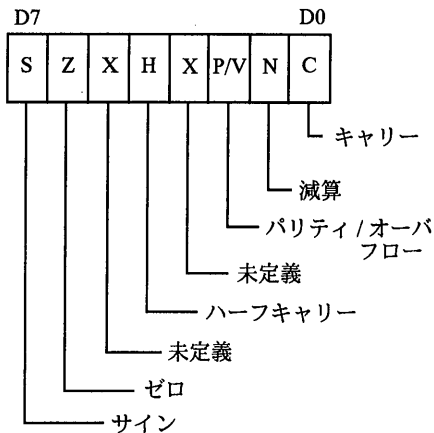


図 5-4 フラグレジスタ

フラグ・レジスタの各ビットの機能を以下に示します。

#### キャリー・フラグ (C)

アキュムレータの最上位ビットからの桁上がりでセットされます。(加算命令、減算命令、シフト・ローテイト命令等)

#### ゼロ・フラグ (Z)

加算/減算/論理演算/ INC, DEC / DAA において演算結果がゼロである時にセットされます。またブロック・サーチ命令時の検索データ検出時、ブロック入出力命令時、ローテイト/シフトおよびストリングサーチ、ビットテスト命令等にも用いられます。

#### サイン・フラグ (S)

符号付数値による演算で、演算結果が負ならばセットされます。

#### パリティ・オーバーフロー・フラグ (P/V)

このフラグは 2 つの機能を持ち、論理演算時にはパリティを示し、算術演算ではオーバーフローが発生したことを示します。

オーバーフローでは、アキュムレータ内の 2 の補数値が+127 を超えるか、-128 より小さくなった時にセットされます。またブロック・サーチ命令、ブロック入出力命令、“LD A, I 命令”、“LD A, R 命令” 実行時にも用いられます。

#### ハーフ・キャリー・フラグ (H)

BCD 演算の結果、下位 4 ビットからのキャリー、ボローを示します。

#### サブトラクト・フラグ (N)

減算命令の場合、セットされます。DAA 命令実行時、CPU はこのフラグを見て、先に実行された命令が加算か減算かを自動判定して処理を行います。

### 5.6 機能説明とタイミング

このセクションでは機能説明とタイミングについて説明します。タイミング図に出てくるアドレスバスは説明を簡単にするため MMU の影響を受けない論理アドレス LA[15:0] を使って説明しています。ご注意ください。論理アドレス LA[15:0] は MMU により、物理アドレス A[19:0] に変換されます。またバスサイクルはすべて内部バスサイクルを使って説明しています。外部バスサイクルについては 4 章を参照下さい。

#### 5.6.1 基本動作 (インストラクション・サイクル)

KC82 が命令を実行する過程は、以下の五つの各過程 (マシン・サイクル) の組み合わせにより行います。

- 1) 命令のオペコードをメモリから読み出す (フェッチする) 過程 --- オペコード・フェッチ・サイクル
- 2) メモリからデータを読み出す過程 --- メモリ・リード・サイクル
- 3) メモリにデータを書き込む過程 --- メモリ・

ライト・サイクル

4) I/O デバイスからデータを読み出す過程 --- I/O リード・サイクル

5) I/O デバイスにデータを書き込む過程 --- I/O ライト・サイクル

以下に各マシン・サイクルごとのタイミングを説明します。

#### オペコード・フェッチ・サイクル

図 5-5 に示すとおり、WAIT\_ の入力がない場合は、1 バイト 1 クロックでフェッチします。また MRD\_ の他に M1\_ がアクティブになります。メモリ・リード・サイクルとの違いは、M1\_ が出力されるかされないかの違いだけです。また、オペコードフェッチサイクルは命令シーケンスの最後のバスサイクルで実行されます。尚、最小 1 クロックサイクルです。

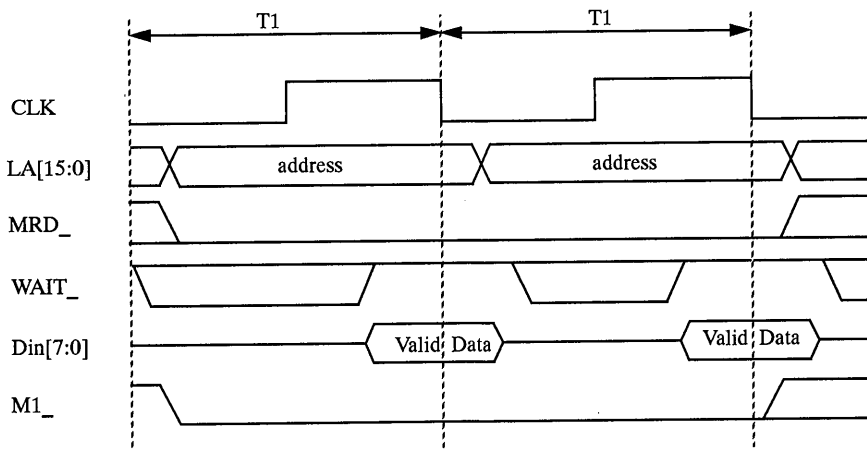


図 5-5 オペコードフェッチサイクル (内部バス・サイクル)  
(上図は、0WAIT の連続リードを表す)

### 内部バス・メモリ・リード・サイクル

オペコード・フェッチ・サイクルとの違いは、  
M1\_がアクティブにならない点です。最小1ク  
ロックサイクルです。

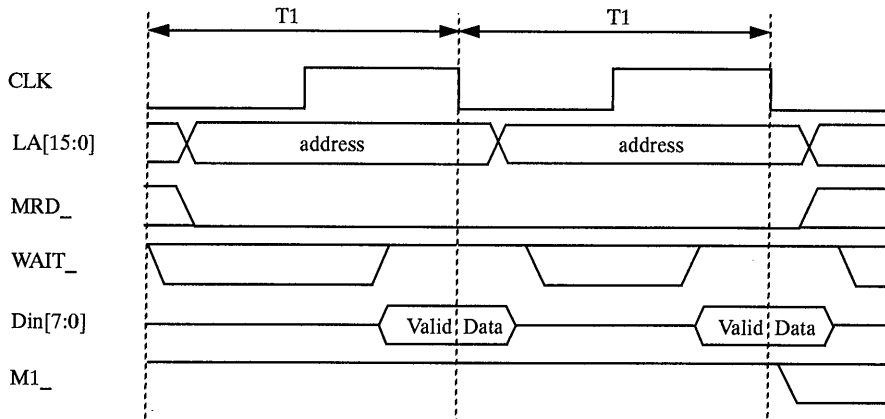


図 5-6 メモリリードサイクル (0 wait)  
(上図は、連続の 0 wait リード動作のタイミング)

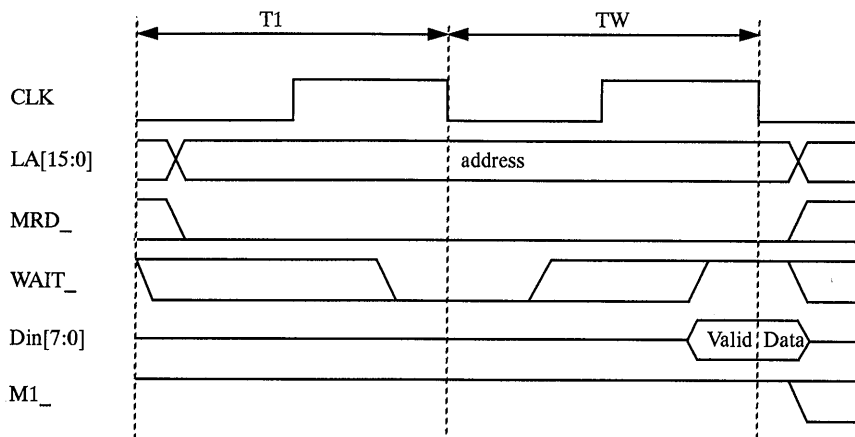


図 5-7 メモリリードサイクル (1 wait)  
(上図は、1 wait リード動作のタイミング)

### 内部バス・メモリ・ライト・サイクル

最小1クロックサイクルです。

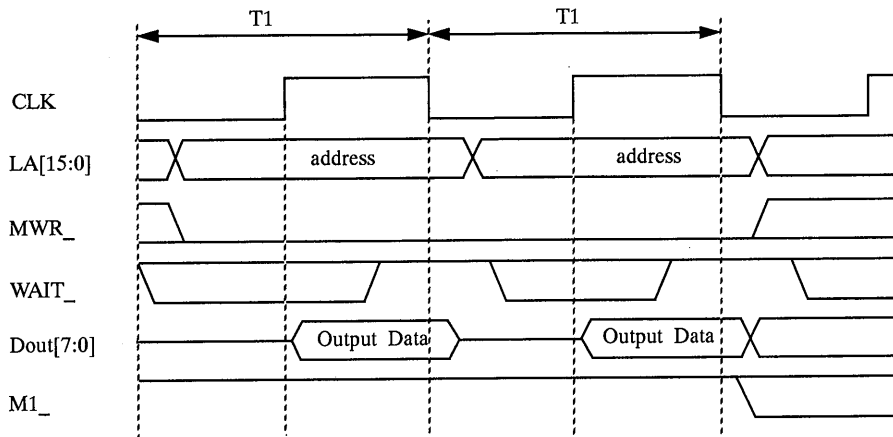


図 5-8 メモリライトサイクル (0 wait)  
(上図は、連続 0 wait ライト動作のタイミング)

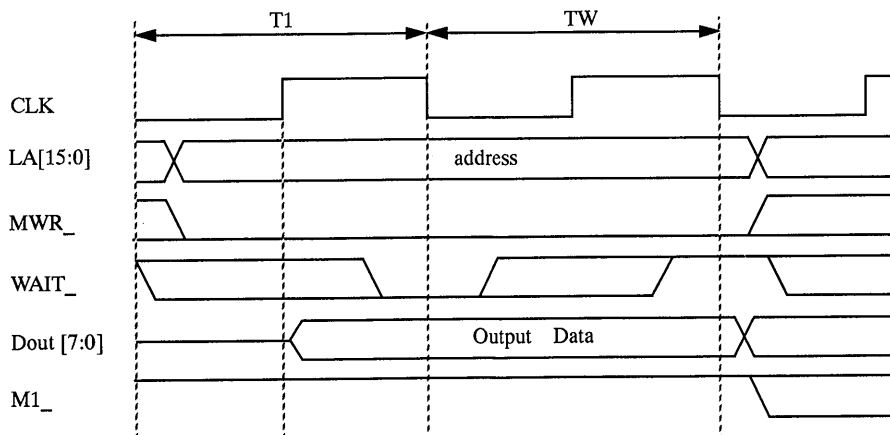


図 5-9 メモリライトサイクル (1 wait)  
(上図は、1 wait ライト動作のタイミング)

内部バス・I/O リード・サイクル  
 最小2クロックサイクルです。

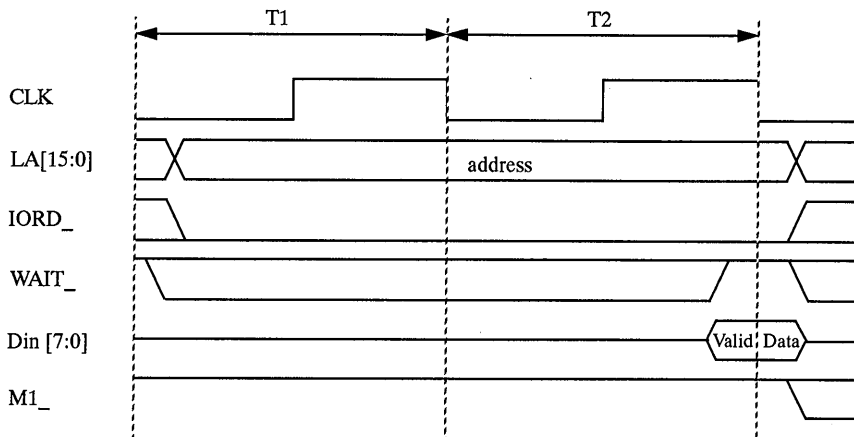


図 5-10 I/O リードサイクル (0 wait)  
 (上図は、0 wait 動作のタイミング)

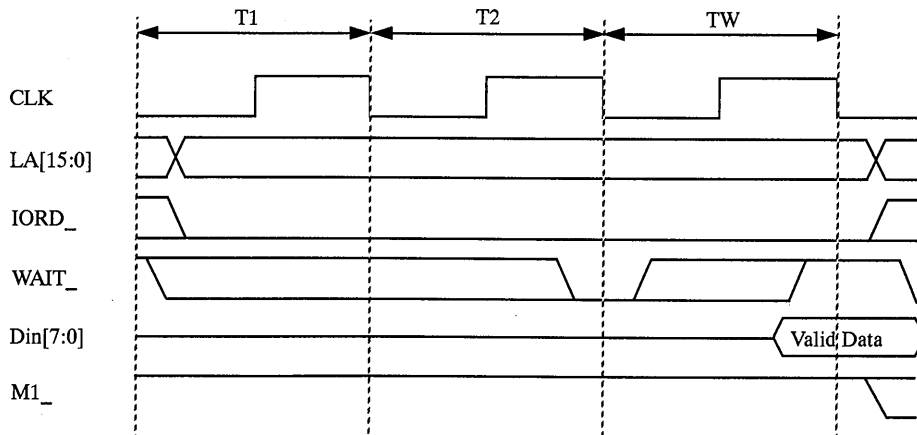


図 5-11 I/O リードサイクル (1 wait)  
 (上図は、1 wait 動作のタイミング)

内部バス・I/O ライト・サイクル  
 最小2クロックサイクルです。

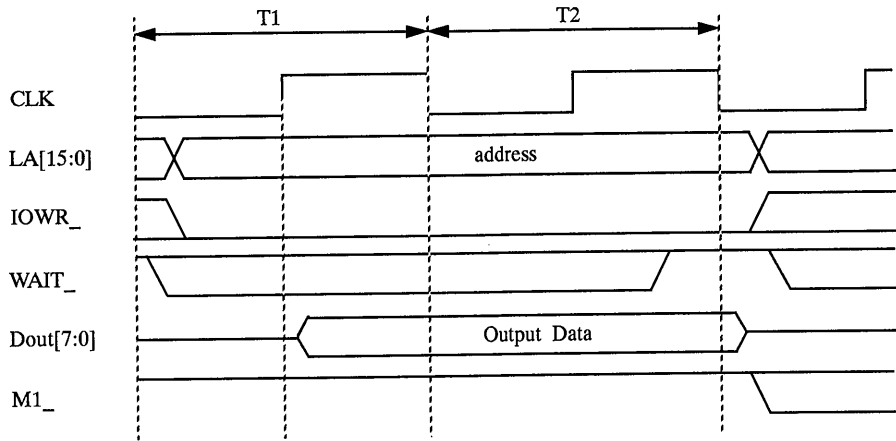


図 5-12 I/O ライトサイクル (0 wait)  
 (上図は、0 wait 動作タイミング)

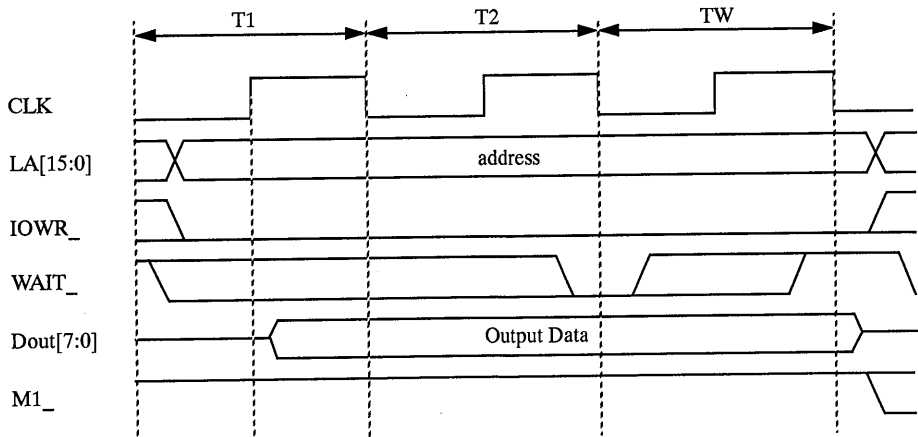


図 5-13 I/O ライトサイクル (1 wait)  
 (上図は、1 wait 動作タイミング)

### 命令プリフェッチサイクル

KC82 の命令サイクルの終りには、必ずプリフェッチサイクルが存在します。図 5-15 に LD A,[1234H] ⇒ ADD A,D ⇒ JP 1000H ⇒ LD [HL], A 命令を実行するタイムチャートを示します。また、図 5-14 には図 5-15 に対応したアセンブラコーディングを示します。図中の ◀▶ (以下“矢印”と呼びます) がプリフェッチサイクルを表します。①の矢印は n+3 番地の '82H' (ADD A,D 命令)を、②の矢印は n+4 番地の 'C3H' (JP xxxx 命令)を、③の矢印は 1000H 番地の '77H' (LD [HL], A 命令)をそれぞれプリフェッチするサイクルで、図でもわかるように次に実行しようとする命令の直前 (現在実行中の命令の終り)にプリフェッチされることがわかります。

address	mnemonic	code
n	LD A, [1234H]	3A 34 12
n+3	ADD A, D	82
n+4	JP 1000H	C3 00 10
.	.	.
1000H	LD [HL], A	77
.	.	.
1234H	.	5A

図 5-14 図 5-15 のアセンブラコーディング

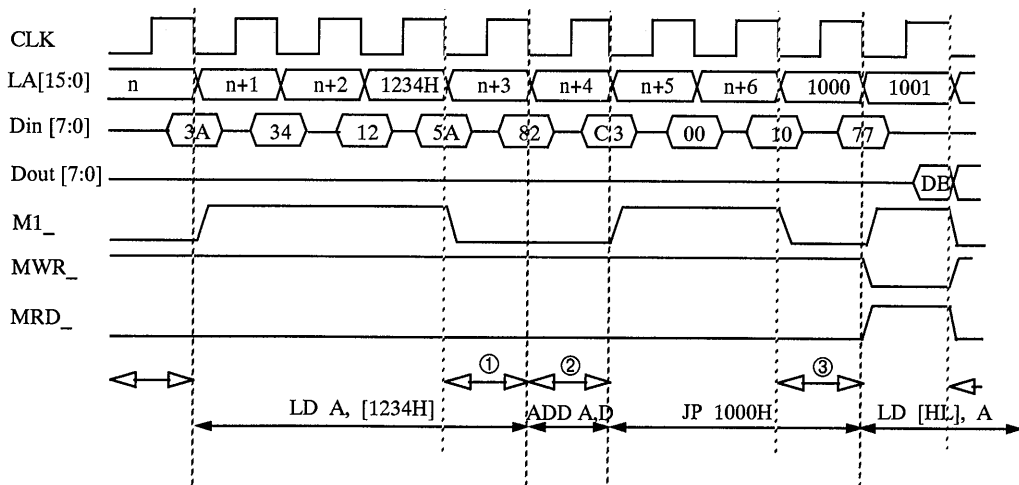


図 5-15 プリフェッチの様子 (◀▶ がプリフェッチサイクルを表す。また 1000H 番地、1234H 番地にはそれぞれ '77H', '5AH' が格納されており、D レジスタは '81H' とする)

### 特殊な場合のプリフェッチ (プリフェッチの無効化)

プリフェッチサイクルは例えば図 5-1 の

JP 1000H 命令のように③でプリフェッチしたデータ '77H' は内部で保持してデコードされ実行されますが、割り込み系に関してはプリフェッチされたデータは保持せずに、廃棄され割り込みルーチンから復帰後に再び廃棄されたデータのフェッチを開始します。つまり、割り込み認識処理で PC をスタックに PUSH する時に PC をマイナス 1 してセーブし、RETI や RETN 命令での POP 動作で再セットされる PC、すなわち廃棄されたデータの格納されているアドレスからプリフェッチをします。図 5-19 に LDIR 命令実行中にモード 1 割り込みが発生したタイミング図を示します。図中の⑥でプリフェッチされたデータ 'ED' は廃棄され割り込み認識処理へ進みます。そして、割り込み復帰後再びこの 'ED' のプリフェッチから開始されます。なお、バスリクエストサイクルでは通常命令と同じく直前にプリフェッチされたデータは保持されバス解放時に next アドレス (ジャンプ命令時はジャンプ先のアドレス) から処理を開始します。図 5-18 に LDIR 命令実行中にバスリクエスト要求が発生した場合のタイミング図を示します。図中の⑥でプリフェッチされたデータ 'ED' は廃棄されずに保持されバス解放時に 'ED' の次のアドレスからプリフェッチがされていることがわかります。

### 5.6.2 バス解放 (バス・リクエスト/アクノレッジ・サイクル)

通常 CPU がアドレス・バス、コントロール・バス (MRD<sub>0</sub>, MWR<sub>0</sub>, IORD<sub>0</sub>, IOWR<sub>0</sub>) の制御権を握っていますが、外部からバス要求 (BREQ = "L") があると、CPU はアドレスバスをハイ・インピーダンス状態にして、全ての割り込みを禁止し、BACK<sub>0</sub> 信号を出力して ("L")、外部にバスの制御権を譲ります。この機能により、CPU を介さないデータ転送が可能となります。図 5-17 にバスリクエスト基本タイムチャートを示します。このサイクルは、BREQ<sub>0</sub> が "L" の期間ずっと継続します。バスリクエストの受け付けタイミングについては、ブロック転送命令、ブロックサーチ命令、入出力命令

のうちくりかえし動作のある命令を除いて命令の切れ目です。特殊な例としては、図 5-18 に示すように LDIR 命令実行中の④でリクエストが発生していますが、LDIR 命令の最後のバスサイクルである⑤で受け付けられます。注意するのは、各バスサイクルの切れ目ではなく命令サイクルの切れ目であることです。

### 5.6.3 割り込み機能とタイミング

KC82 で受け付けられる割り込み要求には、以下の 2 通りがあります。

- 1) INT<sub>0</sub> によるマスク可能な割り込み (マスカブル割り込み)
  - 2) NMI<sub>0</sub> によるマスク不可能な割り込み (ノン・マスカブル割り込み)
- 2) は、1) よりも優先順位が高く、1)、2) 同時に発生した場合は、2) の方が先に受け付けられます。

### マスカブル割り込み

マスカブル割り込みは EI 命令によって割り込みを許可され、DI 命令によって割り込みを禁止されます。この制御は 2 つのイネーブル・フリップフロップ (IFF1, IFF2) によって制御されています。下図はそのフリップフロップの状態遷移とその要因を示したものです。

要因	IFF1	IFF2	
リセット	0	0	
DI 命令	0	0	
EI 命令	1	1	
INT 受け付け	0	0	
NMI 受け付け	0	-	
RETN 命令;受け付け	IFF2	-	IFF1 に IFF2 がコピー
LD A,I 命令	-	-	P/V に IFF2 がコピー
LD A,R 命令	-	-	P/V に IFF2 がコピー

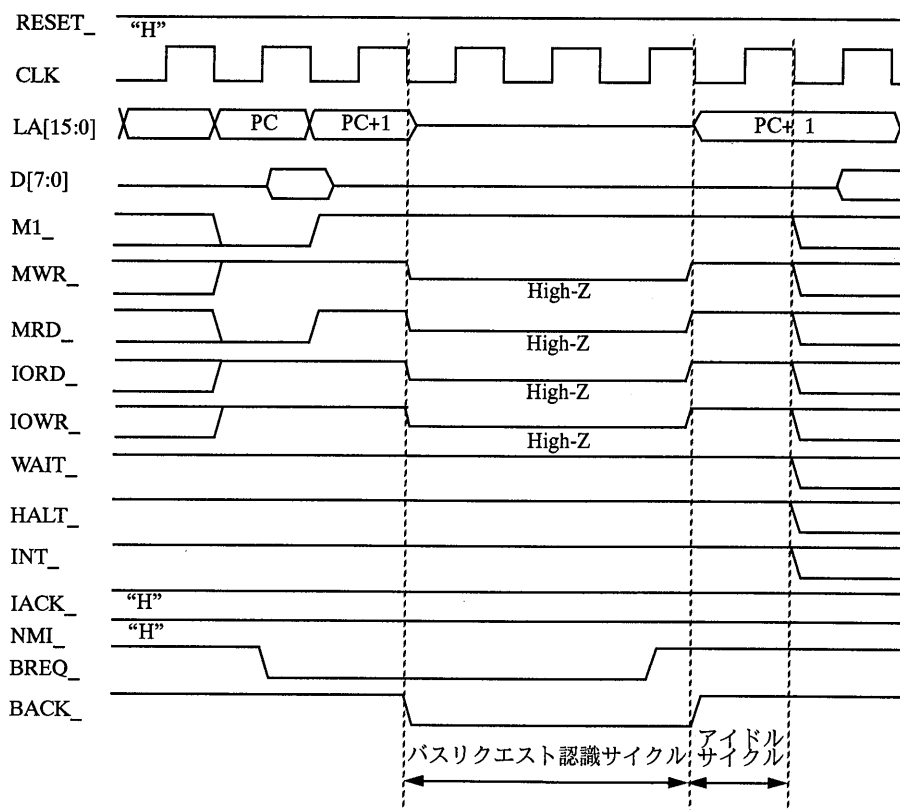


図 5-17 バスリクエストタイミング

マスカブル割り込みが有効なのは、以下の条件を全て満たしている時です。

- 1) 2つのフリップフロップが共にセットされている。(EI 命令や RETN 命令後は、さらに次の 1 命令実行後に初めて有効になります。従って、EI 命令と DI 命令が連続した場合には、割り込みは受け付けられません。)
- 2) BREQ\_ がインアクティブ ("H") になっている。(BREQ\_ が "L" から "H" に変わった直後では、1 命令実行後に初めて有効になります。)
- 3) NMI\_ の立ち下がりエッジを保存していない。

従来の Z80 では M1\_ と IORQ\_ とで外部に割り込み許可を認識させていましたが、KC82 では独立した IACK\_ 端子によってこれを実現しています。

### マスカブル割り込みのモード

マスカブル割り込みには 3 種類のモードがあり、それぞれ処理が異なります。以下に各モードでの動作を説明しますが通常はモード 2 を使って下さい。

#### (1) モード 0

リセット時には、自動的にこのモードにセットされます。また他のモード実行中は IM 0 命令によってこのモードに移ります。このモードは、割り込みサイクル中に読み込んだ命令をそのまま実行します。通常この割り込みで使う命令は、RST 00H-RST 38H、または CALL 命令です。RST 命令、CALL 命令を読み込んだ場合のタイミング図をそれぞれ図 5-20、図 5-21 に示します。なおモード 0, 1, 2 共通して、IM X(X = 0,1,2) 命令実行中に INT\_ が入力された場合、IM X 命令実行直後からモードの変更が有効になります。

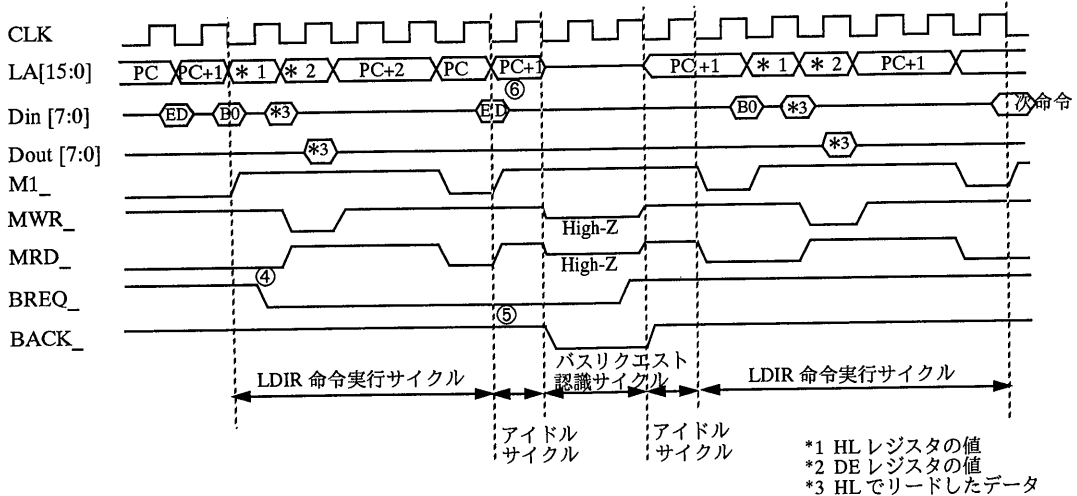


図 5-18 バスリクエスト受け付けタイミング

(上図では、LDIR 命令実行中に BREQ\_入力が④で発生し、命令の切れ目である⑤で受け付ける。尚、⑥でフェッチしたデータは保持される。)

### (2) モード 1

IM 1 命令によってこのモードに移ります。このモードでは、PC(プログラム・カウンタ)の内容をスタックに退避後、割り込みサイクル中に読み込んだデータを無視して、強制的に RST 38H 命令を実行するものです。モード 1 の割り込み認識シーケンスのタイミング図を図 5-22 に示します。

### (3) モード 2

IM 2 命令によってこのモードに移ります。このモードでは、図 5-23-A のように割り込みサイクル中に読み込んだ割り込みベクタを下位 8 ビット (ただし、最下位ビットは '0') とし、I レジスタの内容を上位 8 ビットとする 16 ビットアドレスで割り込みプログラムのスタートアドレステーブルをアクセスします。そして、読み込んだスタートアドレスから始まる割り込みプログラムを実行します。モード 2 の割り込み認識シーケンスのタイミング図を図 5-23-B に示します。

### 割り込みの受け付けタイミング

図 5-19 に割り込み受け付けのタイミングを示します。図では、LDIR 命令実行中に INT が発生した例

ですが、④で INT 入力が発生し、LDIR 命令の最後のプリフェッチサイクルである⑤で初めて受け付けられます。この時、⑥でフェッチしたデータ 'ED' は CPU 内部にはとこまれず割り込み復帰後再フェッチされます。

### ノン・マスカブル割り込み

ノン・マスカブル割り込みはソフトウェアでマスクできない割り込みです。NMI\_の立ち下がりエッジの検出は各命令の最後のステートのクロックの立ち下がりで行ないます。この時点より前に NMI\_の立ち下がりエッジがあれば、割り込みを受け付けます。ノン・マスカブル割り込みは BREQ\_ がインアクティブ ("H") になっている時に有効になります。(BREQ\_ が "L" から "H" に変わった直後では、1 命令実行後に初めて有効になります。)

ノン・マスカブル割り込みを認識すると、PC の内容をスタックにセーブして、0066H 番地へサブルーチンコールします。この割り込みルーチンからの復帰は、RETN 命令の実行により実現します。NMI\_認識シーケンスのタイミング図を図 5-24 に示します。

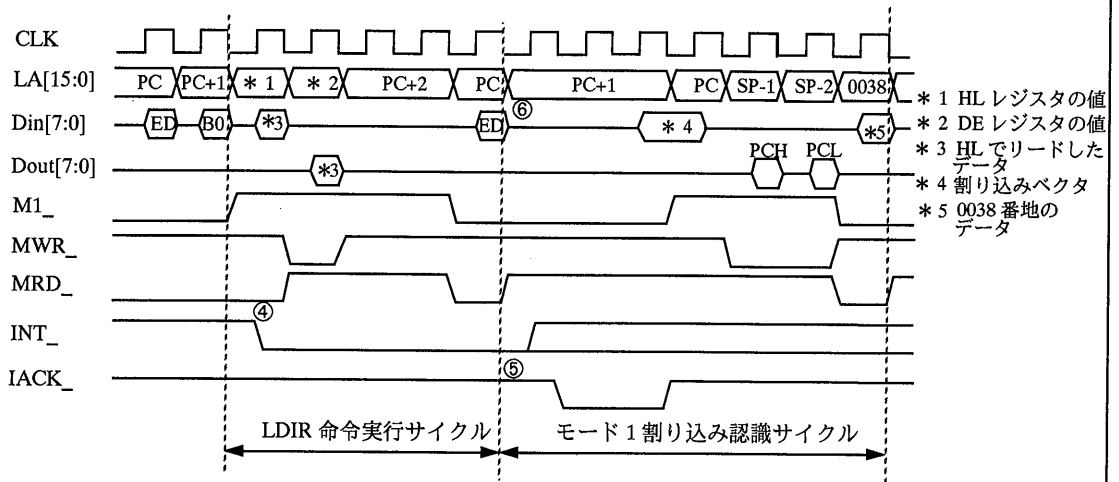


図 5-19 割り込み受け付けタイミング (モード 1)

(上図では、LDIR 命令実行中に INT\_ 入力力が④で発生し、命令の切れ目である⑤で割り込みを受け付ける。割り込みリターン後は⑥の ED データのフェッチから再開するケースを想定)

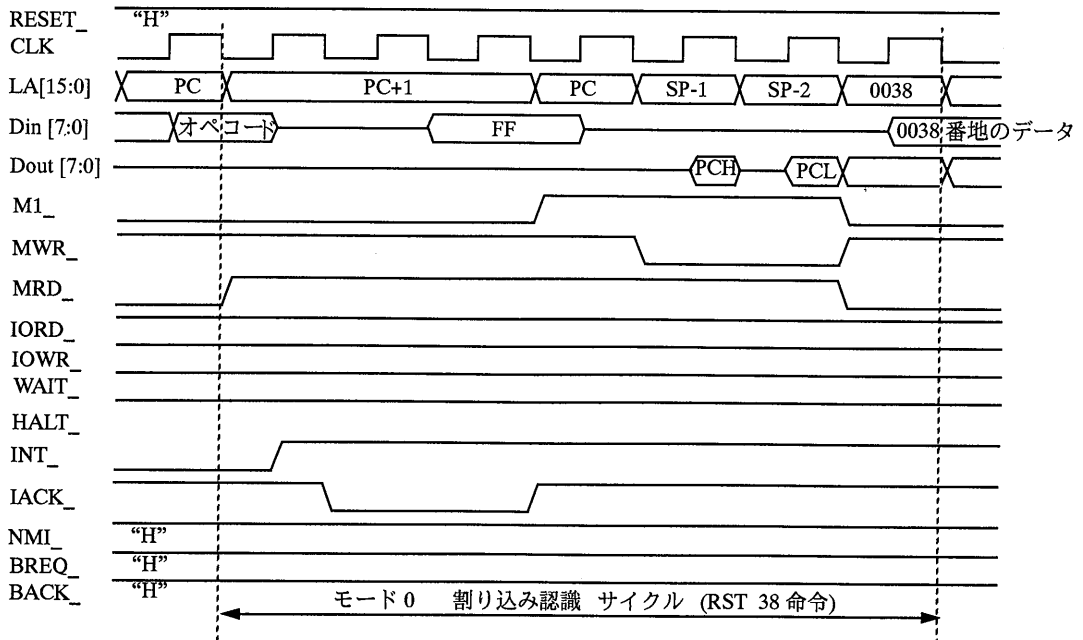


図 5-20 モード 0 タイミング

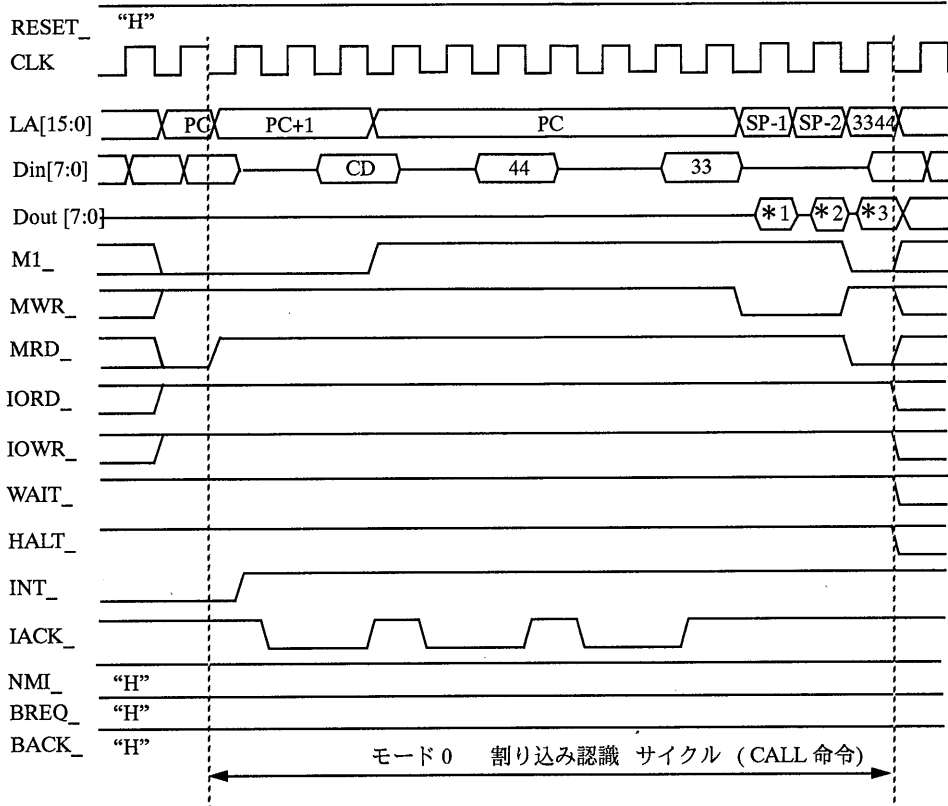


図 5-21 モード0 タイミング

- \* 1 PCH
- \* 2 PCL
- \* 3 3344 番地のデータ

### 割り込みのイネーブルおよびディスエーブルについて

割り込みのイネーブルおよびディスエーブルは内部の IFF1 および IFF2 というフリップフロップで制御されます。図 5-16 に 2 つのフリップフロップの動作の状態を表に示します。マスク可能割り込みが受け付けられるのは、IFF1、IFF2 がともに '1' の状態です。つまり、リセット直後および DI 命令実行直後は '0' でありマスク可能割り込みは受け付けられず、また EI 命令で両フリップフロップが '1' となり割り込みを受け付けられる状態となります。フリップフロップが 2 つあるのはノンマスク可能割り込みが発生したときのイネーブル/ディス

エーブル状態を IFF2 に記憶するためです。例えば、NMI が発生する直前の IFF の値が IFF1='1'、IFF2='1' (イネーブル状態) とすると NMI の発生により IFF1='0' となり INT の受け付けはできなくなります。この時 IFF2 には '1' が保存されています。そして、RETN 命令により IFF2 の値つまり '1' が IFF1 にコピーされ両フリップフロップとも '1' となりイネーブル状態に復帰出来るわけです。

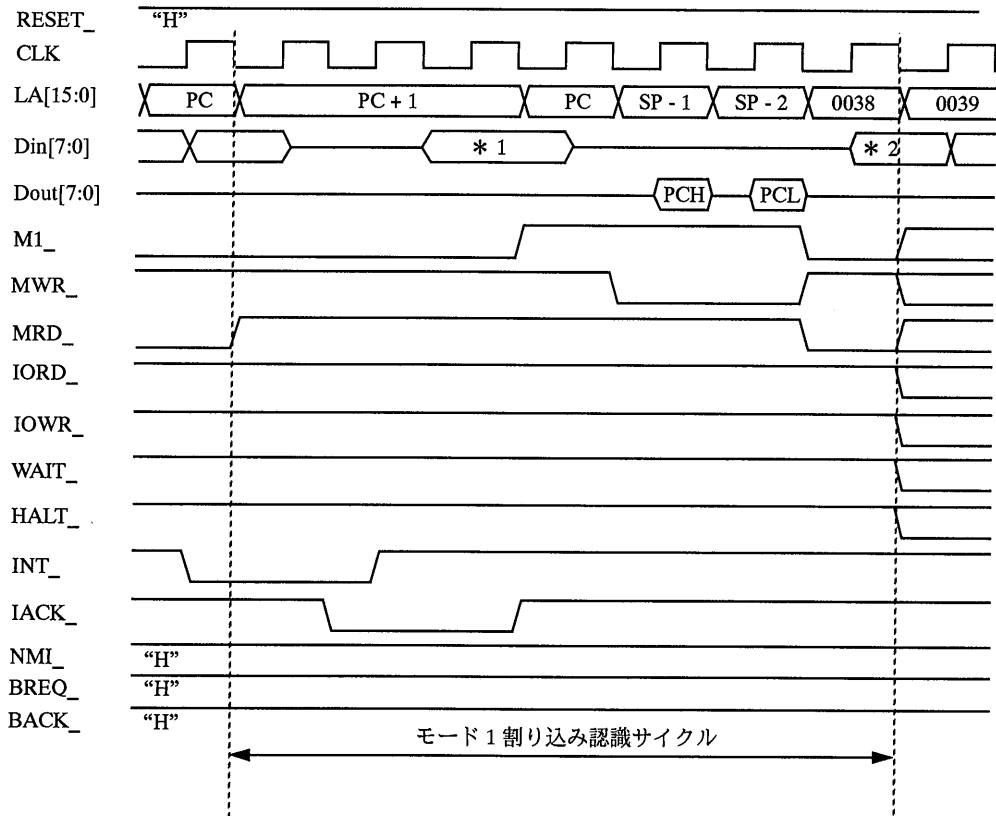


図 5-22 モード 1 タイミング

\* 1 割り込みベクトル  
\* 2 0038 番地のデータ

### 5.6.4 HALT の実行と解除

HALT 命令により CPU 内部では、NOP 命令を実行し続けますが、リセット入力や割り込み入力(ノン・マスクブル割り込みまたは、割り込み許可フラグがセットされている状態でのマスクブル割り込み)があれば、HALT 状態は解除されます。図 5-25 は HALT 命令実行中にモード 2 の割り込みが発生し HALT が解除されるタイミング図です。HALT は 2 つのバスサイクルで構成され第 1 サイクルはアイドルサイクルで、第 2 サイクルがオペコードフェッチサイクルです。HALT 実行中のアドレスは HALT 命令が置かれているアドレスの次のアドレスを出力します。この時第 2 サイクルでフェッチしたデータは KC82 内部には取り込まれません。そして、INT\_入

力により HALT 命令の第 2 サイクルのクロックの立ち下がりで割り込み (図ではモード 2) を受け付けるのと同時に HALT\_="H" となります。

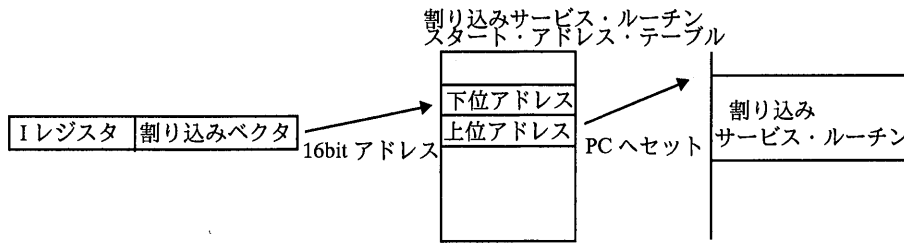


図 5-23-A モード 2 割り込み

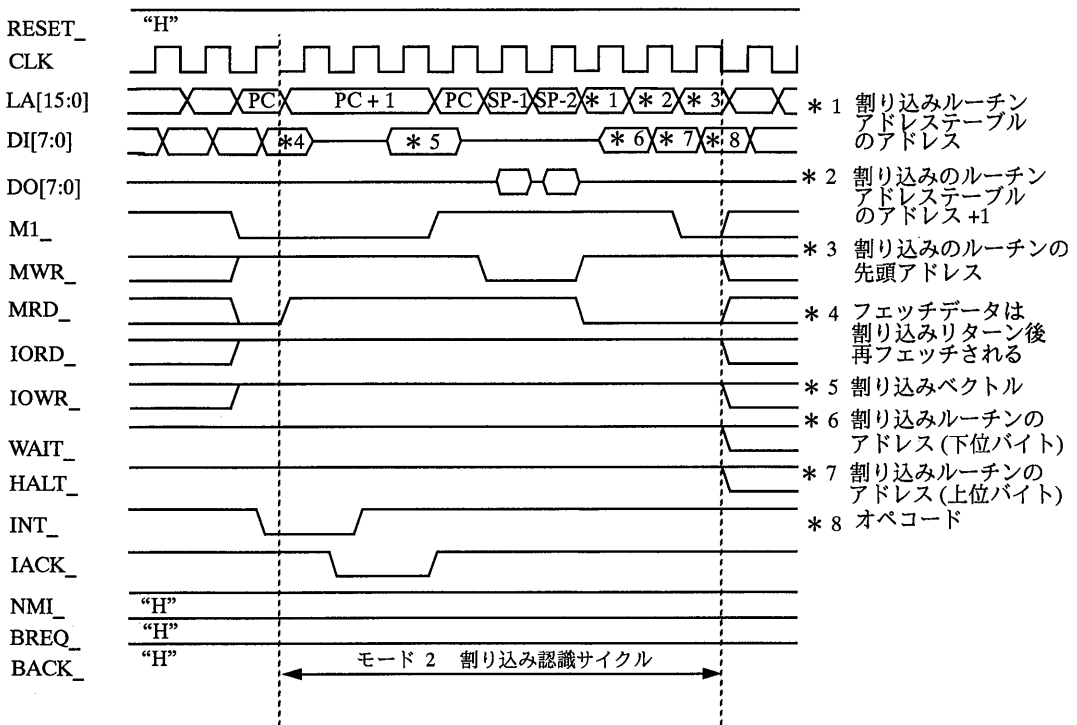
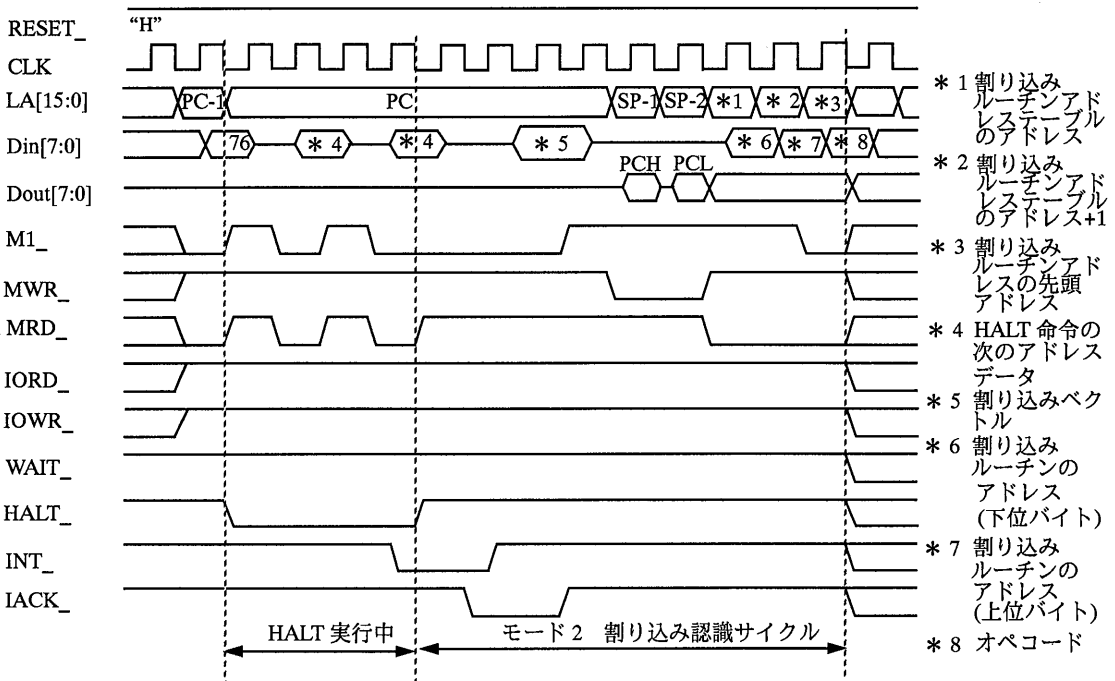
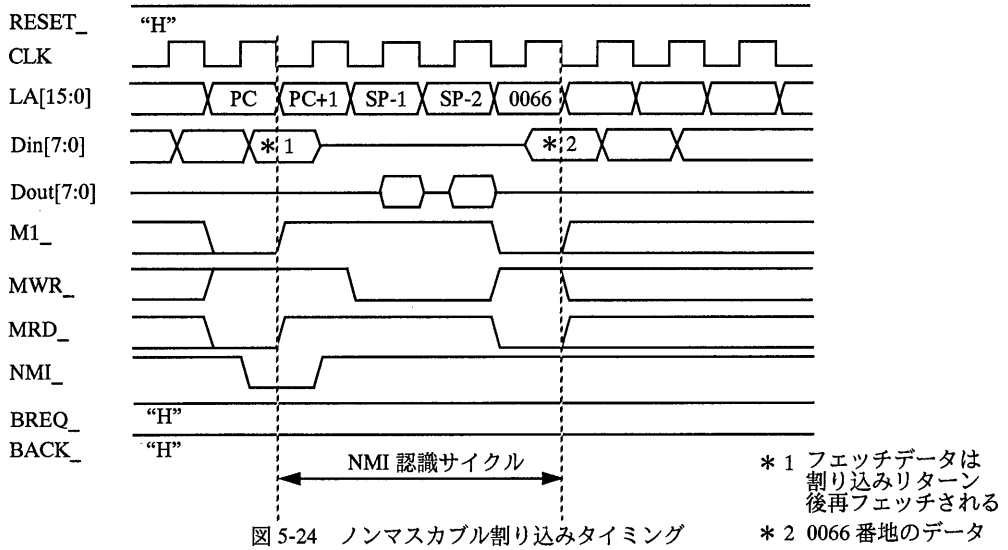


図 5-23-B モード 2 タイミング

### 5.6.5 リセット・タイミング

RESET\_入力を最小 3 クロック期間アクティブ (“L”) にすることによって、CPU はリセットされます。リセット期間中はアドレスバスはハイ・インピーダンス状態となります。RESET\_入力がインアクティブ (“H”) になると、3 クロック目の立ち下がりからオペコード・フェッチ・サイクルが開始さ

れ、0000H 番地の命令から順次実行を開始します。また、割り込みモードは 0 にセットされ、割り込みフラグ IFF1, IFF2 はリセットされます。I レジスタ、R レジスタは ‘00H’ にリセットされます。



### 5.7 MMU

#### 5.7.1 概要

MMU ブロックは KC82 の 16 ビットの論理アドレス LA[15:0] を 20 ビットの物理アドレス A[19:0] に変換する回路です。ただし、MMU がアドレスを変換するのはメモリアクセスの場合だけで、I/O アクセスの場合はアドレス空間に何の変換も加えません。MMU は下

のレジスタと物理アドレス生成回路から構成されています。物理アドレス生成回路の動作は後のセクションで説明しています。

表 5-1 レジスタ構成

名称	ビット数	Read / Write
境界/ ベースレジスタ 1 (BBR1)	8 ビット	R/W
" 2 (BBR2)	"	R/W
" 3 (BBR3)	"	R/W
" 4 (BBR4)	8 ビット*	R/W
ベースレジスタ 1 (BR1)	8 ビット	R/W
" 2 (BR2)	"	R/W
" 3 (BR3)	"	R/W
" 4 (BR4)	"	Read Only

\*上位 2 ビットは Read Only

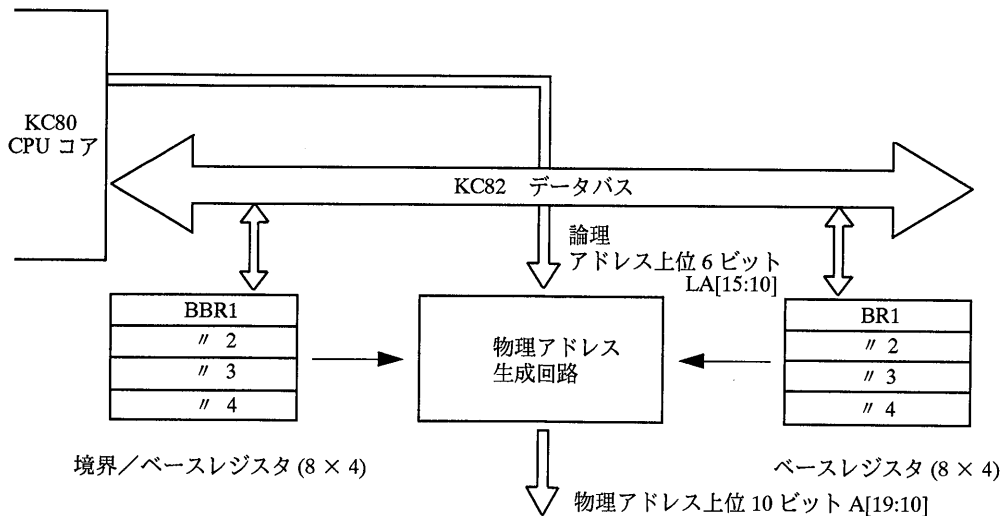


図 5-26 ブロック図

### 5.7.2 各レジスタの説明

MMUブロックには8ビットのレジスタが8本含まれています。この8本のレジスタでMMUブロックの動作に必要な4つの10ビットデータと4つの6ビットデータを保持します。

#### 境界/ベースレジスタ(BBR1~BBR4)

本レジスタの上位2ビット(A1<1:0>~A4<1:0>)はベースレジスタの8ビットとともに物理アドレスのベースアドレス指定に使われます。下位6ビットは論理アドレス境界値(B1<5:0>~B4<5:0>)で、論理空間の領域分割の境界値として使われます。またBBR4の上位2ビットは固定値であり、データを書き込んでもその内容は変化しません。

#### ベースレジスタ(BR1~BR4)

本レジスタは境界/ベースレジスタの上位2ビットとともに10ビットの物理アドレスベース(A1<9:0>~A4<9:0>)を構成し、物理アドレスのベースアドレス指定に使われます。この4本のベースレジスタ(BR1~BR4)のうち、BR4はF0H固定で書き込みできません。

#### 論理アドレス境界値と物理アドレスベース

MMUブロックは論理アドレス空間を5つの領域に分割しており、それぞれを物理アドレス空間にマッピングしています。このためMMUブロックは5つの論理アドレス境界値(B0~B4)と5つの物理アドレスベース(A0~A4)を必要とします。このうちA0とB0は固定でそれぞれA0=000H B0=00Hです。残りのデータは図5-27にあるように境界/ベースレジスタとベースレジスタに割りつけられています。

I/Oアドレス レジスタ名

		bit7	6	5	0
00H	BBR1	A1<1:0>		B1<5:0>	
01H	BR1	A1<9:2>			
02H	BBR2	A2<1:0>		B2<5:0>	
03H	BR2	A2<9:2>			
04H	BBR3	A3<1:0>		B3<5:0>	
05H	BR3	A3<9:2>			
06H	BBR4	A4<1:0>		00B固定	B4<5:0>
07H	BR4	A4<9:2>		F0H固定	

図 5-27 MMU ブロックのレジスタ

### 5.7.3 MMUによる物理アドレス生成

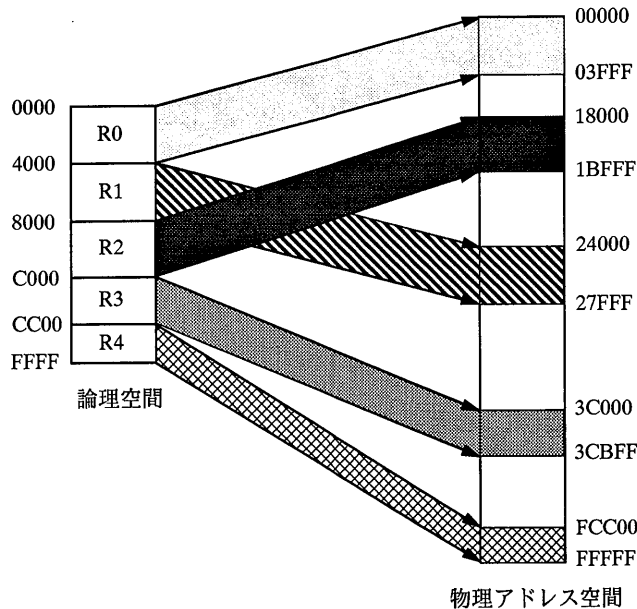
論理アドレス空間は、図 5-28 に示すように5つの領域 (R0~R4) に分割されています。この5つの論理領域は論理アドレス境界値 (B0<5:0>~ B4<5:0>) で決められています。Rn 領域の下限アドレスは (Bn+1) × 400H で決定され、Rn領域の上限アドレスは(B(n+1)+1) × 400H-1となります。(ただし、R0領域の下限アドレスは常に 00000H、R4 領域の上限アドレスは常に FFFFHです。) これらの領域の物理空間上の配置は分割されたそれぞれの領域の物理アドレスベース An(10ビット)を論理アドレスの上位6ビットと足し合わせて生成します(図 5-29 参照)。論理空間の分割は論理アドレスの上位 6 ビットと論理アドレス境界値 (B0 ~ B4) の大小比較で決定されるので、1K バイト単位で設定が可能です。なお R1 領域の下限アドレスの最小値は 0400H で A0 は固定値

000Hのため、R0 領域に属する論理空間の最初の1K バイト (0000H ~ 03FFF) は常に物理空間上の 00000H~003FFFHに割り付けられます。

参考のため設定例を図5-28に示します。例えば

- B0 = 00H (固定)
- B1 = 0FH
- B2 = 1FH
- B3 = 2FH
- B4 = 32H
- A0 = 000H (固定)
- A1 = 080H
- A2 = 040H
- A3 = 0C0H
- A4 = 3C0H (固定)

と設定した場合、5つの論理領域と物理領域の対応は以下のようになります。



領域 R0	論理アドレス 0000H ~ 3FFFH	物理アドレス 00000H ~ 03FFFH
領域 R1	論理アドレス 4000H ~ 7FFFH	物理アドレス 24000H ~ 27FFFH
領域 R2	論理アドレス 8000H ~ BFFFH	物理アドレス 18000H ~ 1BFFFH
領域 R3	論理アドレス C000H ~ CBFFFH	物理アドレス 3C000H ~ 3CBFFFH
領域 R4	論理アドレス CC00H ~ FFFFH	物理アドレス FCC00H ~ FFFFFH

図 5-28 論理アドレスと物理アドレスの対応例

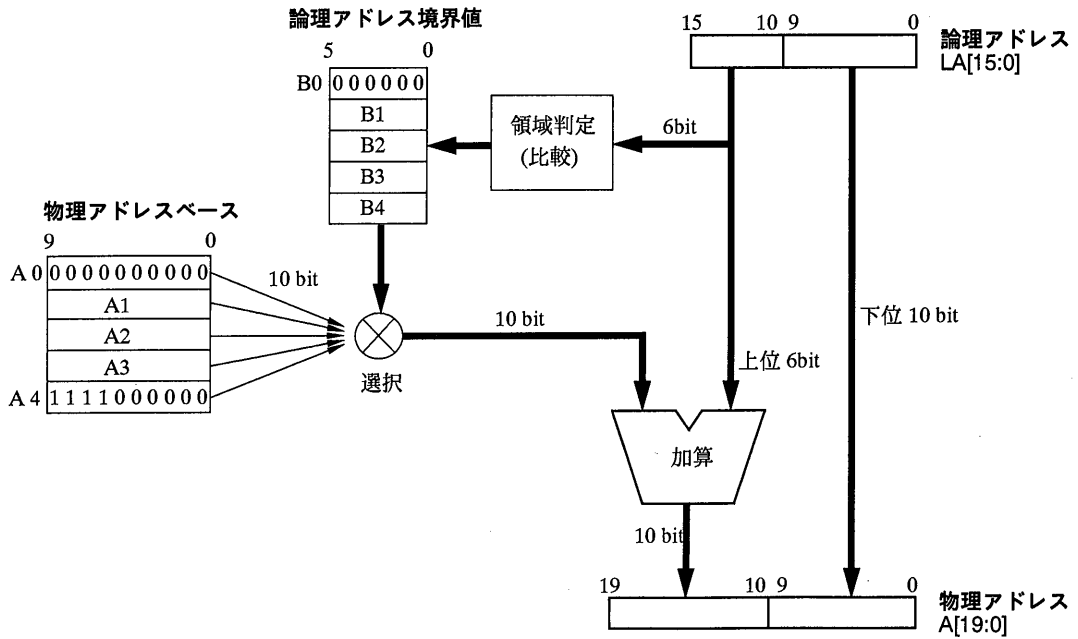


図 5-29 物理アドレス生成

### 5.7.4 MMU の動作

#### (1) メモリ空間

MMU は、CPU がメモリ空間をアクセスするときにそのアドレスを変換します。これは具体的には次のような場合です。

1. 命令のフェッチ
2. 命令によるメモリ空間へのリードあるいはライト
3. 割り込みのリスタートアドレスへのフェッチ
4. モード2割り込みにおけるスタートアドレステーブル

#### (2) I/O 空間

CPU が I/O 空間をアクセスするときには、MMU はアドレスを変換せず、論理アドレスをそのまま出力します。このとき、アドレスの上位 4 ビットには 0H が出力されます。

### 5.7.5 リセット時

リセット時には下記のように初期化されます。

- B0 = 00H (固定)
- B1 = 3FH
- B2 = 3FH
- B3 = 3FH
- B4 = 3FH
- A0 = 000H (固定)
- A1 = 000H
- A2 = 000H
- A3 = 000H
- A4 = 3C0H (固定)

このとき論理アドレス空間は R0 領域のみになり、論理アドレス空間の 64K バイトは物理アドレス空間先頭の 64K バイトにマップされます。

### 5.7.6 MMU 使用上の注意

(1) 論理アドレス境界値は、 $B0 < B1 < B2 < B3 < B4$  となるように設定したとき、すべての領域が有効になります。論理アドレス境界値の大小関係が逆転あるいは等しくなった場合には数字の大きい方が有効となり、数字の小さい方は無効となり、この論理アドレス境界値に対応する領域が消滅します。例えば、 $B1 \geq B2$  に設定したとき、R1 領域は消滅します。

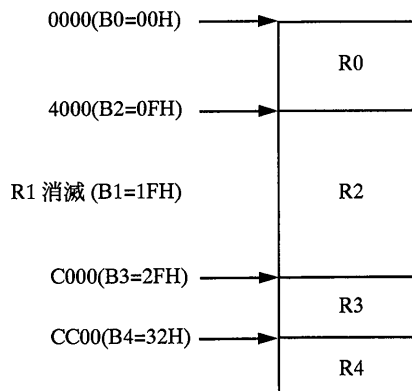


図 5-30  $B1 \geq B2$  としたときの例

(2)MMU レジスタ (境界/ベースレジスタ、ベースレジスタ)にデータを書き込んだとき、その設定が有効になるのは、MMU レジスタにデータを書き込む I/O ライトサイクルの次のバスサイクルからです。物理アドレスの割付の変更などを行なう場合は、十分注意してください。

(3) 各領域 (R0 ~ R4) の用法についてハードウェア的な制約はありませんが、当社の指定のリンク (LSI-C の knil) の制約から以下のような割り付けをおすすめします。

- R0 常駐プログラム領域 (共通プログラム、割り込み処理プログラムなど)
- R1 プログラムバンクウィンドウ
- R2 データバンクウィンドウ 1
- R3 データバンクウィンドウ 2
- R4 常駐データ領域 (スタックなど)

(4) 本 MMU 回路は下記の I/O アドレスを占有します。この I/O アドレスにはユーザ用の I/O は割り付けられないので御注意ください。

表 5-2 MMU 回路が使用する I/O マッピング

I/O アドレス	レジスタ名
00H	BBR1
01H	BR1
02H	BBR2
03H	BR2
04H	BBR3
05H	BR3
06H	BBR4
07H	BR4
08H ~ 0FH	川崎製鉄予約

(5) R1 の論理アドレス境界は R1 論理アドレス境界値 (B1) で決められており、R1 領域の下限アドレス

は  $(B1+1) \times 400H$  となります。このため、R1 領域の下限アドレスの下限値は 0400H となります。一方、R0 領域の下限アドレスは 0000H となっており、論理アドレスの最初の 1K バイトは必ず R0 領域に属し、そのマップ先は固定されており 00000H ~ 003FFH となります。

(6)B1 ~ B4 を設定するときは 3FH 未満の値を設定してください。3FH は無効設定として扱われ、その領域は消滅します。

LSI-C は LSI ジャパン社の登録商標です。

### 6. 割り込みコントローラ

#### 6.1 概要

本チップは割り込みコントローラとしてKP69マクロセルを搭載しています。KP69は弊社CPU(KC80あるいはKC82)専用に開発された小型割り込みコントローラです。KP69はKC80あるいはKC82のモード2割り込みに対して16レベルの割り込み要求入力をサポートできます。

各割り込み要求入力は、HIGH/LOW2つの優先順位グループに分割設定でき、各グループの中では割り込み要求入力レベルのビット番号が大きいものほど優先順位は高く設定されます。また、各割り込み要求入力のエッジ/レベル動作、マスク状態、および割り込みベクタの上位3bitはプログラム可能となっています。

KP69は割り込み要求を受け付けるとマスク状態、優先順位を判定し、CPUへINT\_信号を出力します。そしてCPUからIACK\_信号が返ってくると、プログラムされた割り込みベクタをデータバスに出力します。また、CPUのEOI\_信号="L"によって割り込み処理終了を認識します。このCPUのEOI\_信号はRETI命令実行時にLになります。したがって割り込み処理ルーチンの最後にRETI命令を置くことにより、KP69は自動的にRETI命令を認識して割り込み処理終了を認識します。さらに不正割り込み検出機能を備えています。

#### 特徴

- ・クロック同期式の割り込みコントローラです。
- ・KC82のモード2割り込み対応です。
- ・16レベルの割り込み要求入力の、優先順位が制御できます。
- ・割り込み入力は個々にマスク可能です。
- ・多重割り込み処理が可能です。
- ・割り込み要求入力のエッジ/レベル切替が可能です。
- ・不正割り込み検出機能を有します。
- ・CPUのRETI命令実行を検出し、割り込み処理終了を自動的に検出できます。

### 6.2 ブロック図

KP69の全体ブロック図を以下に示します。KP69のIACK\_入力はCPUのIACK\_出力、EOI\_入力はCPUのEOI\_出力、INT\_出力はCPUのINT\_入力にそれぞれ内部で接続されています。また、IR[15:0]には下表のような信号が接続されています。

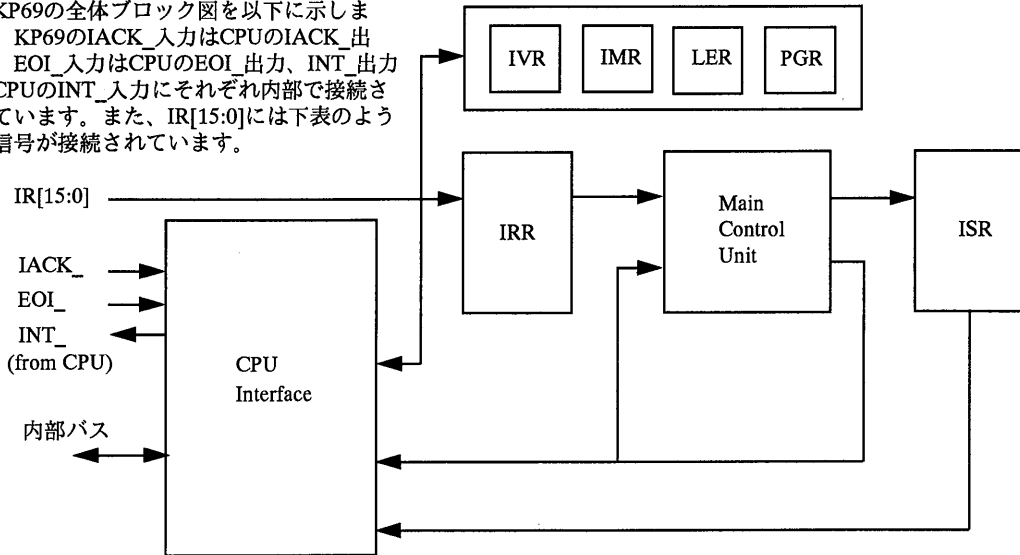


図 6-1 KP69 ブロック図

表 6-1 KL5C8012 の割り込み

	割り込み要求元
IR[15]	タイマ/カウンタ B チャンネル 2 OUTS 出力
IR[14]	タイマ/カウンタ B チャンネル 1 OUTS 出力
IR[13]	タイマ/カウンタ B チャンネル 0 OUTS 出力
IR[12]	タイマ/カウンタ A チャンネル 1 OUT 出力
IR[11]	タイマ/カウンタ A チャンネル 0 OUT 出力
IR[10]	USART TXEMPTY 出力
IR[9]	USART RXRDY 出力
IR[8]	USART TXRDYPIN 出力
IR[7]	外部入力 P07/IR7
IR[6]	外部入力 P06/IR6
IR[5]	外部入力 P05/IR5
IR[4]	外部入力 P04/IR4
IR[3]	外部入力 P03/IR3
IR[2]	外部入力 P02/IR2
IR[1]	外部入力 P01/IR1
IR[0]	外部入力 P00/IR0

### 6.3 レジスタの構成とI/Oレジスタマッピング

KP69 は以下のレジスタを備えています。

表 6-1 I/O レジスタマッピング

I/O アドレス	ブロック名	ライト時	リード時
34H	割り込みコントローラ	LERL/PGRL	ISRL
35H		LERH/PGRH	ISRH
36H		IMRL	IMRL
37H		IVR / IMRH	IMRH

#### IRR (Interrupt Request Register)

書き込み不可/読み出し不可

割り込み要求が発生すると該当するビットがセットされます。割り込み要求の受付がレベルモードの場合は要求の消失によって、またエッジモードの場合は割り込みサービス開始によってリセットされます。リセット直後の初期状態では全ての割り込みレベルのエッジ検出回路がリセットされます。

#### ISR (In Service Register)

書き込み不可/読み出し可

IACK<sub>n</sub> 信号が返って来ると、受け付けた割り込み要求の割り込みレベルに該当するビットがセットされます。割り込みサービスが終了するとリセットされます。リセット直後の初期状態では全てのビットがリセットされます。8 ビットごとの読み出しが出来ます。

#### IVR (Interrupt Vector Register)

書き込み可/読み出し不可

KC82 のモード 2 割り込みの割り込みベクタの上位 3 bit をこのレジスタで指定します。このレジスタに書き込みを行なう前と後で、書き込み可能なレジスタが変化します。

	D7	D6	D5	D4	D3	D2	D1	D0
ISRH	ISR[15]	ISR[14]	ISR[13]	ISR[12]	ISR[11]	ISR[10]	ISR[9]	ISR[8]

	D7	D6	D5	D4	D3	D2	D1	D0
ISRL	ISR[7]	ISR[6]	ISR[5]	ISR[4]	ISR[3]	ISR[2]	ISR[1]	ISR[0]

ISR[n]	意味
0	非サービス中
1	サービス中

	D7	D6	D5	D4	D3	D2	D1	D0
IVR	IVR[7]	IVR[6]	IVR[5]	X	X	X	X	X

### LER (Level / Edge Register)

書き込み可／読み出し不可

割り込み要求入力（以下IR入力とします。）のレベル／エッジのモード切替を制御します。各IR入力ごとに設定可能です。リセット直後の初期状態では全てのビットがレベルモードになります。なお、このレジスタへの書き込みはIVR設定“前”に行なって下さい。このレジスタは書き込み専用です。

注意: タイマ/カウンタの割り込みを使用する場合は対応するIR入力をエッジモードに設定する必要があります。

### PGR (Priority Group Register)

書き込み可／読み出し不可

IR入力の優先順位グループを設定します。各IR入力ごとに設定できます。優先順位グループには“HIGH”と“LOW”の2種類があります。リセット直後の初期状態では全てのビットが“LOW”に設定されます。なお、このレジスタへの書き込みはIVR設定“後”に行なって下さい。このレジスタは書き込み専用です。

	D7	D6	D5	D4	D3	D2	D1	D0
LERH	LER[15]	LER[14]	LER[13]	LER[12]	LER[11]	LER[10]	LER[9]	LER[8]

	DI7	DI6	DI5	DI4	DI3	DI2	DI1	DI0
LERL	LER[7]	LER[6]	LER[5]	LER[4]	LER[3]	LER[2]	LER[1]	LER[0]

LER[n]	意味
0	LEVEL mode
1	EDGE mode

	D7	D6	D5	D4	D3	D2	D1	D0
PGRH	PGR[15]	PGR[14]	PGR[13]	PGR[12]	PGR[11]	PGR[10]	PGR[9]	PGR[8]

	D7	D6	D5	D4	D3	D2	D1	D0
PGRH	PGR[7]	PGR[6]	PGR[5]	PGR[4]	PGR[3]	PGR[2]	PGR[1]	PGR[0]

PGR[n]	意味
0	“LOW” group
1	“HIGH” group

### IMR (Interrupt Mask Register)

書き込み可／読み出し可

IR入力のマスク状態を設定します。各IR入力ごとに設定できます。リセット直後の初期状態では全てのビットがマスクセット状態になっています。なお、このレジスタへの書き込みはIVR設定“後”に行なって下さい。このレジスタは書き込み／読み出し可能です。

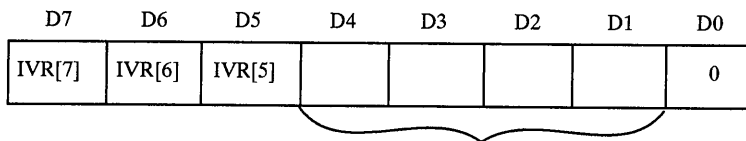
	D7	D6	D5	D4	D3	D2	D1	D0
IMRH	IMR[15]	IMR[14]	IMR[13]	IMR[12]	IMR[11]	IMR[10]	IMR[9]	IMR[8]

	D7	D6	D5	D4	D3	D2	D1	D0
IMRL	IMR[7]	IMR[6]	IMR[5]	IMR[4]	IMR[3]	IMR[2]	IMR[1]	IMR[0]

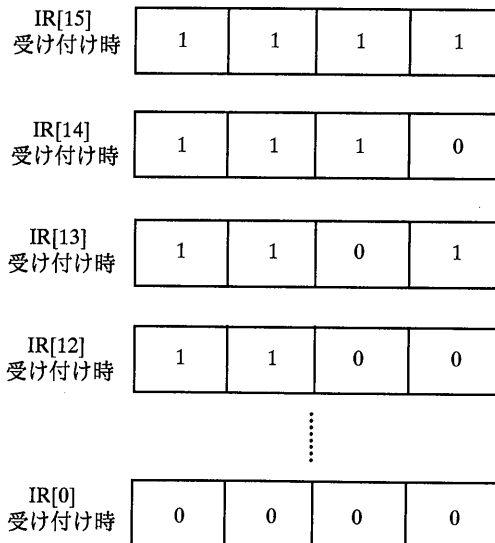
IMR[n]	意味
0	非マスク状態
1	マスク状態

### 6. 4 割り込みベクタ出力

割り込みベクタの出力フォーマットを示します。  
IACK\_ に応答して出力される割り込みベクタは図 6-2 のようになります。



割り込み要求入力 (IR[n]) のビット番号に応じて出力



ビット 7 ~ 5 : IVR に設定したデータ

ビット 4 ~ 1 : 受け付けた割り込みレベルを 2 進数にしたコード

ビット 0 : 0 に固定

図 6-2 割り込みベクタ

このように割り込みベクタは、割り込み要求入力のビット番号が大きくなるにしたがって大きくなる連続した偶数値を出力します。

### 6. 5 割り込み要求の優先順位

KP69 では各割り込み要求入力の優先順位は、リセット直後の初期状態では番号が大きい順となっています。また、優先順位グループという考え方をを用いて、優先順位の変更を可能としています。優先順位グループには“LOW”と“HIGH”の2つがあり、各IR

ごとに設定可能です。“HIGH”グループのIRの方が“LOW”グループのIRよりも優先順位が高く、また各グループ内ではIRのビット番号が大きいほど優先順位が高くなっています。以下に例を挙げておきます。

割り込み要求入力	グループ
IR[15]	LOW
IR[14]	LOW
IR[13]	LOW
IR[12]	HIGH
IR[11]	LOW
IR[10]	HIGH
IR[9]	LOW
IR[8]	HIGH
IR[7]	HIGH
IR[6]	LOW
IR[5]	LOW
IR[4]	HIGH
IR[3]	LOW
IR[2]	HIGH
IR[1]	HIGH
IR[0]	HIGH

⇒

割り込み要求入力	グループ	優先順位
IR[12]	HIGH	最高
IR[10]	HIGH	↓
IR[8]	HIGH	↓
IR[7]	HIGH	↓
IR[4]	HIGH	↓
IR[2]	HIGH	↓
IR[1]	HIGH	↓
IR[0]	HIGH	↓
IR[15]	LOW	↓
IR[14]	LOW	↓
IR[13]	LOW	↓
IR[11]	LOW	↓
IR[9]	LOW	↓
IR[6]	LOW	↓
IR[5]	LOW	↓
IR[3]	LOW	最低

図 6-3 割り込み優先順位

### 6. 6 レジスタ設定シーケンス

KP69 の内部レジスタは、表 6-2 のように I/O アドレスが割り付けられています。したがって、リセット解除後の各レジスタの設定は、LER → IVR → IMR(または PGR) という順に行ないます。IVR を設定した後は IMR および PGR の設定のみ可能となりますのでご注意ください。

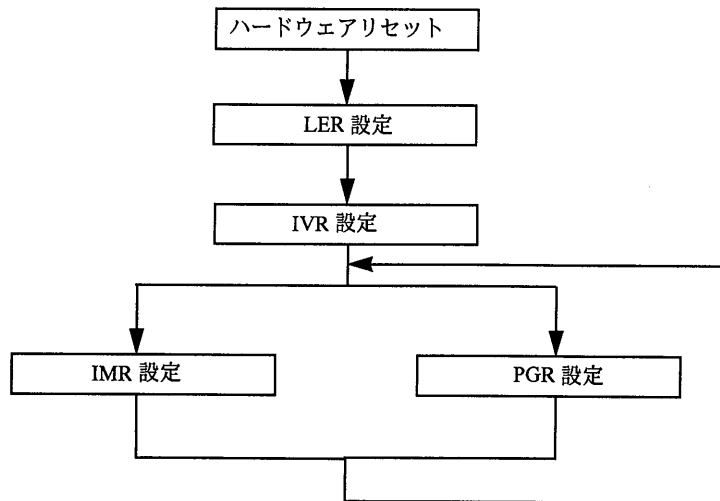


図 6-4 レジスタの設定シーケンス

### 6. 7 レジスタの読み出し

KP69 のレジスタのうち、ISR と IMR は常時読みだし可能です。不正割り込み判定のためには ISR の読み出しを行なう必要があります。

### 6. 8 割り込み要求の受け付け

割り込み要求の受け付けにはレベル/エッジの2種類のモードがあり、LER で設定します。

#### レベルモード

レベルモードでは IR 入力の “H” レベルを割り込み要求として認識します。リセット後 LER 未設定の状態ではこのモードです。

#### エッジモード

エッジモードでは IR 入力の立ち上がりを割り込み要求として認識します。この場合その割り込み要求は受け付けられるまで保持されます。

#### 動作シーケンス

IR[n] 端子への割り込み要求が発生した時の動作説明をします。

IR 入力が “H” になり割り込み要求が発生すると、IRR の該当ビットがセットされます。この割り込み要求は IMR によるマスクや ISR、PGR による優先順位の判断を経て INT\_ 信号になります。KC82 はそれを受けて IACK\_ 信号を “L” にします。IACK\_ を受けると KP69 は割り込み要求の発生した IR[n] に対応する割り込みベクタを出力し、対応する ISR のビットのセットと IRR のビットのリセット（エッジモードの場合）を行ないます。これで割り込みサービス状態に入ります。ISR のセットが行なわれた時点で次のエッジの割り込み要求の受付が可能になります。

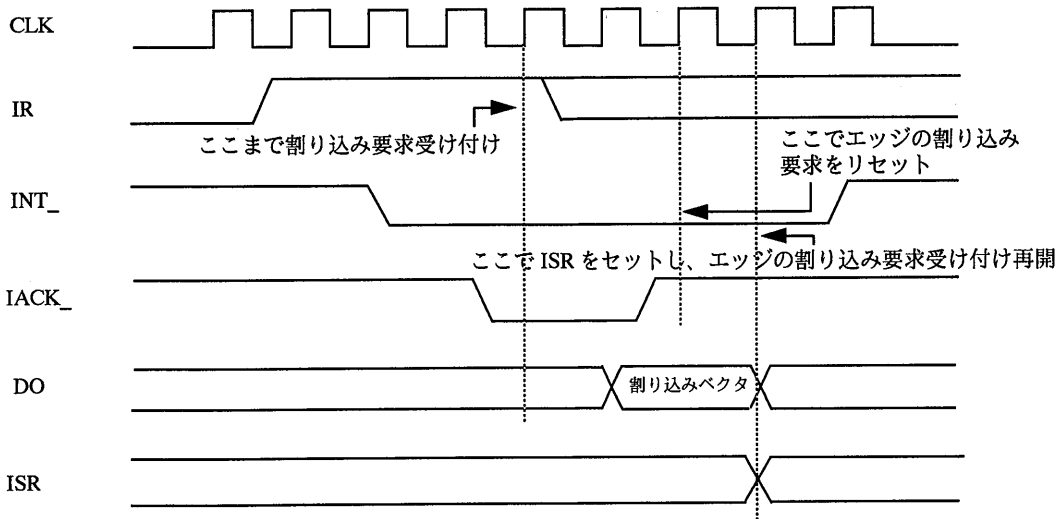


図 6-5 割り込み要求の受け付けタイミング

### 6. 9 割り込みの終了

KP69 は KC82 からの EOI\_ 信号が "L" になることにより、現在サービス中の割り込みの中で最優先レベルに相当する ISR ビットをリセットします。これによりそのレベルの割り込みサービスが終了したことになります。

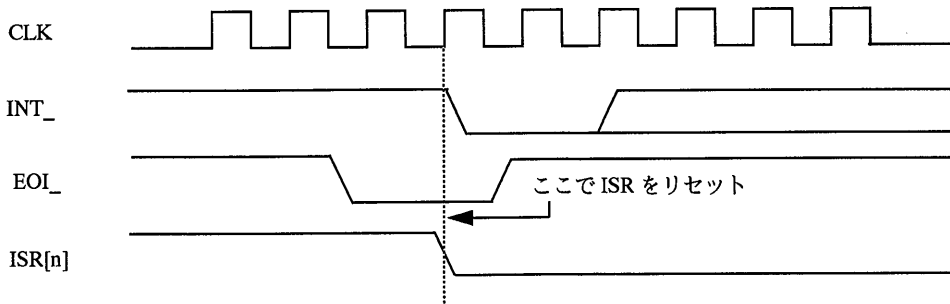


図 6-6 割り込み終了タイミング

### 6. 10 多重割り込み処理

KP69 は割り込み要求のあった割り込みレベルと現在サービス中の割り込みレベルをあらかじめ設定された優先順位で比較します。そのとき新たに要求のあった割り込みレベルが現在サービス中の割り込みレベルより優先順位が高ければ新しい割り込み要求を受け付け、優先順位が同じかあるいは低ければ割り込み要求を拒否あるいは待たせます。これによって多重割り込み処理を実現しています。多重割り込みの状況は ISR を読み出して知ることが出来ます。

### 6. 11 不正割り込み処理

不正割り込みとは、レベルモードに設定されている割り込み要求入力において、割り込み要求の発生によって INT<sub>0</sub> が “L” になり、それを受けて KC82 が IACK<sub>0</sub> を返した際に、すでにその割り込み要求がなくなってしまう、しかも「現在サービス中の割り込みのうち優先順位が最高のも」より上位の割り込み要求入力がない場合をいいます。このとき、KP69 は ISR のセットは行わず、IR[0] に相当する割り込みベクタを出力し不正割り込み処理状態になります。この状態は KC82 からの EOI<sub>0</sub> によって不正割り込み処理ルーチンが終了するまで続きます。このときの EOI<sub>0</sub> においては、ISR はリセットされません。また不正割り込み処理状態においては他のいかなる割り込み要求（正常、不正）も受け付けません。このため、IR[0] の割り込み処理ルーチン内では、ISR[0] がセットされているかどうかによって、正常な割り込み要求だったのか不正な割り込み要求だったのかを判断し処理を分ける必要があります。なお、IR[0] の割り込みサービス中に不正割り込み処理状態になった場合、ISR を読み出すと ISR[0]='0' となります。したがって、ISR[0]='1' ならば、正常な割り込み要求、ISR[0]='0' ならば不正割り込み要求、と区別することが可能となっています。

### 6. 12 リセット

RESET 端子を “L” レベルにすると、次のように動作します。

- (1) IMR を 'FFFFH' にセットします。（全レベルマスク状態）
- (2) IRR、ISR、LER、PGR を '0000H' にリセットします。
- (3) 不正割り込み処理状態を無効にします。
- (4) IVR を未設定状態にします。（再設定が必要です。）

### 6. 13 注意事項

- (1) CPU の割り込みモードはモード 2 を使って下さい。
- (2) 割り込み処理ルーチンの最後には必ず RETI 命令（コード ED 4D）を置いて下さい。
- (3) タイマ/カウンタの割り込みを使用する場合は対応する IR 入力をエッジモードに設定する必要があります。

### 6. 14 応用 (ピン 80 ~ 83, 85 ~ 88 の使い方)

外部入力 IR7 ~ IR0 はピン 80 ~ 83, 85 ~ 88 の入力バッファに常に接続されています。したがってポートが入力方向のとき、IR7 ~ IR0 に入力されている割り込み要求の有無をポート読み込みにより確かめることができます。これは多重割り込みの割り込み要求有無チェックに有効です。またポートを出力方向にセットすることにより外部割り込みをポート出力からソフトウェアにより発生させることも可能です。以下に使用上の注意を説明します。

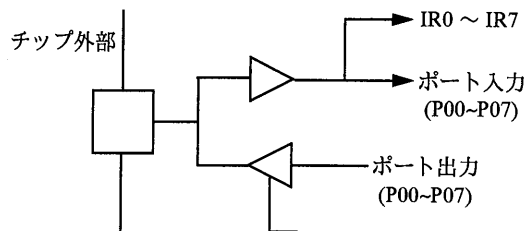


図 6-7 ピン 80 ~ 83, 85 ~ 88 の構造

#### 使用上の注意

- (1) 外部入力を IR7 ~ IR0 に入れる場合  
パラレルポートは入力方向に設定して下さい。
- (2) パラレルポートを出力方向で使う場合  
IR7 ~ IR0 へはパラレルポートが出力している値が入力されます。IR7 ~ IR0 を使用しないときは対応する外部割り込み要求レベルをマスクして下さい。

## 7. USART

### 7.1 概要

本チップにはシリアルポートとしてUSART(KP51マクロセル)が使用されています。KP51は、マイクロコンピュータ用に設計されたプログラマブルな通信用マクロです。一度初期設定されれば、設定されたモードに従って、データの送信・受信を行ないます。本マクロは、モデムや端末とシステムバス間に位置し、データ通信に必要な以下の機能を備えています。

#### 同期モード・非同期モード

以下の様な設定が可能です。

##### (1)同期(sync)モード

- ・キャラクタ長 5~8ビット
- ・同期キャラクタ数 1~2キャラクタ
- ・パリティビット付加(偶数/奇数)/無付加・内部同期検出/外部同期検出

##### (2)非同期(async)モード

- ・キャラクタ長 5~8ビット
- ・ストップビット長 1,1.5,2ビット
- ・パリティビット付加(偶数/奇数)/無付加
- ・ボーレート x1, x16, x64

#### 受信時のエラー検出

受信時、ステータス情報として以下のエラーを検出します。

- ・パリティエラー(パリティイネーブル時)
- ・オーバーランエラー
- ・フレーミングエラー(asyncモードのみ)

#### ブレイク信号の送出・検出

##### 送信イネーブル・受信イネーブル

送信イネーブルフラグを '1' にすることにより送信可能となり、 '0' にすることにより現在バッファ内にある送信データを送出後、送信データは '1' (マーキング状態)になります。

受信イネーブルフラグは、 '0' にすると、asyncモードにおいては全ての受信動作(データの受信、フラグの変化)を禁止します。syncモードにおいては、データの受信は禁止せず、RXRDY、オーバーランフラグをリセット状態にします。 '1' で通常動作となります。

##### エンタハント機能

syncモード同期検出の時、エンタハントコマンドの発行によりビット境界ごとの同期検出が可能となります。

### 7.2 KP51内部端子機能説明

端子記号	I/O	機能説明
TXRDYPIN	O	<p>送信レディ信号出力端子</p> <p>送信データの書き込みが可能で、送信可能状態 (CTS = "L" かつ txen= '1')にあるとき "H" を示します。送信データの書き込みを行なうと、"L" を出力し、送信可能状態ならば、データの送信が開始されます。KP51 の内部は二重バッファ構造の為、データの送信を開始すると、この端子は "H" を出力し、次の送信データの書き込みが可能であることを知らせます。この端子は、ステータスレジスタのビット 0(TXRDY) とは以下の点で異なります。</p> <p>ステータスビット0-送信データの書き込みが可能であるとき "H"</p> <p>TXRDYPIN端子 -送信データの書き込みが可能であり、かつ送信可能状態(CTS = "L" かつ txen= '1')にあるとき "H"</p>
TXEMPTY	O	<p>トランスミッタエンpty出力端子</p> <p>送信すべきデータがないとき "H" を出力します。送信データの書き込みを行なうと "L" を出力します。</p> <p>送信ディセーブル状態 (CTS = "H" または txen= '0') のときは、この端子は "H" のままです。また sync モードでは一旦データがライトされ、そのデータ送信後、送信データがない時は sync キャラクタを自動的に送信し続けます。このときも TXEMPTYは "H" を出力します。</p>
RXRDY	O	<p>受信レディ信号出力端子</p> <p>1 キャラクタデータを受信し終えて、データの読み出しが可能になると "H" を出力します。データの読み出しが行なわれると、"L" にリセットされます。受信ディセーブルコマンド (rxen= '0' &lt;コマンドレジスタのビット 2 を '0' にする&gt;) の実行でも "L" にリセットされます。また sync モードでは、エンタハントコマンド (コマンドレジスタのビット 7 を '1' にする)の発行に伴い、"L" にリセットされます。</p>

KP51の外部入出力端子(CTS\_, RXC, DSR\_, RXD, TXD, TXC, DTR\_, RTS\_, SYNDBD, SYDTIN)に関しては3章を参照下さい。

### 7.3 ブロック図

KP51の全体ブロック図を以下に示します。KP51の内部端子(TXRDYPIN, TXEMPTY, RXRDY)は内蔵の割り込みコントローラに接続されています。

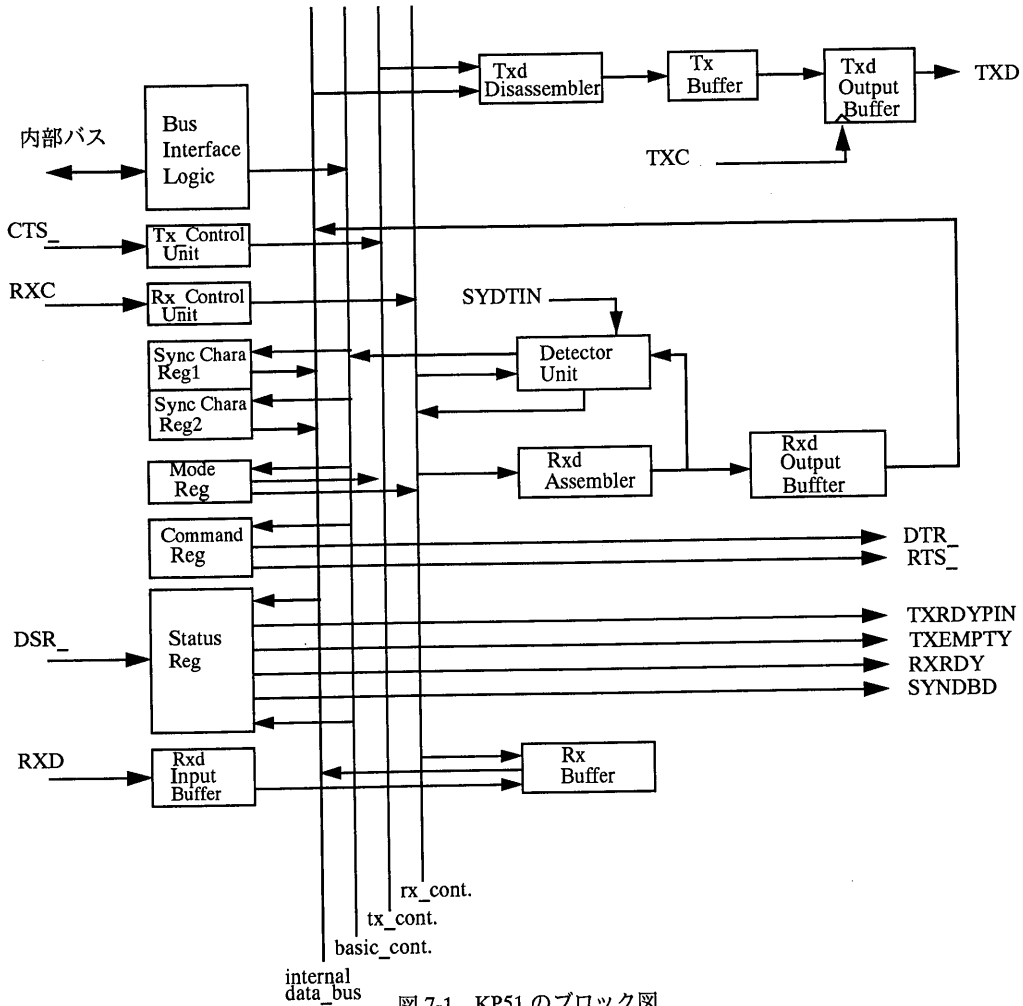


図 7-1 KP51 のブロック図

#### I/O レジスタマッピング

I/Oアドレス	ブロック名	ライト時	リード時
38H	USART	送信データ	受信データ
39H		モード/コマンド/同期キャラクタ設定	ステータス

### 7.4 初期設定

データの送受信を始める前に、リセット後、図 7.2 に示す様にモード設定とコマンドの書き込みを行なう必要があります。

#### モード設定

リセット後、モード/コマンド書き込みを行なうと、モードレジスタへのデータ書き込みとなります。リセット (外部 RESET 入力、またはソフトウェアリセットコマンド書き込み) 後は必ずこのモード設定待ちの状態になっています。

ここでは、sync/async モードの選択、ボーレートの設定、キャラクタ長の設定などを行ないます。次ページのフォーマットに従って、モード設定を行なって下さい。

#### 同期キャラクタの設定

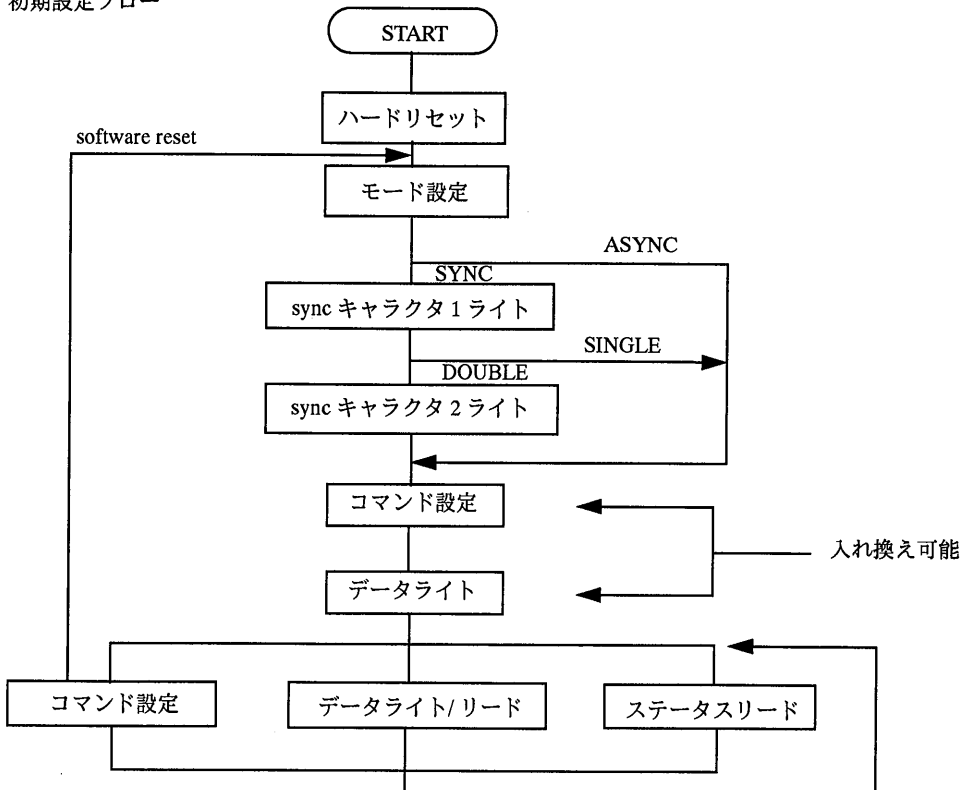
モード設定によって sync モードにプログラムされた場合、次に同期キャラクタの書き込み待ちになります。ここで、シングルモードなら 1 つ、ダブル (bi-sync) モードなら 2 つの同期キャラクタを、書き込んで下さい。なおモード設定によって async モードにプログラムされた場合は、次のコマンド設定に自動的にスキップします。

#### コマンド設定

モード設定、同期キャラクタの設定が終わると、I/O アドレス = 39H でのデータ書き込みは、全てコマンドレジスタへの書き込みとみなされます。

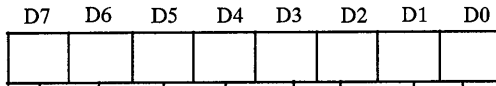
このコマンド設定により、データの転送が可能となります。ページ 7-6 に従って、コマンド設定を行なって下さい。

図 7-2 初期設定フロー



### モードレジスタ

#### (1) sync モード設定の場合



どちらも '0' をセット

キャラクタ長	5	6	7	8
D2	0	1	0	1
D3	0	0	1	1

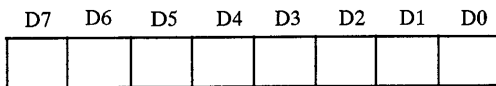
パリティイネーブル  
1:イネーブル  
0:ディセーブル

パリティチェック  
1:偶数  
0:奇数

同期検出  
1:外部同期検出  
0:内部同期検出

同期キャラクタ  
1:シングル  
0:ダブル

#### (2) async モード設定の場合



ボーレート	x1	x16	x64
D0	1	0	1
D1	0	1	1

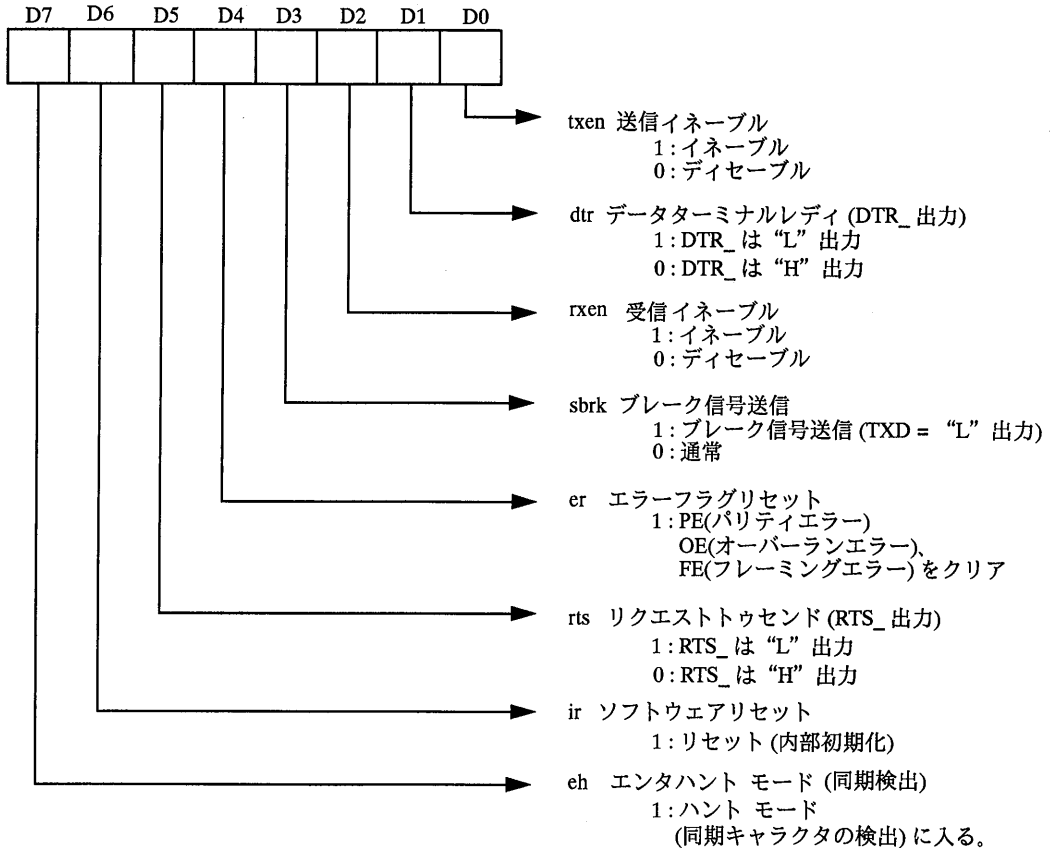
キャラクタ長	5	6	7	8
D2	0	1	0	1
D3	0	0	1	1

パリティイネーブル  
1:イネーブル  
0:ディセーブル

パリティチェック  
1:偶数  
0:奇数

ストップビット	無効	1	1.5	2
D6	0	1	0	1
D7	0	0	1	1

### コマンドレジスタ



\* コマンドレジスタの D7,D6,D4 は、単発動作のため、その動作を解除するために'0'を書き込む必要はありません。

## 7.5 動作モード

以下にモードごとの動作を示します。

### async モード送信

CPU からデータがライトされると、モード設定によって与えられた条件に基づいて、スタートビット、パリティビット、ストップビットをキャラクタデータと共に送信フォーマットにデイスアSEMBルします。

コマンドにより送信ブレイクコマンド (sbrk= '1') が設定されると、TXD 端子は "L" を出力します。

送信ブレイクコマンド (sbrk= '1') が設定されていなければ、送信イネーブル状態 (CTS\_ 端子= "L" かつ txen= '1') ならデータの送信が行なわれます。送信開始前は送信データをライトするまで TXD 端子は "H" (マーキング状態) になります。またデータ 1 キャラクタ送信終了後次に送信すべきデータがなくなると、同じくマーキング状態になります。

送信データは TXC の立ち下がりに同期して、設定されたボーレートに応じて、TXC のレートの 1 倍、1/16 倍、1/64 倍で送信されます。

### async モード受信

受信イネーブル (rxen= '1') 時、RXD 端子の入力の "H" を検出すると RXD 端子の入力が "L" になるのを待ちます。RXD 入力が "L" になるとスタートビットが有効となり、さらにそのビットの中心で "L" をサンプルすると、スタートビットであると認識し、データの受信を開始します。受信動作中はパリティエラー、フレーミングエラー、オーバーランエラー、ブレイク信号検出を行ないます。各種エラーは、後述のステータスレジスタの読み出しによって確認できます。これらのエラーは、受信動作そのものには影響しません。

またフレーミングエラーは、設定されたストップビット数によらず、1 個目のストップビットが "L" であれば、フレーミングエラーと判断します。

### sync モード送信

TXD 端子は、送信イネーブル状態 (CTS\_ 端子 = "L" かつ txen= '1') でも、最初のデータを書き込まない限り "H" (マーキング状態) のままです。送信すべきデータがあれば、TXC の立ち下がりに同期してデータの送信が行なわれます。そして、送信すべきデータがなくなる (TXEMPTY="H" になる) 前に、次の送信データを供給してやらなくてはなりません。データの供給が遅れると、次のデータが供給されるまで、sync キャラクタの送出を自動的に繰り返します。

sync キャラクタ送出時には、次のデータが供給されるまで TXEMPTY 端子は "H" を示し、データが供給された時点で、"L" になります。

なお sync キャラクタの送出は、最初の送信データの書き込みが行なわれない限り実行されません。

### sync モード受信

同期検出には、内部同期検出と外部同期検出の二通りあります。sync モードの場合、受信開始前にエンタハントコマンド (コマンドレジスタのビット 7 を '1' にセットする) を発行しておく必要があります。エンタハントコマンドの発行によって同期キャラクタの検出が行なわれます。

内部同期検出では、RXC の立ち上がりごとに sync キャラクタとの一致検出が行なわれます。bi-sync モードの場合は、2 番目の sync キャラクタとの比較も行なわれます (連続的に 2 個の同期キャラクタを受信しなければ、無効です)。

同期キャラクタの検出が終了すると、ここで SYNDBD 端子が "H" にセットされてハントモードから抜けます。SYNDBD 端子はデータの最終ビットの中央でセットされますが、パリティを含んでいる場合は、パリティビットの中央でセットされます。

また同期キャラクタの検出は、ハント終了後も、規定のキャラクタ境界ごとに行なわれます。

外部同期検出では、SYDTIN 端子に RXC の 1 サイクル以上 "H" を入力することで、同期検出完了と判断され、ハントモードから抜けます。このとき内部同期検出同様に、SYNDBD 端子が "H" にセットされます。

この外部同期検出の時は、内部同期検出は行なわれません。

また sync モードにおいては、送信イネーブル、ディセーブル状態にかかわらずデータの受信は行なわれます。パリティエラーの検出は async モードと同じで、ハントモード中以外の時行なわれます。

### 7.6 ステータスレジスタ

送信受信の際のステータス情報が格納される8ビットのレジスタです。このレジスタの内容を読み出すことにより、ポーリング動作が可能です。

以下にステータスレジスタの割付を示しますが、ステータスによってはセットされる条件が外部端子と異なるものがあるので、注意して下さい。

D7	D6	D5	D4	D3	D2	D1	D0
DSR	SD/ BD	FE	OE	PE	TxE	RxR	TxR

#### D0 TXRDY

TXRDYPIN 端子とはセットされる条件が異なります。

ステータス = 内部 Data Buffer が空の時 '1'

TXRDYPIN 端子 = 内部 Data Buffer が空で、かつ CTS\_ 端子 = "L" で、  
かつコマンドレジスタビット0 (txen) = '1' の時 "H"

#### D1 RXRDY

RXRDY 端子と同じ機能。

#### D2 TXEMPTY

TXEMPTY 端子と同じ機能。

#### D3 PE (パリティエラー)

受信データにパリティエラーを検出すると '1' になります。

受信 イネーブル (コマンドレジスタビット2 (rxen = '1') 受信ディセーブル (rxen = '0') には関係なく変化します。

sync モード時、ハントモード中は動作しません。

エラークリアコマンド (コマンドレジスタビット4 (er) = '1') で、'0' にクリアされます。

#### D4 OE (オーバーランエラー)

データ受信後、データの読み出しが遅れて、その次のデータを受信完了してしまうと、受信イネーブル (rxen = '1') 状態なら '1' にセットされます。

エラークリアコマンドまたは受信ディセーブルコマンドで '0' にクリアされます。

#### D5 FE (フレーミングエラー)

async モード時、ストップビットに '0' を検出すると '1' になります。

ストップビット長によらず、1 個目のストップビット中央でのみサンプルされます。

エラークリアコマンドで '0' にクリアされます。

#### D6 SYBD (同期検出・ブ레이크信号検出)

ステータス読み出し中は変化しない点を除いて、SYNDBD 端子と同じ値を出力します。

sync モード時、外部同期・内部同期に関わらず、同期検出されると '1' になり、ステータス読み出しによって '0' にクリアされます。

async モード時、ステータスビット検出後、ブ레이크信号 (RXD 入力 = "L") として2キャラクタ以上 '0' を受信すると '1' になり、RXD = "H" 検出またはリセットにより '0' にクリアされます。

#### D7 DSR (データセットレディ)

入力端子 DSR\_ = "H" の時 '0'、"L" の時 '1' になります。

CLK に同期して取り込みます。

\* ステータスレジスタの内容は、読み出し中は更新されません。

## 7.7 注意事項

### イニシャライズについて

ソフトウェアリセットコマンド (コマンドレジスタビット6をセット)によるイニシャライズを行なう場合は、コマンドレジスタ選択の状態、'00H'を3回書き込んでから、ソフトウェアリセットコマンドを実行して下さい。

### エンタハントの実行について

受信イネーブルフラグのセット (コマンドレジスタビット2をセット)と、エンタハントコマンド(コマンドレジスタビット7をセット)する時は、同時にエラークリアコマンド (コマンドレジスタビット4をセット)を実行して下さい。

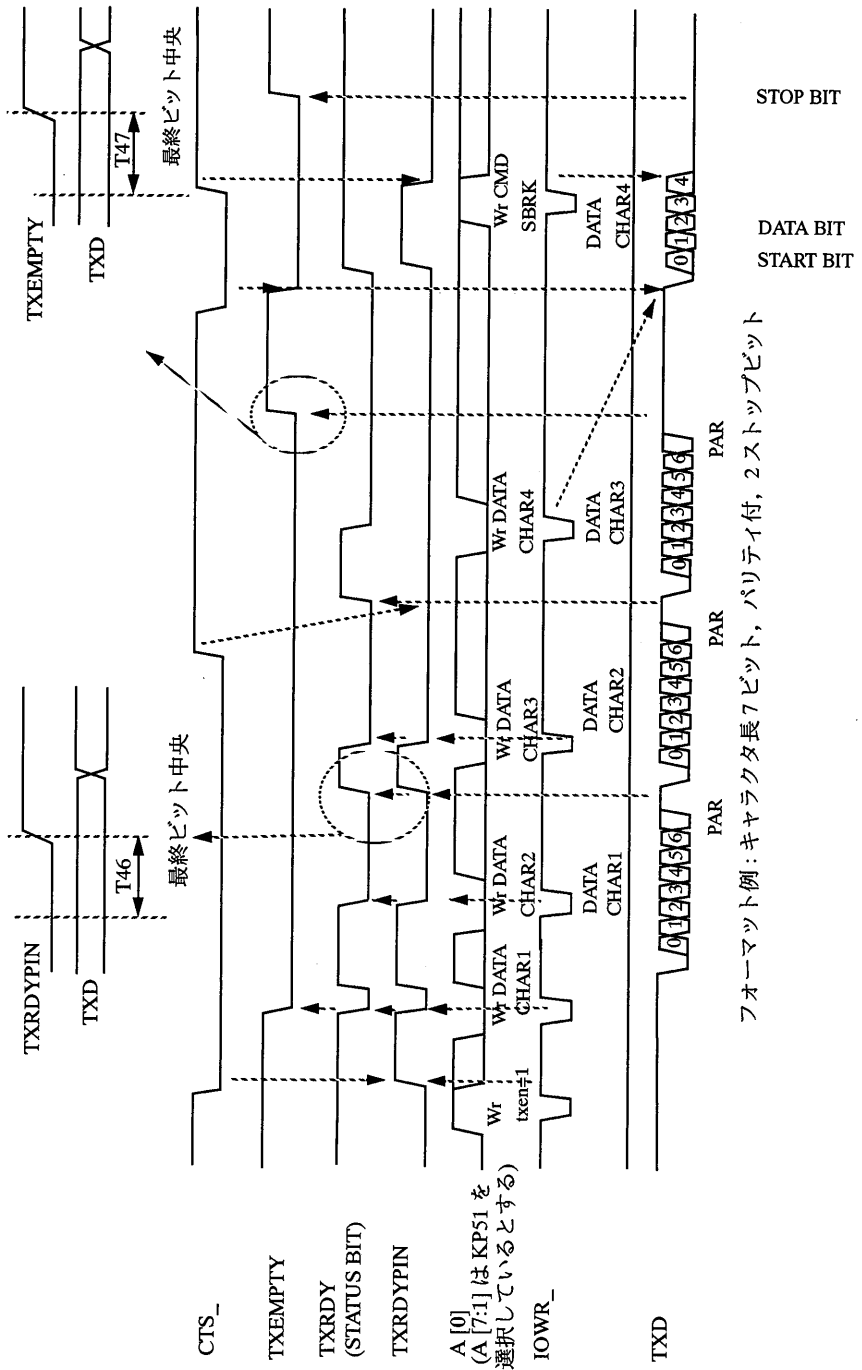
### フレーミングエラー後の受信データについて

フレーミングエラー後の受信データは、保証されません。

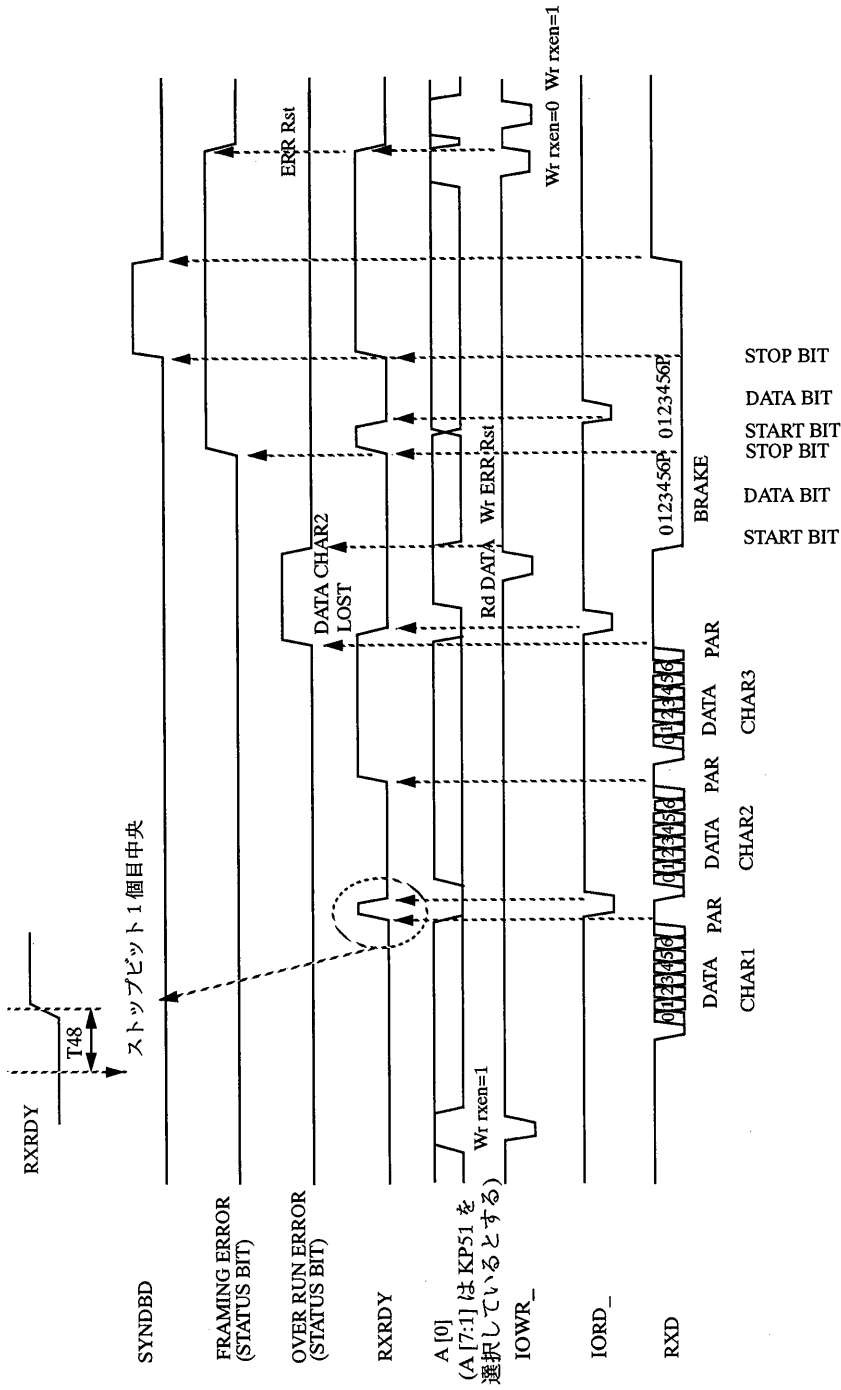
### 外部同期検出について

外部同期検出において、SYDTINの入力は、RXC="H"の時常に認識されます。したがって、ステータスリードによって SYNDBD 端子および同期検出ステータス(ステータスレジスタビット6)はクリアされても、RXC="H"の時に、SYDTIN入力が"H"なら再度 SYNDBD 端子及び同期検出ステータスはセットされます。

### トランスミッタ・コントロールとフラグ変化のタイミング (非同期モード送信)

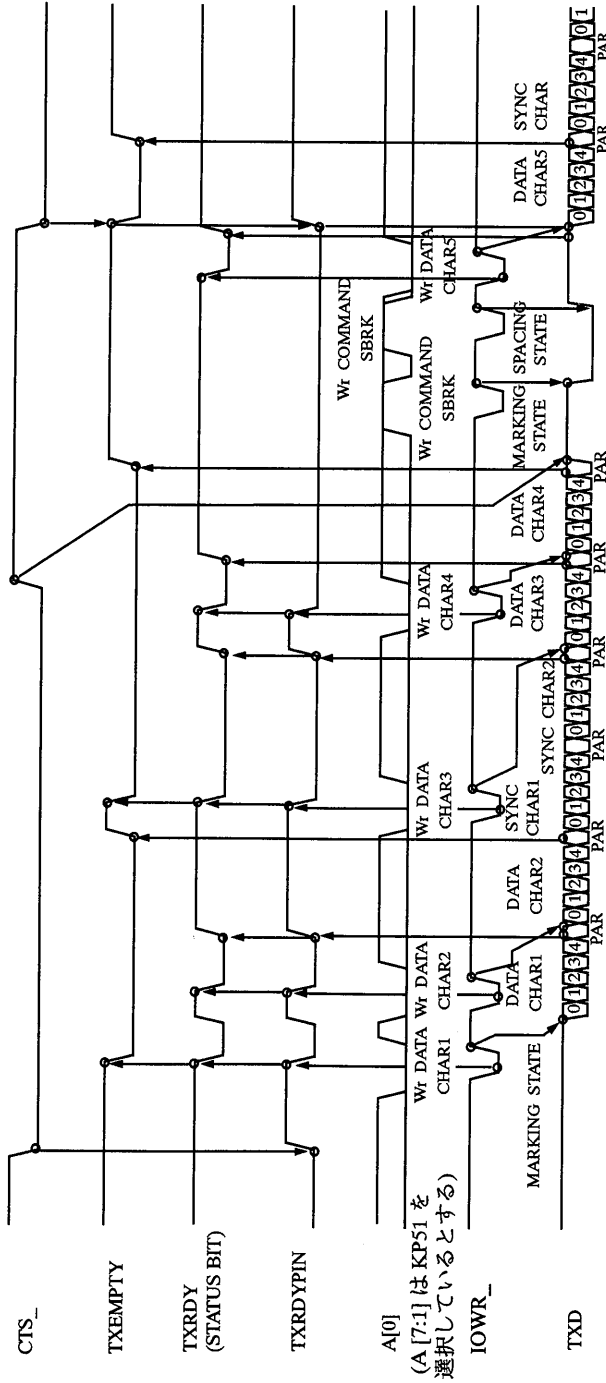


### レーザーコントロールとフラグ変化の タイミング (非同期モード受信)



フォーマット例: キャラクタ長7ビット, パリティ付, 2ストップビット

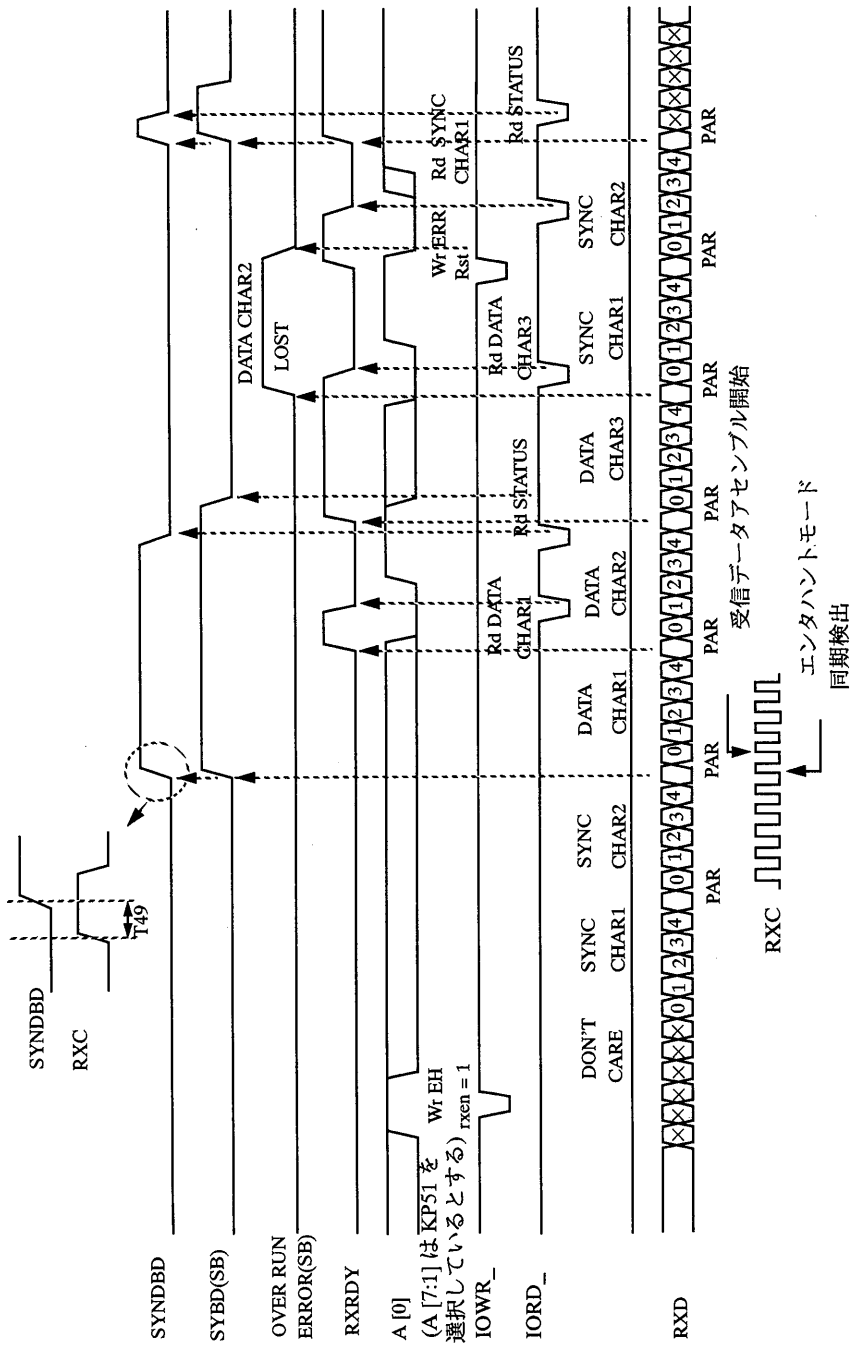
トランスミッタ・コントロールとフラグ変化のタイミング(同期モード送信)



(A[7:1]はKP51を  
選択しているとする)

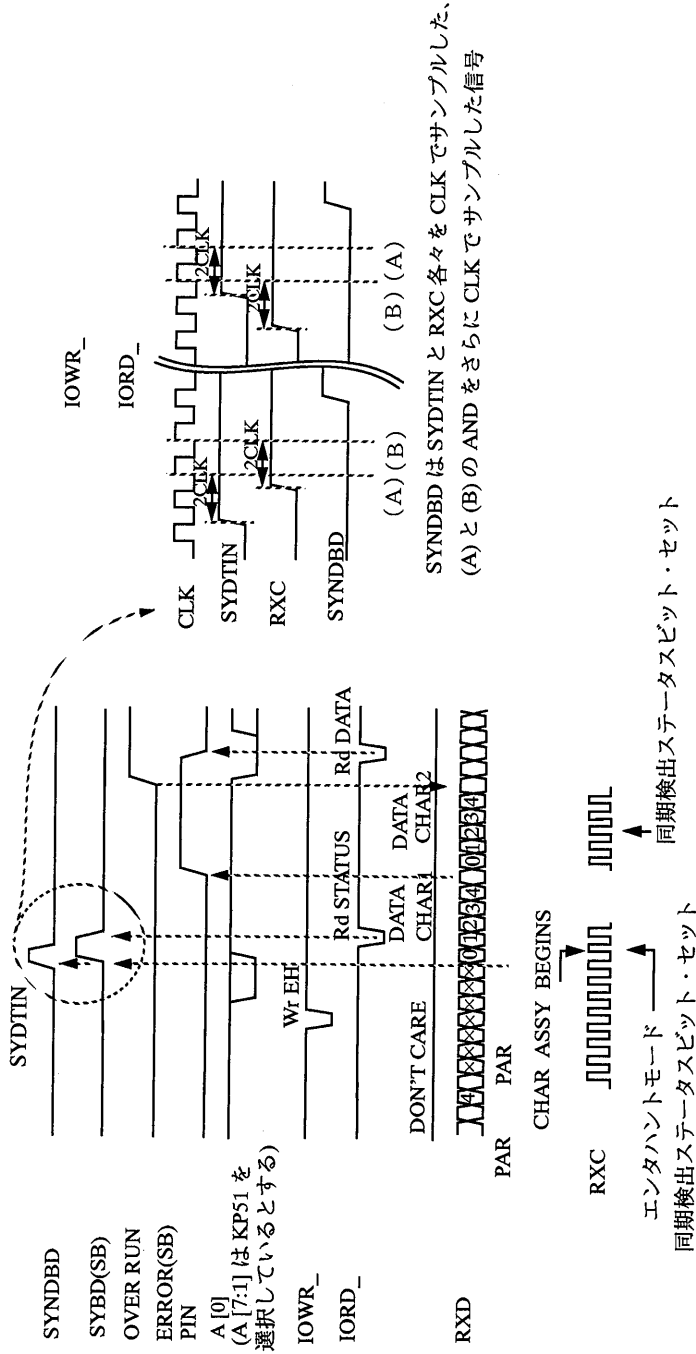
フォーマット例: キャラクタ長5ビット、パリティ付、2シンク キャラクタ

### レシーバコントロールとフラグ変化のタイミング (同期モード受信内部同期検出)



フォーマット例: キャラクタ長5ビット, パリティ付, ダブル同期キャラクタ

### レシーバーコントロールとフラグ変化のタイミング (同期モード受信外部同期検出)



### 8. タイマ/カウンタ A

#### 8.1 概要

タイマ/カウンタ A は 2 個のタイマ/カウンタ・マクロセル KP64 で構成されたタイマ/カウンタ・ブロックです。2 つのチャンネルはまったく同じ構成になっていますので、本章では 1 個分の回路の説明をします。この KP64 はマイクロコンピュータ用に設計されたクロック同期式 16bit プログラマブル・バイナリカウンタ/タイマです。本マクロセルは、4 種類の動作モード（分周モード、パルス幅変調（PWM）モード、パルス発生モード、パルス幅/周期測定モード）をもつダウンカウンタ 1 個で構成され、CPU からのモード制御命令によりモード設定できます。カウント値は、カウンタラッチコマンドで一旦内部レジスタに保持して読み出すことができます。また、ステータスリードコマンドで設定モードや OUT 出力端子の状態を読み出すことができます。

#### 特徴

- ・各チャンネル当り 1 個の 16bit ダウンカウンタ/タイマを内蔵し 4 種類の動作モード設定、動作が可能
- ・カウント動作に影響を与えずに、安定したカウント値の読み出しが可能
- ・カウンタクロックとして、外部クロック (XCLK) の他に、システムクロックの選択も可能
- ・設定モード、OUT 出力の状態等のステータスの読み出しが可能
- ・2 本のチャンネルをカスケード接続して 32bit カウンタとして使えます (12 章参照)

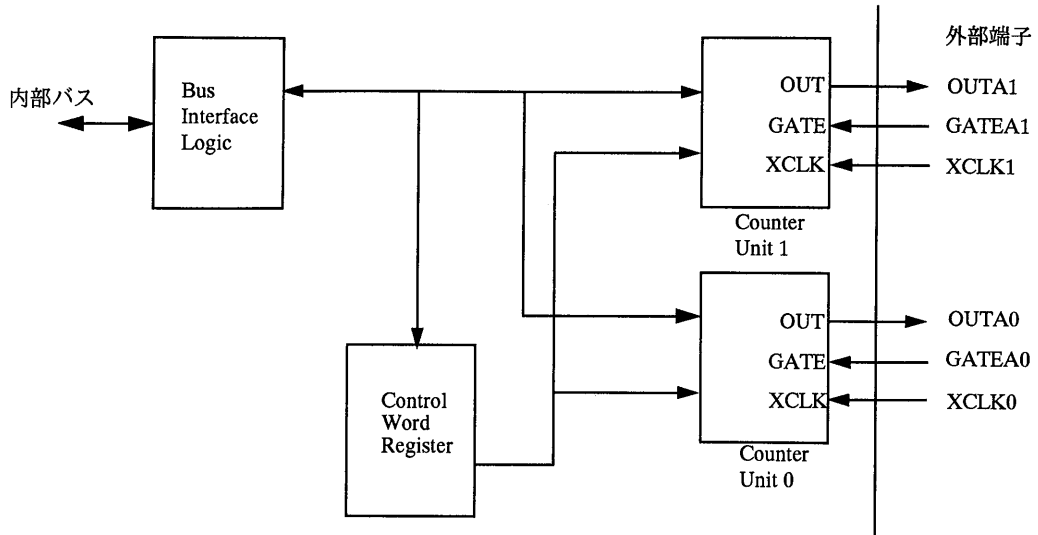
#### 8.2 端子機能詳細説明

端子機能	I/O	機能説明
XCLK	I	外部カウンタクロック入力端子 (カウンタのカウント・レートを決定します。モード設定時にシステムクロックをカウンタクロックとして選択している場合を除く)。カウント動作はこの XCLK の立ち下がりで行なわれます。
GATE	I	ゲート入力端子 カウンタに対するカウント動作イネーブル/ディセーブルの指定や、トリガ信号を与えます。パルス幅/周期測定モード選択時には、測定対象の信号を入力します。
OUT	O	アウト出力端子 モードの指定によって、分周出力、ワンショット出力、ストロープ出力、測定完了出力が得られます。リセット信号入力により本出力はカウンタクロックとは非同期に“L”になります。

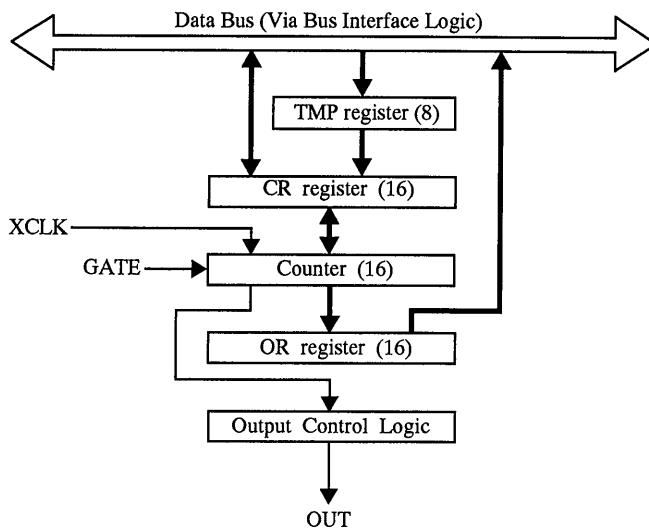
注意) チャンネル1、チャンネル0のXCLK, GATE, OUTの各端子は外部端子ではそれぞれXLCK1, XCLK0, GATEA1, GATEA0, OUTA1, OUTA0という名前になっています。

### 8.3 ブロック図

タイマ/カウンタ A の全体ブロック図と I/O レジスタマッピングを以下に示します。OUTA1, OUTA0 出力は外部端子以外にも、内蔵の割り込みコントローラにも接続されています。



タイマ/カウンタ A のブロック図



各チャネルカウンタ・ユニット内のブロック図

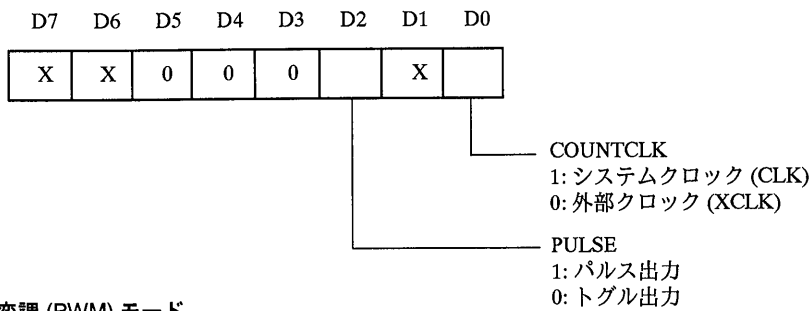
表 8-1 I/O レジスタマッピング

I/O アドレス	ブロック名	ライト時	リード時
28H	タイマ/カウンタA	チャンネル0カウンタ	チャンネル0カウンタ
29H		チャンネル0コントロールワード	チャンネル0ステータス
2AH		チャンネル1カウンタ	チャンネル1カウンタ
2BH		チャンネル1コントロールワード	チャンネル1ステータス

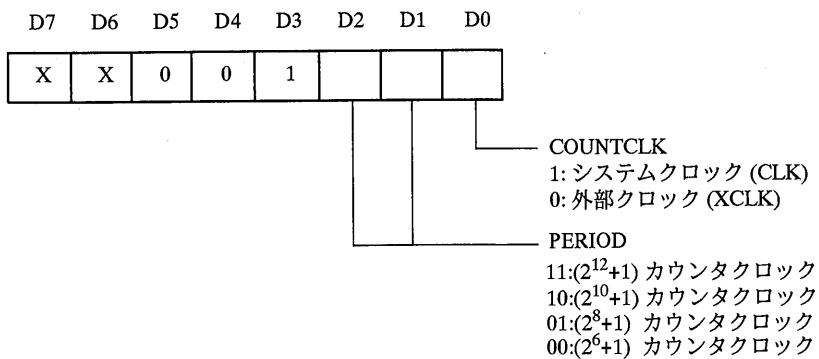
### 8.4 モード設定

モード設定は各チャンネルにコントロールワードを書き込むことで行われます。

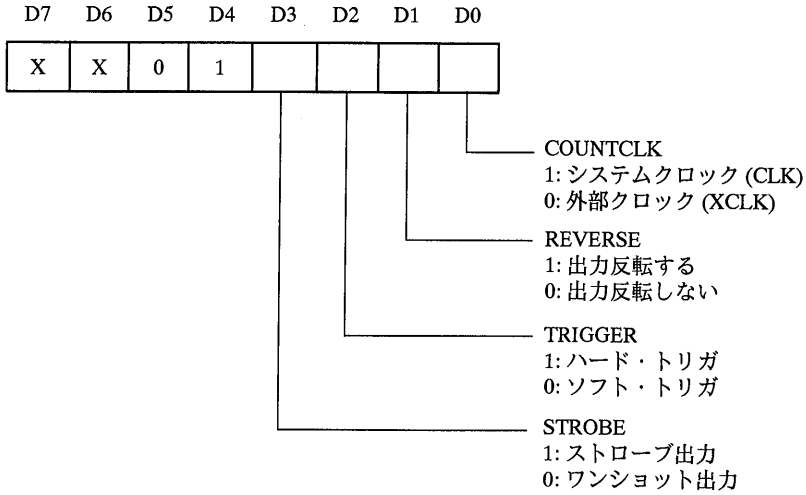
#### 分周モード



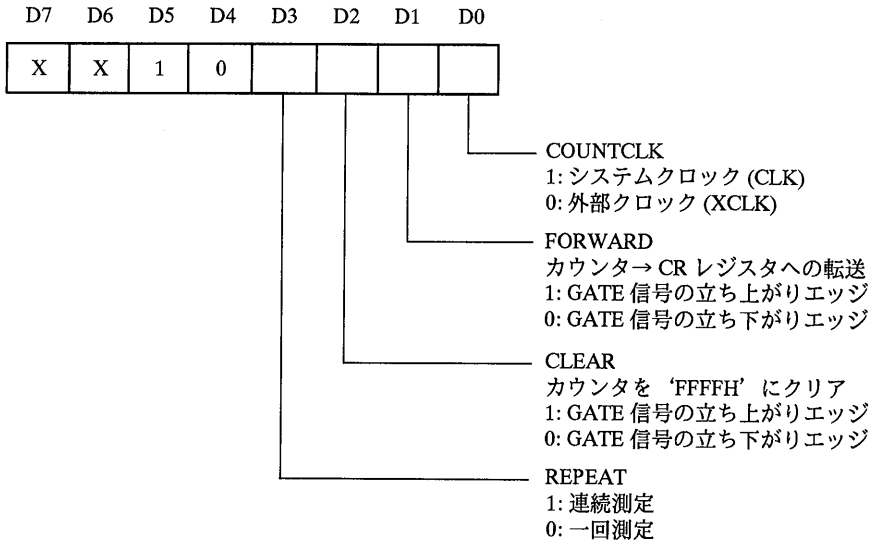
#### パルス幅変調 (PWM) モード



### パルスモード



### パルス幅/周期測定モード



### 8.5 各種コマンド

#### カウンタ・ラッチ・コマンド

カウンタ・ラッチ・コマンドはコントロールワードのD[5:3]= '111B'、D[1:0]= '00B' で実行され、カウンタのカウンタ値が保持されます。一度取り込まれたカウンタ値は次にカウンタ・ラッチ・コマンドが実行されるまで保持されます。データを保持した後にA=28Hあるいは2AHで連続2回のリードを行うことで、カウンタの値を知ることができます。

D7	D6	D5	D4	D3	D2	D1	D0
X	X	1	1	1	X	0	0

#### リードライト・シーケンス・クリア・コマンド

カウンタへのカウンタ初期値の設定やカウンタの読み出しでは、2回のライト動作、リード動作で行いますが、何らかの原因で上位側を読み出しているのか下位側を読み出しているのか判らなくなった場合に、便利なのがリードライト・シーケンス・クリア・コマンドです。リードライト・シーケンス・クリア・コマンドはコントロールワードのD[5:3]= '111B' とし、シーケンス・クリア・ビット(D[1]=1)で実行するとカウンタ内部のレジスタ読み出しの際のリードシーケンス、およびカウンタ値書き込みの際のライトシーケンスをクリアすることができます。

#### フラグ・クリア・コマンド

フラグ・クリア・コマンドはコントロールワードのD[5:3]= '111B' とし、フラグ・クリア・ビット(D[0]=1)にして実行すると、パルスモードでのトリガ受け付けフラグ、およびパルス幅/周期測定モードでの測定完了フラグ(ステータスD7)をクリアすることができます。

D7	D6	D5	D4	D3	D2	D1	D0
X	X	1	1	1	X		

R/W SEQUENCE

- 1: シーケンス・クリアする
- 0: シーケンス・クリアしない

FLAG

- 1: フラグ・クリアする
- 0: フラグ・クリアしない

注意) D[1:0]=00B のときカウンタ・ラッチ・コマンドとして認識されます。

### 8.6 カウンタへのカウンタ初期値の設定

カウンタは 16bit 長のダウンカウンタですが、データを設定するためのバスが 8bit 幅なので、データ書き込みは下位バイト、上位バイトの順に 2 回に分けて行います。書き込みはリセット、モード設定、リードライト・シーケンス・クリア・コマンド直後からの連続 2 回が有効です。

1 回目の書き込み:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(数字はデータビットを表す)

2 回目の書き込み:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(数字はデータビットを表す)

### 8.7 カウンタの読み出し

カウンタの読み出しは、カウンタ・ラッチ・コマンドを実行した直後からの連続2回の読み出し動作によってのみ行なわれます。カウンタの値をカウンタ・ラッチ・コマンドにより、あらかじめタイム/カウンタA内部のレジスタに一旦保持してから、後で読み出す方法をとります。一度取り込まれたカウンタ値は、次にカウンタ・ラッチ・コマンドが実行されるまで、または読み出しが完了するまで保持されます(読み出し例参照)。

1 回目の読み出し:

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(数字はデータビットを表す)

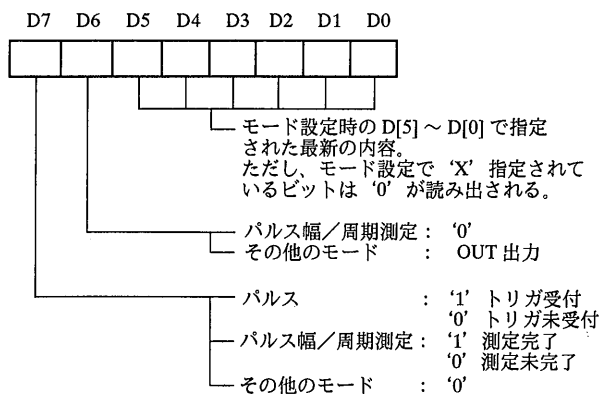
2 回目の読み出し:

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(数字はデータビットを表す)

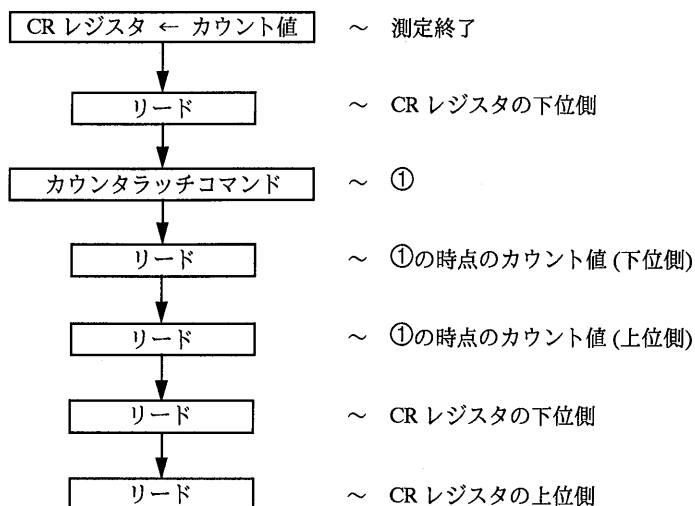
### 8.8 ステータスの読み出し

ステータス情報は、ステータスレジスタの読み出しで読み出すことができます。ステータスのフォーマットは以下の通りです。



#### (例) 読み出し例

パルス幅/周期測定モード



## 8.9 動作モード

### 分周モード

分周モードではカウント初期値をロード→デクリメントし、カウント値がゼロになるとOUT出力が変化するとともに初期値のリロード→デクリメントを繰り返します。モード設定によってCRレジスタ(カウント初期値書き込み用レジスタ)、およびカウンタの値は‘FFFFH’となり、OUT端子は“L”を出力します。出力形態はトグルあるいはパルスのどちらかを選択でき、これを割り込みに利用することができます。

なお、カウント動作はカウント初期値をロードすることで開始し、ゲート機能(GATE端子の入力レベル“H”でカウント・イネーブル、“L”でカウント・ディセーブル)を有します。

OUT出力にはカウント値がゼロになる度に信号が反転するトグル出力とゼロになると1カウンタクロック幅のパルス出力するパルス出力がモード設定で選択できます。リロードはカウント値がゼロになることでのみ発生し、カウント中に初期値の書き込みによるリロードは発生しません。

なお、GATE端子に入力するパルスの幅は“H”、“L”共にシステムクロックの2サイクル分以上にしてください。

### パルス幅変調(PWM)モード

パルス幅変調モードではカウント初期値で設定するパルス幅(“H”の期間)と選択した4種類の周期で構成される繰り返しパルスを発生します。

パルス周期はカウンタクロックの $(2^n+1)$ :  $n=6, 8, 10, 12$ 倍の周期が選択でき、パルス幅はカウント初期値をロード→デクリメントし、カウント値がゼロになるまでの時間、つまり(カウント初期値+1)の幅となります。

モード設定によってCRレジスタは‘FFFFH’となり、カウンタには選択したモードで決定する値 $(2^n-1)$ :  $n=6, 8, 10, 12$ を読み込み、OUT端子は“L”を出力します。

カウント動作はカウント初期値をロードすることで開始し、(カウント初期値のロード→デクリメント→カウント初期値の補数のロード→デクリメント→カウント初期値のロード)を繰り返します。また、分周モードと同様にゲート機能を有します。リロードはカウント値がゼロになることでのみ発生し、カウント中に初期値の書き込みによるリロードは発生しません。

なお、GATE端子に入力するパルスの幅は“H”、“L”共にシステムクロックの2サイクル分以上にしてください。

### パルスモード

パルスモードでは初期設定したカウント値をロード→デクリメントし、カウント値がゼロになるとOUT出力が変化し、再トリガが発生するまでカウント動作を停止します。

カウントトリガはカウント初期値設定をカウント開始のトリガとするソフトトリガおよびGATE端子に入力される信号の立ち上がりをトリガとするハードトリガをモード設定で選択できます。OUT出力には、ロード→カウント動作中“L”で、カウント値=0で“H”となるワンショット出力とカウント値がゼロになると1カウンタクロック幅のパルスを出力するストローブ出力が選択でき、さらにOUT出力の極性が反転するモードも選択できます。

モード設定によってCRレジスタおよびカウンタの値は‘FFFFH’となり、OUT出力は出力反転をディセーブルとした場合は“H”、イネーブルの場合は“L”を出力します。カウント動作はトリガ受け付けにより開始し、トリガ受け付けフラグに‘1’をセットします。カウント値がゼロになるとOUT出力が変化し、カウンタは‘FFFFH’にクリアされた状態で停止します。カウント中に新しいカウント初期値を書き込んだ場合は、すぐにCRレジスタに新しい値がセットされます。

ソフトトリガ選択時には、次のカウンタクロックで新しい値をカウンタに取り込んでカウント動作を継続します(カウント中の再トリガ可能)。このときGATEはカウントイネーブルとして機能します。ハードトリガ選択時には、次の再トリガが行われた後の最初のカウンタクロックで新しい値をカウンタに取り込み、カウント動作を行います(カウント中の再トリガ可能)。

なお、再トリガをかける場合には1つ前のトリガとの間をカウンタクロックの1サイクル分以上にしてください。また、ソフトトリガ選択時にはGATE端子に入力するパルスの幅は“H”、“L”共にシステムクロックの2サイクル分以上にしてください。

### パルス幅/周期測定モード

GATE端子に入力される信号のパルス幅、または周期を測定し、測定が完了するとOUT出力にシステムクロック1サイクル分のパルスを出力するモードです。

測定対象とする信号のパターンは、GATE端子の入力信号の立ち上がりから立ち上がりまたは立ち下がりまで、あるいは立ち下がりから立ち上がりまたは立ち下がりまでの4通りについて設定できます。測定は、モード設定で1回測定と連続測定が選択できます。モード設定によってCRレジスタ、およびカウンタの値は‘FFFFH’となります。以下に、入力信号の立ち上が

りから立ち上がりまで連続測定する場合を例に説明します。

GATE端子の入力信号の立ち上がりを認識すると、カウンタ値の補数をCRレジスタに転送し、カウンタを‘FFFFH’にクリアしてから、次のカウンタクロックでカウントを開始します。ただし、一番最初のCRレジスタへの転送時にはOUT出力にパルスは出力されません。

次の入力信号の立ち上がりを認識すると、再度カウンタ値の補数をCRレジスタに転送し、カウンタを‘FFFFH’にクリアして、OUT出力に“H”パルスをシステムクロック1サイクル分出力し、測定完了フラ

グに“1”をセットします。そして、カウンタ動作を続けます。このように、パルス周期はCRレジスタの値を読み出すことで知ることができます。

つまり、パルス幅/周期測定モードでCRレジスタは、バッファレジスタとして働きます。また、1回測定の場合、測定開始から測定終了までの動作は連続測定の場合とまったく同様ですが測定が終了するとカウンタは最後のカウンタ値を保持して停止します。

なお、GATE 端子からの入力信号の立ち上がりから立ち下がりまでの期間、または立ち下がりから立ち上がりまでの期間は、システムクロックの2サイクル分以上にしてください。

### 8.10 モードごとのゲート入力によるカウンタ動作

モード	ゲート入力/カウンタ動作		
	“L” または立ち下がり	立ち上がり	“H”
分周	カウンタ停止(保持)	-----	カウンタ
パルス幅変調(PWM)	カウンタ停止(保持)	-----	カウンタ
パルス：ソフトトリガ ：ハードトリガ	カウンタ停止(保持)	-----	カウンタ
	カウンタ待機(続行)	カウンタ開始	-----
パルス幅/周期測定	測定対象波形の入力(パルス幅/周期測定モード参照)		

### 8.11 リセット

リセットはRESET\_入力“L”により次の機能に影響を与えます。

- (1)カウンタの動作を停止します。また、リセット解除後もカウンタは停止状態を保持します。
- (2)カウンタ内部のレジスタ読み出しの際のリードシーケンス、およびカウンタ初期値書き込みの際のライトシーケンスをクリアします。
- (3)カウンタ内部のCRレジスタ(カウンタ初期値書き込み用レジスタ)およびカウンタレジスタは‘FFFFH’にクリアされます。
- (4)カウンタ内部のORレジスタ(カウンタ値読み出し用レジスタ)は‘FFFFH’にクリアされます。
- (5)分周モード(システムクロック同期、トグル出力)となり、OUT出力を“L”に設定しリセット解除後もモード設定が行われるまで、その値を保持します。

### 8.12 注意事項

#### カウンタ初期値の最大/最小値

カウンタ初期値の最大/最小値を以下に示します。

モード	最小値	最大値
分周	0001H	FFFFH
パルス幅変調(PWM)*	0001H	0FFFFH
パルス	0001H	FFFFH
パルス幅/周期測定	設定不可	設定不可

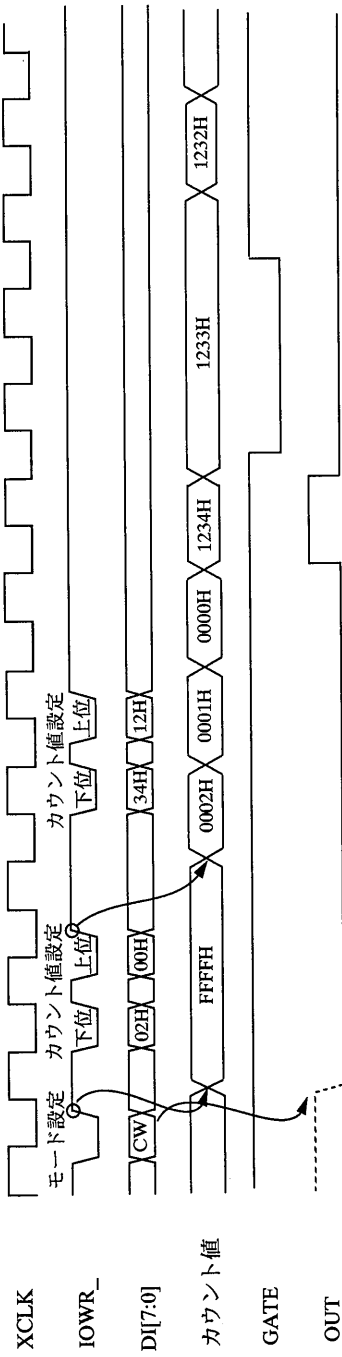
\* 有効な値はモード設定の PERIOD によって異なり、有効ビット幅は下位 6、8、10、12 のいずれかです。

#### 用語の定義

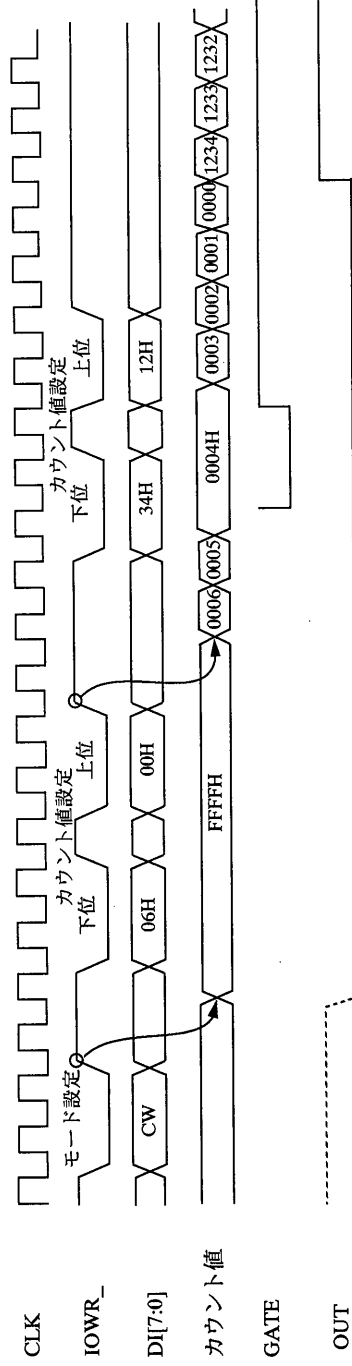
ハード・トリガ：ゲート入力の立ち上がりエッジ  
ソフト・トリガ：カウンタ初期値設定 (2バイト書き込み) 完了

### 動作モード

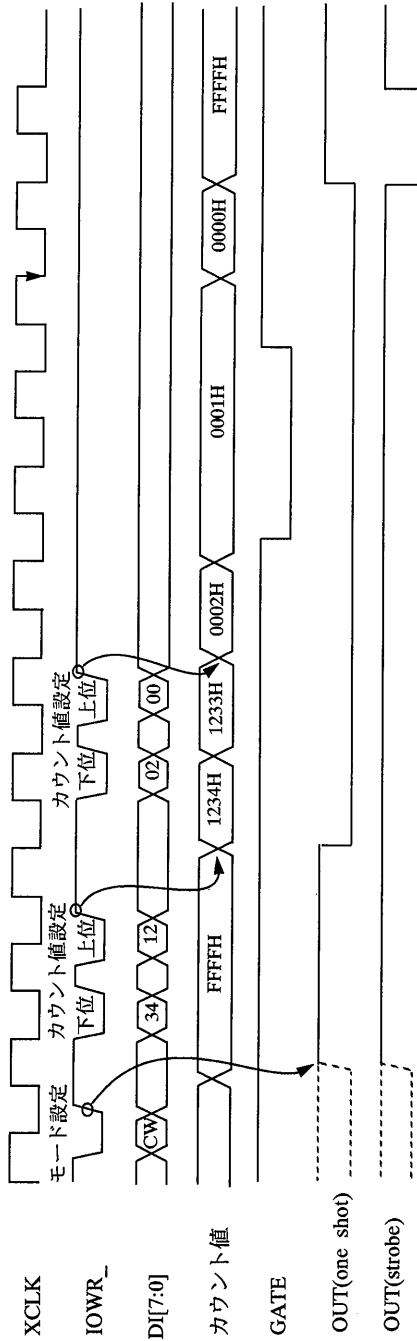
#### 分周モード (外部クロック、パルス出力)



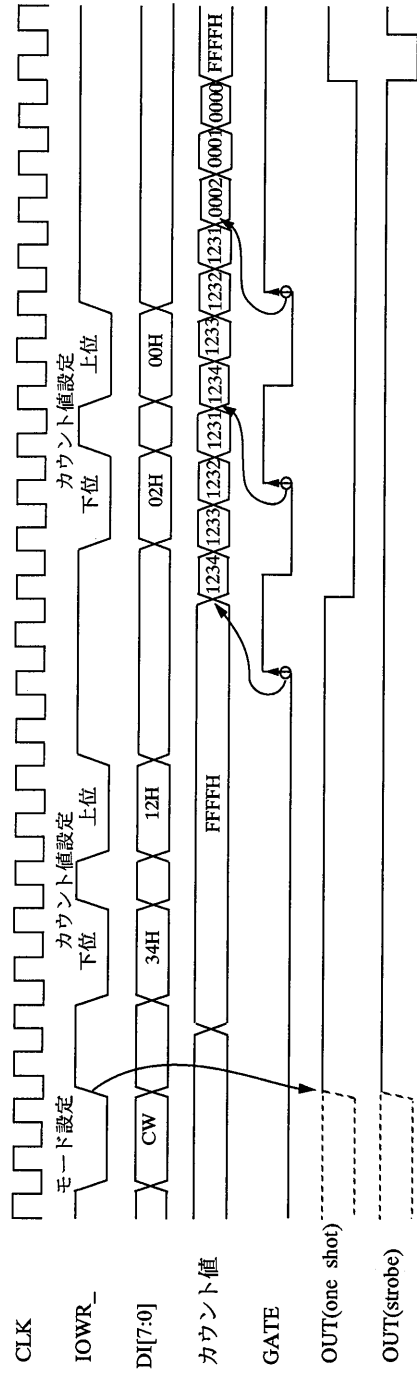
#### 分周モード (システムクロック、トグル出力)



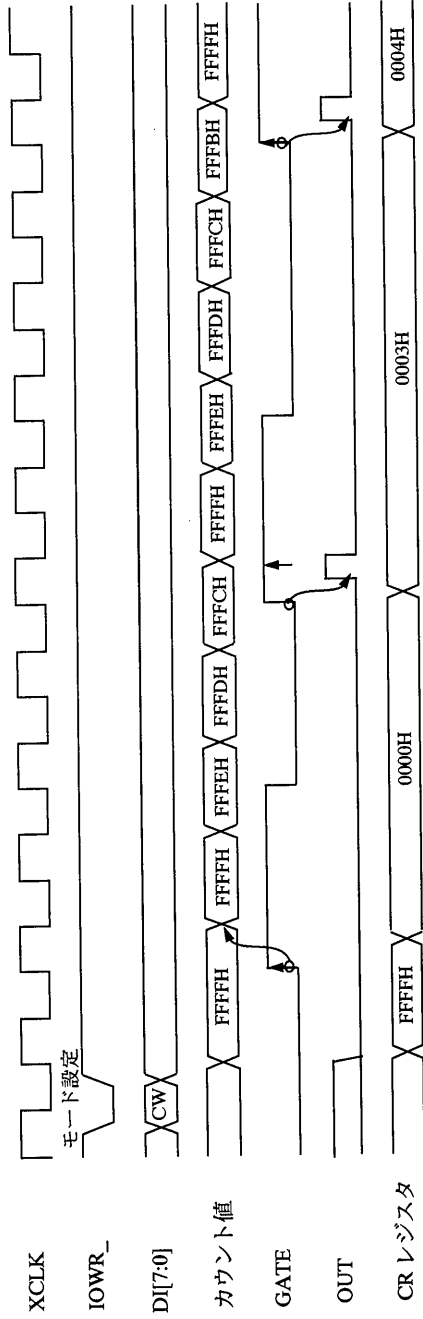
パルスモード (外部クロック、ソフトトリガ、REVERSE = 0)



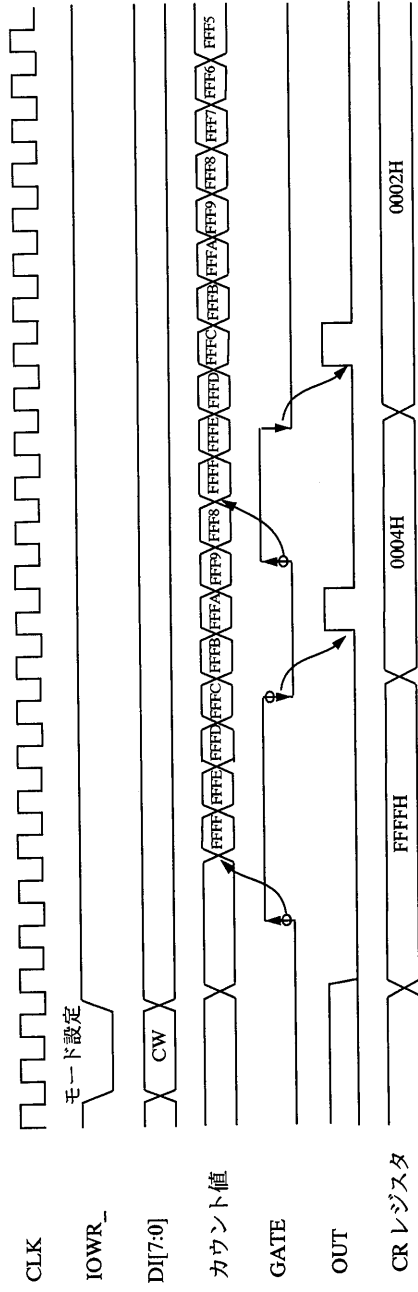
パルスモード (システムクロック、ハードトリガ、REVERSE = 0)

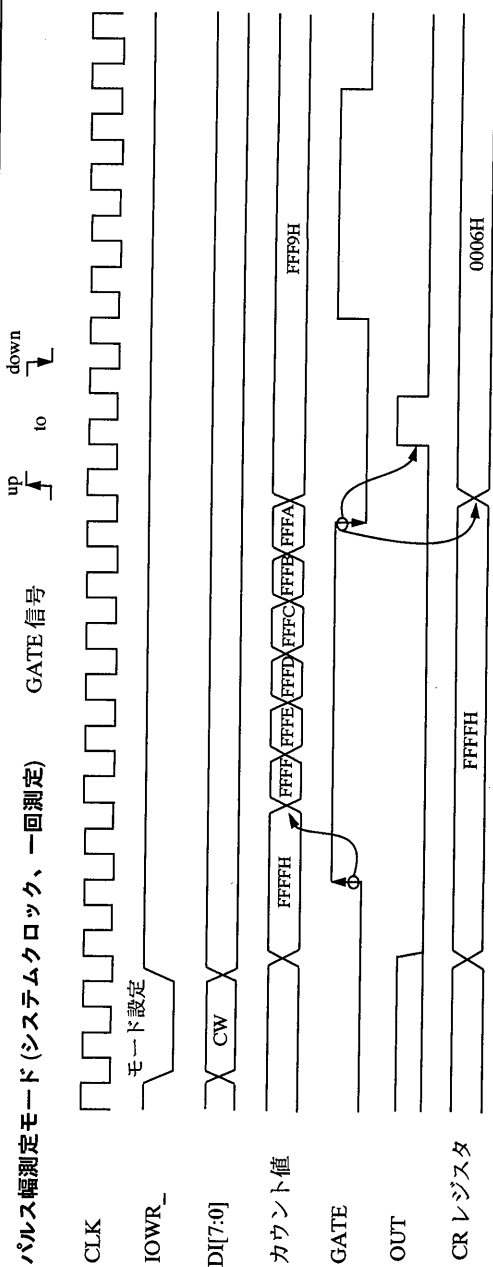


### パルス周期測定モード (外部クロック、連続測定)

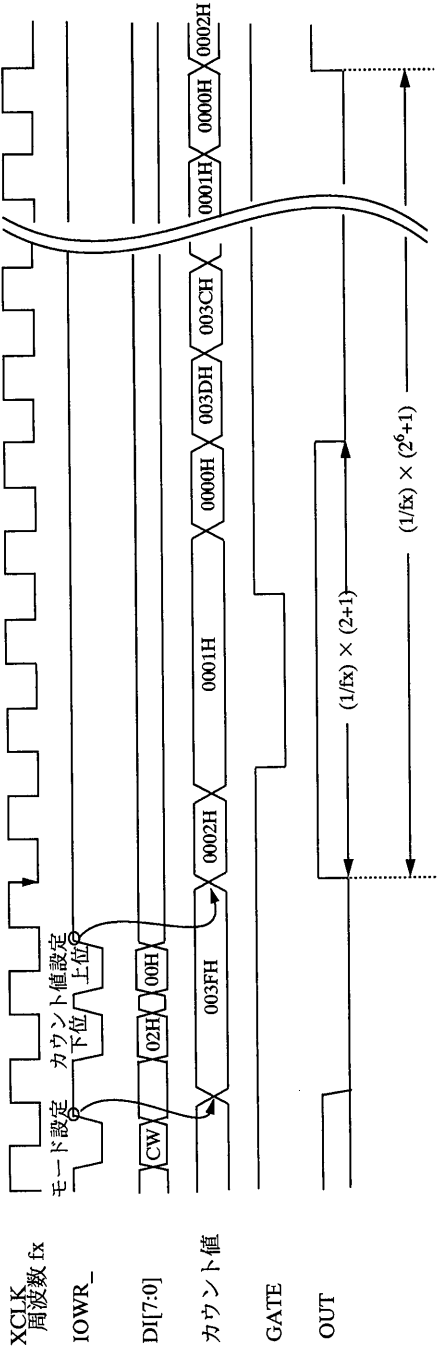


### パルス幅測定モード (システムクロック、連続測定)

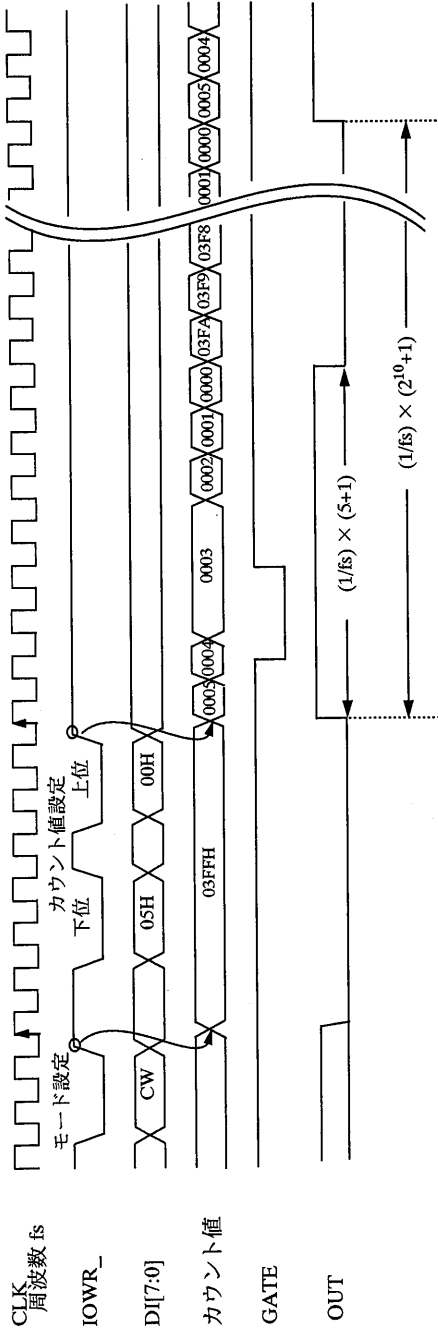




PWM モード (外部クロック、 $(2^6+1)$  倍周期)



PWM モード (システムクロック、 $(2^{10}+1)$  倍周期)



### 9. タイマ/カウンタ B

#### 9.1 概要

KL5C8012 のタイマ/カウンタ B は 3 チャンネル版の KP63(16 ビット高機能タイマ/カウンタマクロセル) です。KP63 はマイクロコンピュータ用に設計された 8 ビットプリスケータつきクロック同期式 16bit プログラマブル・バイナリカウンタ/タイマです。本マクロセルは、3 種類の動作モード (パルス幅変調 (PWM) モード、パルス発生モード、ウォッチドックタイマモード) をもつダウンカウンタ 3 個で構成され、CPU からのモード制御命令によりモード設定できます。カウント値は、バスから読み出すことができます。また、ステータスリードコマンドで設定モードや OUT 出力端子の状態を読み出すことができます。

**注意** KP63 は本来 4 チャンネル構成ですが、KL5C8012 のタイマ/カウンタ B は 3 チャンネルに変更したものです。本章では 4 チャンネル構成として説明していますが、チャンネル 3 の機能は使わないようご注意ください。

#### 特徴

- ・ 8 ビットプリスケータつき。
- ・ 3 個の 16bit ダウンカウンタ/タイマを内蔵し各々 3 種類の動作モード設定、動作が可能
- ・ カウント動作に影響を与えずに、安定したカウント値の読み出しが可能
- ・ 設定モード、OUT 出力の状態等のステータスの読み出しが可能

#### 9.2 端子機能説明

端子機能	I/O	機能説明
GATE[3:0]	I	ゲート入力端子 各チャンネルのカウンタに対するカウント動作イネーブル/ディセーブルの指定を与えます。
OUTP[3:0]	O	パルスアウト出力端子 モードの指定によって、分周出力、PWM 出力が得られます。リセット信号入力により本出力はシステムクロックとは非同期に“L”になります。
OUTS[3:0]	O	ストローブアウト出力端子 4システムクロック幅のストローブを出力します。
SYNC	O	シンク信号出力端子 システムクロックを 4 分周した信号であり、この信号の立ち上がりで GATE[3:0] をサンプルします。

**注意)** 各チャンネルの GATE, OUTP, OUTS の各端子は外部端子ではそれぞれ GATEB2, GATEB1, GATEB0, OUTP2, OUTP1, OUTP0, OUTS2, OUTS0 の名前になっています。ブロック図で外部端子名を確認してご使用下さい。

### 9.3 ブロック図

タイマ/カウンタ B の全体ブロック図と I/O レジスタマップを示します。  
各 OUTS 出力は外部端子以外に KL5C8012 内蔵の割り込みコントローラにも接続されています。

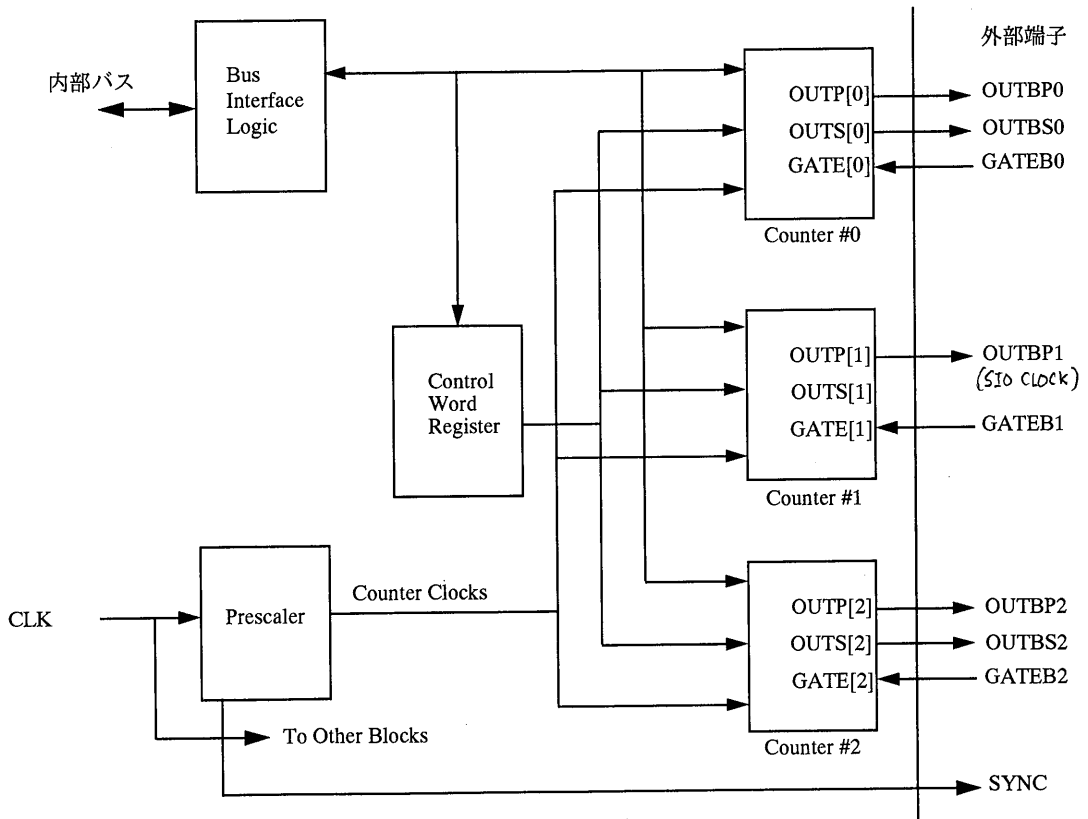


図 9-1 タイマ/カウンタ B のブロック図

表 9-1 I/O レジスタマッピング

I/O アドレス	ブロック名	ライト時	リード時
20H	タイマ/カウンタB	チャンネル0カウンタ	チャンネル0カウンタ
21H		チャンネル0コントロールワード	チャンネル0ステータス
22H		チャンネル1カウンタ	チャンネル1カウンタ
23H		チャンネル1コントロールワード	チャンネル1ステータス
24H		チャンネル2カウンタ	チャンネル2カウンタ
25H		チャンネル2コントロールワード	チャンネル2ステータス
26H		川崎製鉄使用予約	川崎製鉄使用予約
27H		川崎製鉄使用予約	川崎製鉄使用予約

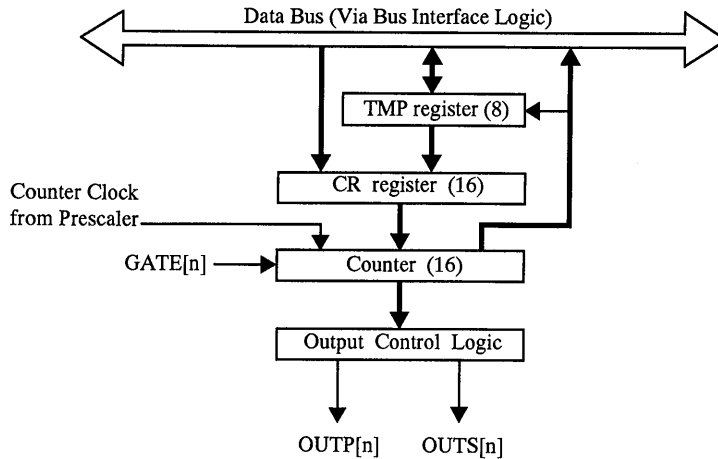
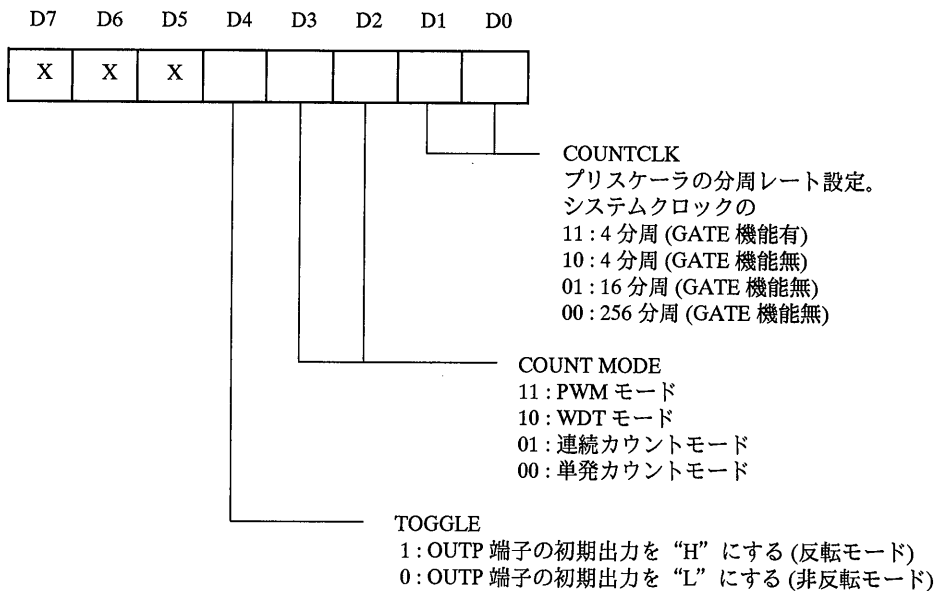


図 9-2 各チャンネルカウンタ・ユニット内のブロック図

### 9.4 モード設定

モード設定は各チャンネルにコントロールワードを書き込むことで行われます。



プリスケーラは全チャンネルで共有していますが、プリスケーラの分周レートは各チャンネル毎に設定できます。

### 9.5 カウンタへのカウント初期値の設定

カウンタは 16bit 長のダウンカウンタですが、データバスが 8bit 幅なので、データ書き込みは下位バイト、上位バイトの順に 2 回に分けて行います。ただし、書き込み用と読み込み用で TMP レジスタを共用しているので、書き込みを完了する前にカウンタの読み出しを行うと、書き込みデータが壊れます。

下位側：

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(数字はデータビットを表す)

上位側：

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(数字はデータビットを表す)

### 9.6 カウンタの読み出し

カウンタの読み出しは、各カウンタのアドレスをアクセスして読み出すことになります。その際、8bit 幅で 2 回に分けて、下位バイト、上位バイトの順で行います。上位側は下位側を読み出す時に TMP レジスタに移されたカウンタの上位バイトなので、2 回の読み出し動作中にカウンタの値が変化しても、読み出される値は変化しません。ただし、読み出し用 TMP レジスタと書き込み用 TMP レジスタを共用しているので、読み出しを完了する前にカウント初期値の書き込みを行うと、読み出しデータが壊れます。

下位側：

D7	D6	D5	D4	D3	D2	D1	D0
7	6	5	4	3	2	1	0

(数字はデータビットを表す)

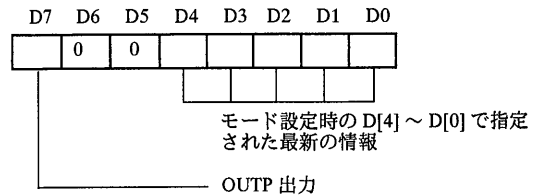
上位側：

D7	D6	D5	D4	D3	D2	D1	D0
15	14	13	12	11	10	9	8

(数字はデータビットを表す)

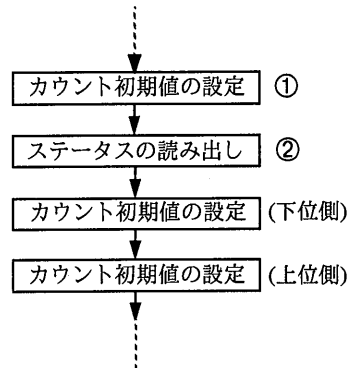
### 9.7 ステータスの読み出し

ステータスは各チャンネルごとに用意されています。ステータスのフォーマットは以下の通りです。



なお、ステータスの読み出しを行うと、リードライト・シーケンスがクリアされます(下例参照)。したがってカウント初期値の設定やカウンタの読み出しを行う前に一度ステータスの読み出しを行うことにより、誤りを防ぐことができます。

(例)



①は上位側、下位側、どちらでも良い。  
②でリードライト・シーケンス・クリアされる。

## 9.8 動作モード

以下の説明はモード設定時、OUTP非反転モード選択の場合について行います。

### 連続カウントモード

連続カウントモードでは、カウント初期値をロードした後デクリメントし、カウント値がゼロになるとOUTP出力とOUTS出力が変化すると共に初期値をリロードした後デクリメントを繰り返します。

モード設定によって、OUTP および OUTS 端子は“L”を出力します。カウント動作はカウント初期値をロードすることで開始し、OUTP 端子に“H”を出力します。カウント値がゼロになる度にOUTP出力はトグルし、OUTS 端子には4システムクロック幅の“H”が出力されます。

リロードはカウント値がゼロになることでのみ発生し、カウント中に初期値を書き換えてもリロードは発生しません。

### 単発カウントモード

単発カウントモードでは、カウント初期値をロードした後デクリメントし、カウント値がゼロになるとOUTPおよびOUTS出力が変化し、カウント動作は次にカウント初期値が設定されるまで停止します。

モード設定によってOUTPおよびOUTS端子は“L”を出力します。カウント動作はカウント初期値の設定により開始し、OUTP 端子はカウント値がゼロになるまで“H”を出力します。OUTS 端子はカウント値がゼロになると4システムクロック幅の“H”を出力します。

リロードは、カウント初期値の設定により行われます。

### ウォッチドッグタイマ(WDT)モード

ウォッチドッグタイマモードでは、CRレジスタ(カウント初期値書き込み用レジスタ)への書き込みはできません。CRレジスタへの書き込み動作はカウント開始およびリトリガとして認識されます。

モード設定により、カウンタは動作を停止し、OUTP および OUTS 端子は“L”を出力します。カウント動作はCRレジスタへの書き込み動作(書き込み動作時のデータ値には無関係;1回のライト動作)により、CRレジスタからデータをロードして開始し、カウントを開始すると、OUTP 端子は“H”を出力します。

また、カウント中にCRレジスタへの書き込み動作で再び、CRレジスタからデータをロードしてカウン

トダウンを行います。

ただし、一旦カウント値がゼロになると、OUTP 端子は、“L”に、OUTS 端子は4システムクロック幅の“H”を出力し、次にCRレジスタへの書き込み動作が行われるまでカウント動作は停止します。

なお、モード設定では、CRレジスタが初期化されません。CRレジスタにカウント初期値を設定するには次の様な手順を用います。すなわち、単発カウントモードにモード設定後、CRレジスタにカウント初期値を設定し、再度モード設定によりWDTモードにします。そして、CRレジスタへの書き込み動作を行うと、先に設定したCRレジスタの値をロードして、カウントダウンを開始します。

また、カウント中にCRレジスタへの書き込み動作を行うことにより、再度CRレジスタの値をロードしてカウントダウンを実行します。

以上の手順によれば、WDTモードにおいて、任意のカウント初期値を設定することが可能となります。

なお、本タイマ/カウンタをウォッチドッグタイマとして使用する場合はこのモード設定をするだけでなく、OUTSあるいはOUTP出力をKC82のNMI\_入力に外部で接続する必要があります。

### パルス幅変調(PWM)モード

パルス幅変調モードでは、カウント初期値により設定する、パルス幅およびパルス周期により構成される繰り返しパルスをOUTP端子に出力します。

パルス幅はCRレジスタの上位バイトで、パルス周期はCRレジスタの下位バイトによって設定します。まず、パルス周期の設定に関して説明します。

パルス周期は、CRレジスタの下位側バイトをカウンタにロードした後、上位バイトと独立にデクリメントし、下位バイトのカウント値がゼロになるまでの時間となります。すなわち、(1回目のCRレジスタへの書き込みにおける設定値+1)の時間を意味します。

一方、パルス幅はCRレジスタの上位側バイトをカウンタにロードした後、下位バイトと独立にデクリメントし、上位バイトのカウント値がゼロになるまでの時間となります。すなわち、(2回目のCRレジスタへの書き込みにおける設定値+1)の時間を意味します。

このように上位、下位ともに同時にカウントダウンを独立に行いますので、任意のパルスがOUTP端子から得られることとなります。

モード設定によってカウンタは停止し、OUTP 端子およびOUTS 端子は“L”を出力します。カウント動作はカウント初期値を上位、下位共に設定後、ロードすることにより開始し、カウントダウンは上位バイト、下位バイトずつ各々独立に行います。ただし、設定した値が(パルス幅≧パルス周期)の場合には、常にOUTP=“H”となります。

リロードは下位バイトがゼロになることでのみ発生し、カウント中に初期値を書き込むことによるリロードは発生しません。

(例:4分周設定時、カウント初期値 = '0308H' とした場合)

パルス幅は  $4(=3+1) \times 4=16$  システムクロック  
 パルス周期は  $9(=8+1) \times 4=36$  システムクロック  
 となります。)

### 9.9 OUTPおよびOUTS出力について

以下の説明はモード設定時、OUTP 非反転モード選択の場合について行ないます。

OUTS 出力は、カウント値がゼロ (PWM モードでは下位バイトがゼロ) になると、4システムクロック幅の“H”を出力します。このようにOUTS出力はCPUへの割り込み等に利用できます。なお、OUTS出力は反転/非反転モードの選択ができません。

OUTP 出力は、モード設定により反転が可能な信号です。連続モード時には、カウント値がゼロになる度にトグルします。単発モードおよび WDT モードでは、通常時カウント中に“H”を出力し、カウント値がゼロになると“L”を出力します。PWM モードでは、所望のパルス幅および周期のパルス信号を出力します。以上の様に OUTP 出力は様々なパルス信号を得る為に利用することが可能です。

### 9.10 動作説明

ここでは、タイマ/カウンタ B の動作に関し、図を用いて説明します。

#### カウントダウンタイミングと SYNC 信号 (図 9-3-A)

図 9-3-A について説明します。タイマ/カウンタ B はカウントダウンタイミングに同期した信号として、SYNC 信号を出力します。図 9-3-A は、全てのチャンネルを GATE 機能無の 4 分周に設定した場合です。

図中にあるように、カウンタチャンネル 1 (以下 CH1 と略記する) は CH0 に対し、カウントダウンは 1 システムクロック遅れた、システムクロックの立ち下がりで行われます。同様に CH2、CH3 はそれぞれ CH0 に対して、2 システムクロック、3 システムクロック遅れることになります。

SYNC 信号は基本的にシステムクロックを 4 分周した信号であり、システムクロックとの関係は図 9-3-A にある通りです。

#### 各カウントクロック選択時のカウントダウンタイミング (図 9-3-B)

図 9-3-B は CH0、CH1 に GATE 機能無の 4 分周、CH2 に 16 分周、CH3 に 256 分周のカウントクロックを設定した場合です。

CH0、CH1 では、それぞれ 1 から 1'、2 から 2' 迄の 1 回のカウントダウン迄の時間が 4 システムクロックであり、3 から 3' 迄が 16 システムクロック、4 から 4' 迄が 256 システムクロックとなります。

#### GATE 信号のサンプルタイミング (図 9-3-C)

図 9-3-C は全チャンネルに GATE 機能有の 4 分周を設定した場合です。GATE 信号のサンプルは、4 システムクロックに 1 回 (図中 CLK の↑時; SYNC 信号の立ち上がり)、全チャンネル分を同時に行います。したがって、この時サンプルした GATE 信号の極性により、次のカウントダウンタイミングでカウントダウンするかしないかが判断されます。GATE 機能が使えるのは 4 分周の分周レートのみです。

#### カウントダウンタイミングとコントロールワードの書き込み (図 9-4 ~ 9-7)

CH0、CH1、CH2 および CH3 に関するカウントダウンタイミングとコントロールワード書き込みの関係について、それぞれ図 9-4、9-5、9-6、9-7 に示します。

各図の左半分はコントロールワード書き込みによるモード設定が反映される迄の時間、すなわち OUTP、OUTS 信号が初期化される迄の時間が最も長い場合です。

一方、各図の右半分は最も短い場合を示します。

#### カウントダウンタイミングとカウント値書き込み (図 9-8, 9-9, 9-10, 9-11)

連続カウントモードと PWM モードの場合の CH0、CH1、CH2、CH3 に関するカウントダウンタイミングとカウント値書き込みの関係について、それぞれ図 9-8、9-9、9-10、9-11 に示します。

各図の左半分は上位カウント値設定がカウンタに反映される迄の時間が最も長い場合であり、各図の右半分は最も短い場合を示します。

#### 連続カウントモードの動作説明 (図 9-12)

連続カウントモードについて、GATE 機能無の 4 分周、OUTP 非反転モード選択時を例にとり、図 9-12 を用いて説明します。これは CH0 の場合です。

モード設定を行うと、次のカウントダウンタイミングで OUTP および OUTS 信号が初期化され、“L”に

なります。下位カウント値 '02H'、上位カウント値 '00H' を書き込むと、次のカウントダウンタイミングでカウンタに '0002H' がロードされ、カウントダウンを開始すると共に、OUTP 出力がトグルします。

カウント中、すなわち図中の 1 から 2 の間にカウント初期値 'F0ABH' を再設定すると、次にカウント値がゼロとなった次のカウントダウンタイミング、すなわち 2 の時点でカウント値が 'F0ABH' になります。

また、それと同時に OUTP 出力がトグルし、OUTS 端子に 4 システムクロック幅のパルスが出力されます。

### 単発カウントモードの動作説明 (図 9-13)

単発カウントモードについて、GATE 機能無の 4 分周、OUTP 非反転モード選択時を例にとり、図 9-13 を用いて説明します。これは CH0 の場合です。

モード設定を行うと、次のカウントダウンタイミングで OUTP および OUTS 信号が初期化され、"L" になります。下位カウント値 '02H'、上位カウント値 '00H' を書き込むと、次のカウントダウンタイミングでカウンタに '0002H' がロードされ、カウントダウンを開始すると共に OUTP 出力が "H" となります。

カウント中、すなわち図中の 1 から 2 の間にカウント初期値 '0003H' を再設定すると、次のカウントダウンタイミング (図中の 2) で、リトリガがかかります。そして、この再設定値からカウントダウンが続きます。そして 2 から 3 の間にカウント初期値の再設定がなく、カウンタがゼロになると (図中の 3) 'FFFFH' となり、カウントダウンを停止し、OUTP 出力は "L" に、また OUTS 端子には 4 システムクロック幅の "H" が出力されます。

なお、1 にあるような場合、すなわちカウント値がゼロから 'FFFFH' になるタイミングでカウント初期値 'ABCDH' の再設定を認識した場合にはリトリガと判断し、OUTP および OUTS 信号は変化しません。

### ウォッチドッグタイマ (WDT) モードの動作説明 (図 9-14)

ウォッチドッグタイマ (WDT) モードについて GATE 機能無の 4 分周、OUTP 反転モード選択時を例にとり、図 9-14 を用いて説明します。これは CH0 の場合です。

WDT モードでは、カウント初期値の書き込み (CR レジスタへの書き込み) 動作がリトリガと解釈され、カウント初期値レジスタの値は変化しません。カウント初期値レジスタに所望の値を設定するためには、一旦、単発カウントモードに設定し、CR レジスタに値を書き込み、WDT モードに設定し直します。

WDT モードの基本動作は単発カウントモードに準じます。図 9-14 の説明を行います。まず、単発カウ

ントモードにモード設定を行うと、次のカウントダウンタイミングで OUTP 出力が "H" に、OUTS 出力が "L" になります。下位カウント値 '03H'、上位カウント値 '00H' を書き込むと、次のカウントダウンタイミングでカウンタに '0003H' がロードされ、カウントダウンを開始すると共に OUTP 出力が "L" になります。

次に WDT モードに再度モード設定を行います (図中の 1)。すると、次のカウントダウンタイミングで OUTP 出力が "H" になり、このときカウンタは停止状態です。次にカウント値書き込み動作 (データは任意) を 1 行くと、次のカウントダウンタイミングでカウンタに CR レジスタの値 '0003H' がロードされ、WDT モードとして初めて、カウントダウンを開始すると共に OUTP 出力が "L" となります (図中の 2)。

カウント中、すなわち図中の 2 から 3 の間にカウント書き込み動作 (データは任意) を行うと、リトリガとして認識され、次のカウントダウンタイミングで再び '0003H' をロードします (図中の 3)。そしてまたカウントダウンを開始し、カウント値がゼロになると、次のカウントダウンタイミングで OUTS 端子に 4 システムクロック幅の "H" を、OUTP 端子に "H" を出力し、カウント値を 'FFFFH' にクリアし、停止します (図中の 4)。

その後、カウント値書き込み動作 (データは任意) を 1 行くと、次のカウントダウンタイミングでカウンタに '0003H' がロードされ、再度カウントダウンを開始します。

OUTP 出力は、一旦単発カウントモードに設定することにより、WDT としては不要な変化をしますが、OUTS 端子はカウント値がゼロになるまで、パルスを出力しません。

詳細な使用例は別冊のアプリケーションノートを参照して下さい。

### パルス幅変調 (PWM) モードの動作説明 (図 9-15)

パルス幅変調 (PWM) モードについて、GATE 機能無の 4 分周、OUTP 非反転モード選択時を例にとり、図 9-15 を用いて説明します。これは、CH0 の場合です。

基本動作は連続カウントモードに準じます。モード設定を行うと、次のカウントダウンタイミングで OUTP および OUTS 信号が初期化され "L" になります。下位カウント値 (パルス周期データ) '03H'、上位カウント値 (パルス幅データ) '01H' を書き込むと、次のカウントダウンタイミングで、カウンタに '0103H' がロードされ、カウントダウンを開始すると共に、OUTP 出力がトグルします。

図にあるように、カウントダウンは上位バイト、下位バイトで独立に、かつ同時に行われ、上位側がゼロ

になると OUTP 出力がトグルします。1 回のカウンタダウンは 1 から 2、すなわちカウンタ初期値がロードされてから、下位側がゼロとなった後までを示します。

このように、1 から 2 の間において、パルス幅  $(1+1)/fc = 2/fc$ 、パルス周期  $(3+1)/fc = 4/fc$  ( $fc$  はカウンタダウ周波数) のパルスが OUTP 端子に出力されます。2 の時点で OUTS 端子に 4 システムクロック幅のパルスが出力されます。

また、カウンタ中にカウンタ初期値 '0204H' を再設定すると次に下位のカウンタ値がゼロとなった次のカウンタダウタイミング、すなわち 2 の時点でカウンタ値として '0204H' がロードされます。

また、上位バイトと下位バイトに等しい、あるいは、上位バイトに下位バイトよりも大きい値を設定した場合は、3 から 4 のように OUTP 端子は常に "H" を出力します。下位バイトがゼロになった次のカウンタダウサイクル (図中の 4) において、カウンタにはカウンタ初期値 '0505H' がロードされ、再びカウンタダウが連続して実行されます。

以上のように、タイマ/カウンタ B の PWM モードでは繰り返し周期を任意に選択することが可能です。

### 9.11 リセット

本リセットは RESET\_ 入力 "L" により次の機能に影響を与えます。

- (1) カウンタの動作を停止します。また、リセット解除後もカウンタは停止状態を保持します。
- (2) カウンタ内部のレジスタ読み出しの際のリードシーケンス、およびカウンタ初期値書き込みの際のライトシーケンスをクリアします。
- (3) カウンタ内部の CR レジスタおよび カウンタレジスタは 'FFFFH' にクリアされます。
- (4) 単発カウンタモードで 256 分周カウンタクロック OUTP 出力反転非選択となり、OUTP および OUTS 出力を "L" に設定し、リセット解除後もモード設定が行われるまで、その値を保持します。

### 9.12 注意事項

#### カウンタ初期値の最大/最小値

カウンタ初期値の最大/最小値を以下に示します。

モード	最小値	最大値
連続カウンタ	0001H	FFFFH
単発カウンタ	0001H	FFFFH
パルス幅変調(PWM)	上位:01H 下位:01H	上位:FFH 下位:FFH
ウォッチドッグタイマ (WDT)	設定不可*	設定不可*

\* 他のモードで設定後、WDT モードに再設定した上で、使用する際は、カウンタ初期値設定時のモードでの条件に従う。

図 9-3-A カウントダウンタイミング (GATE 機能無、4 分周) および SYNC 信号

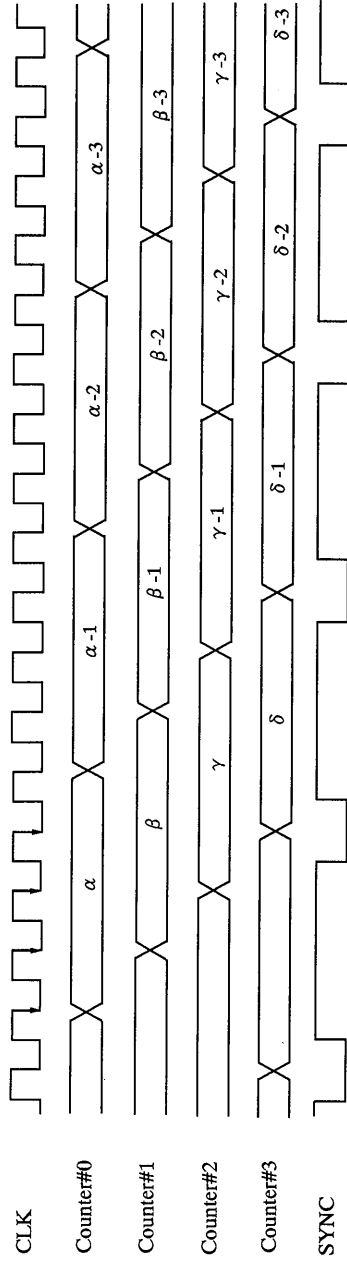


図 9-3-B カウントダウンタイミング (#0,1:GATE 機能無 4 分周,#2:16 分周,#3:256 分周選択時)

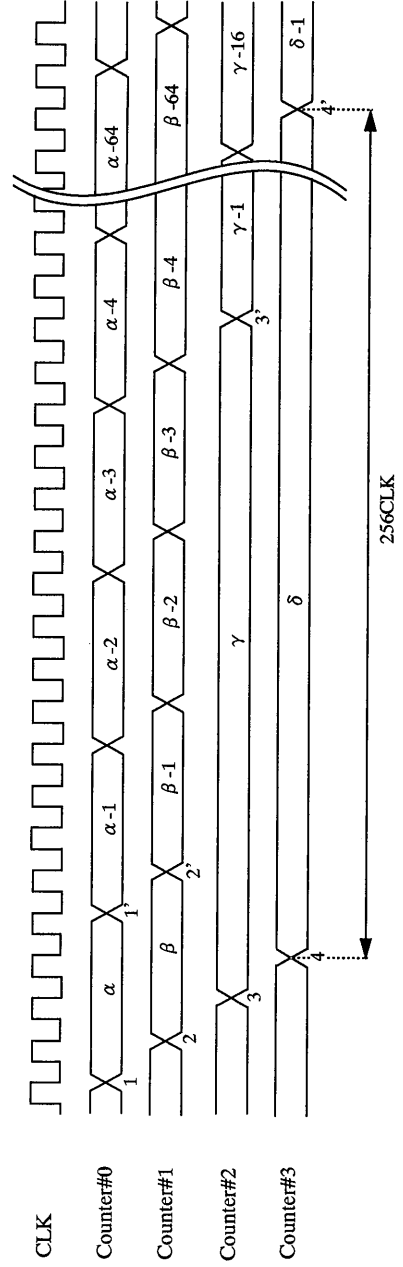


図 9-3-C GATE 信号のサンプルタイミング (GATE 機能有 4 分周)

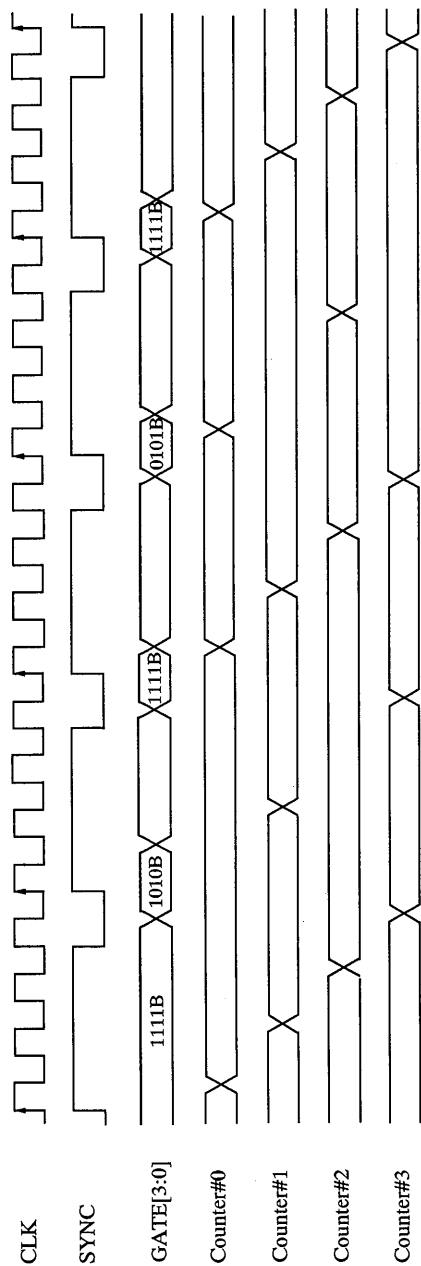


図 9-4 カウントダウンタイミングとコントロールワード書き込み (#0 の場合)

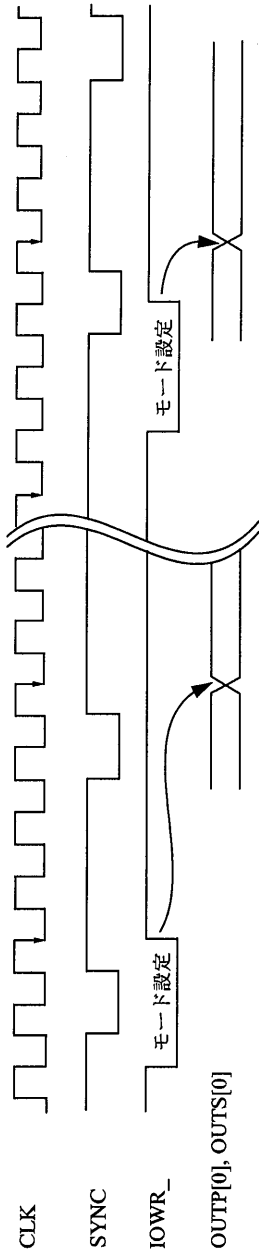


図 9-5 カウントダウンタイミングとコントロールワード書き込み (#1 の場合)

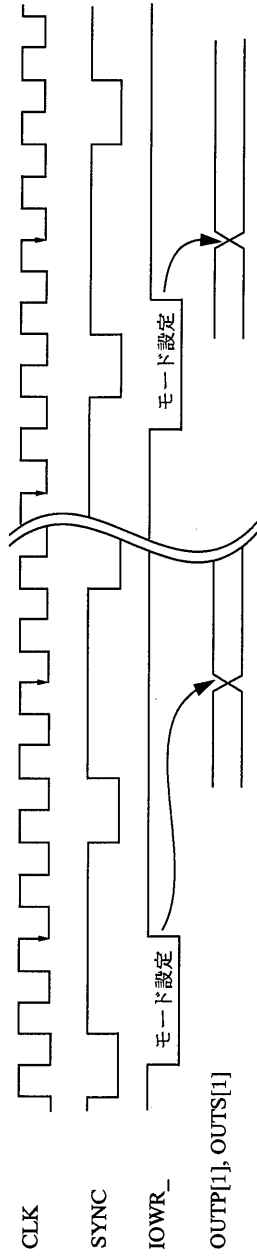


図 9-6 カウントダウンタイミングとコントロールワード書き込み (#2 の場合)

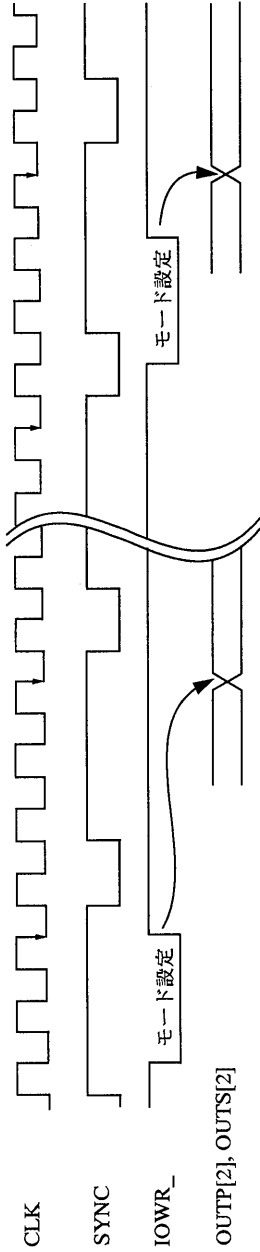


図 9-7 カウントダウンタイミングとコントロールワード書き込み (#3 の場合)

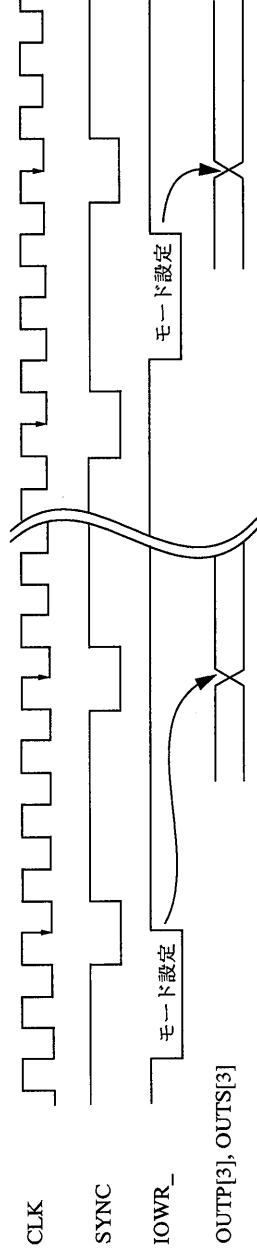


図 9-8 カウントダウンタイミングとカウンタ値書き込み (#0 の場合)

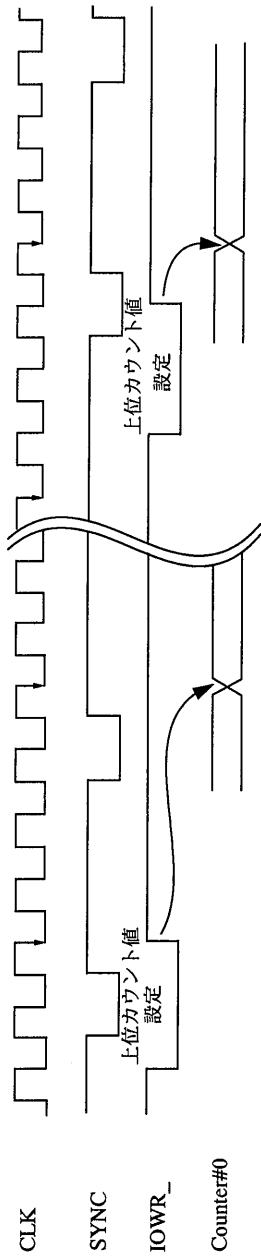


図 9-9 カウントダウンタイミングとカウンタ値書き込み (#1 の場合)

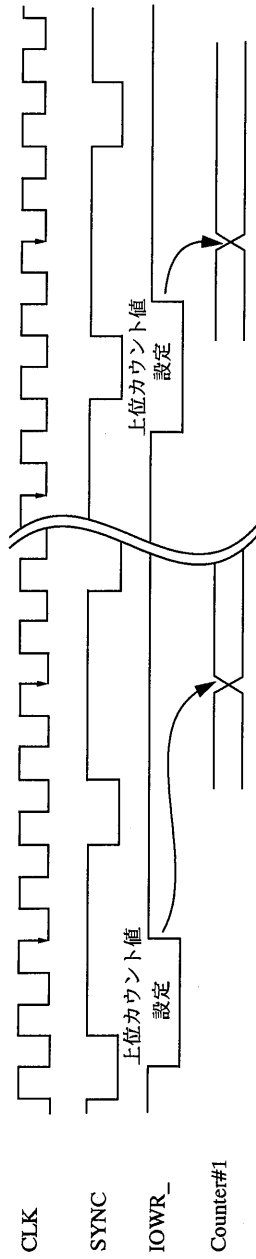


図 9-10 カウントダウンタイミングとカウント値書き込み (#2 の場合)

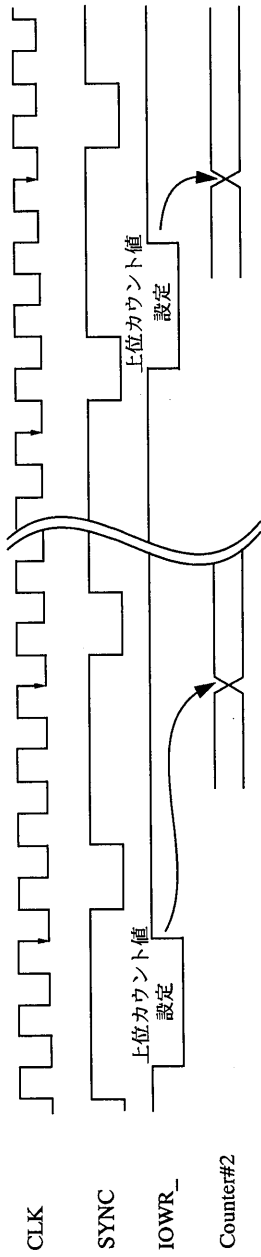


図 9-11 カウントダウンタイミングとカウント値書き込み (#3 の場合)

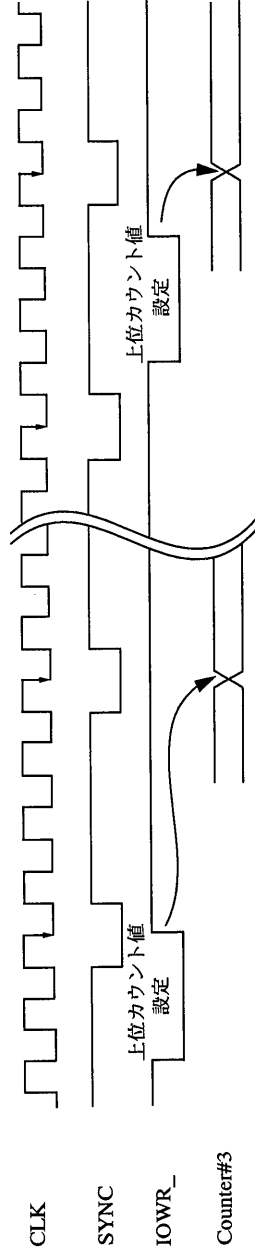


図 9-12 連続カウントモード (GATE 機能無 4 分周、OUTP 非反転モード選択時)

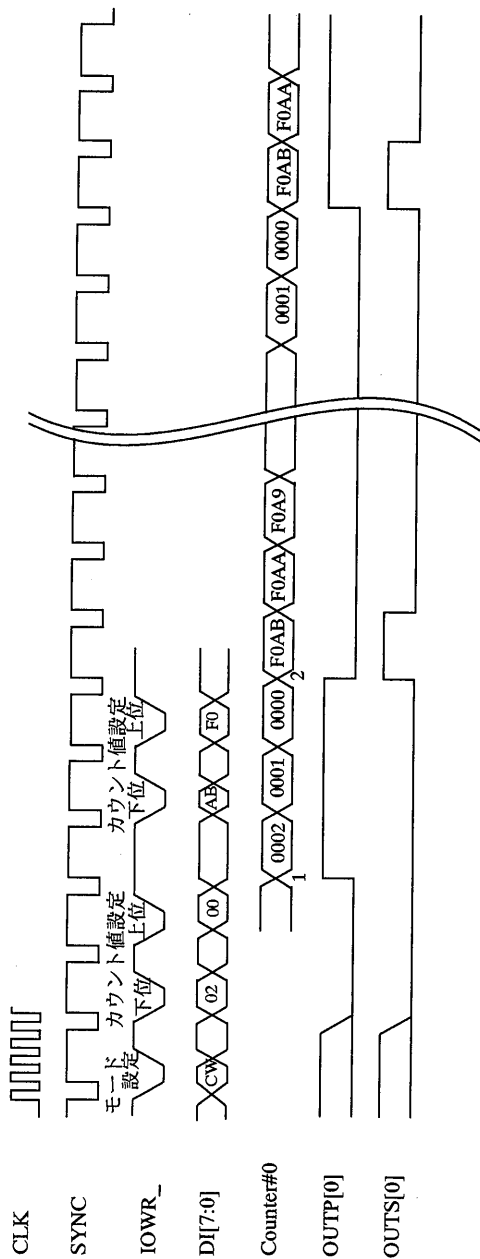
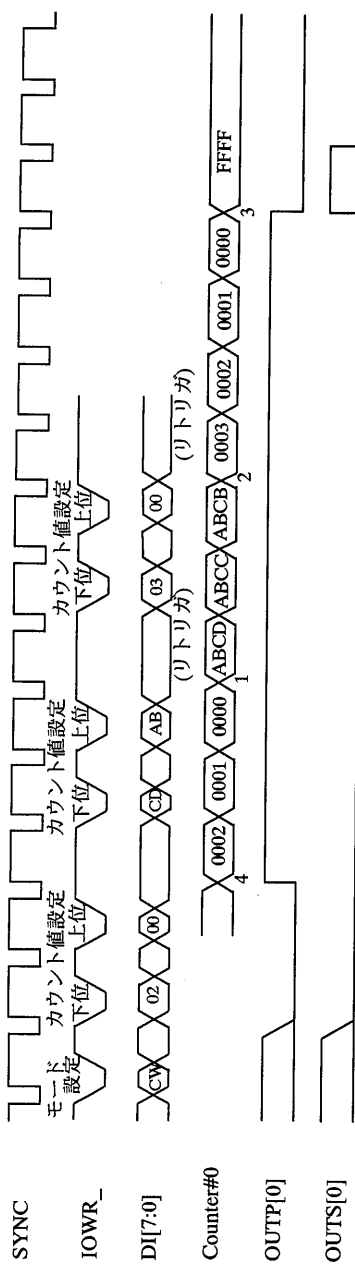


図 9-13 単発カウントモード (GATE 機能無 4 分周、OUTP 非反転モード選択時)





### 10. パラレルポートA

本チップの平行ポートAはKP65マクロセルで構成されています。KP65はKC80あるいはKC82をベースとしたASICマイコンの平行ポート用に開発されたマクロセルです。ポート数は16本で1ビット単位に入力/出力の方向制御が可能で、この設定はプログラムによるレジスタ設定で行なわれます。同様のマクロセルとしてKP66がありますが、KP66との違いは下記の点です。

#### 特徴

##### KP65

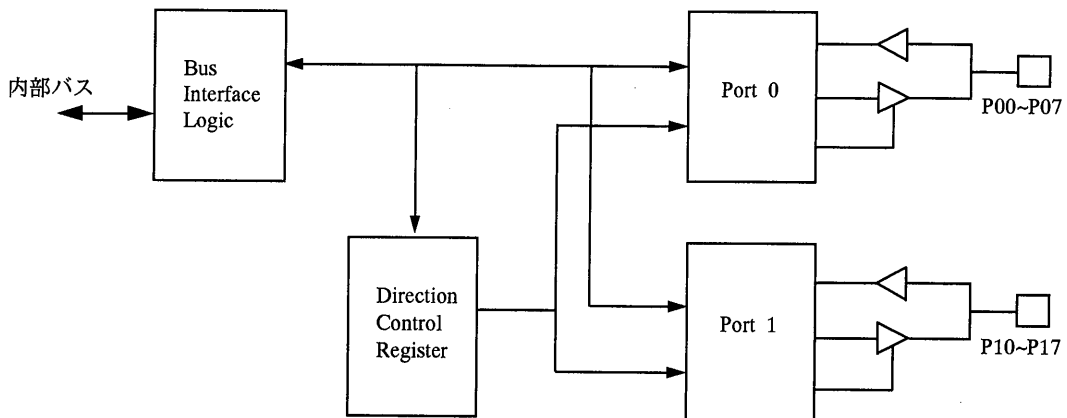
- ・ 16本の平行ポート搭載
- ・ 1ビット単位の入力/出力の方向制御が可能

##### KP66

- ・ 24本の平行ポート搭載
- ・ 4ビット単位の入力/出力の方向制御が可能
- ・ ソフトによりビット単位のセット/リセットが可能

#### 10.1 ブロック図

KP65の全体ブロック図とI/Oレジスタマッピングを以下に示します。



I/Oレジスタマッピング

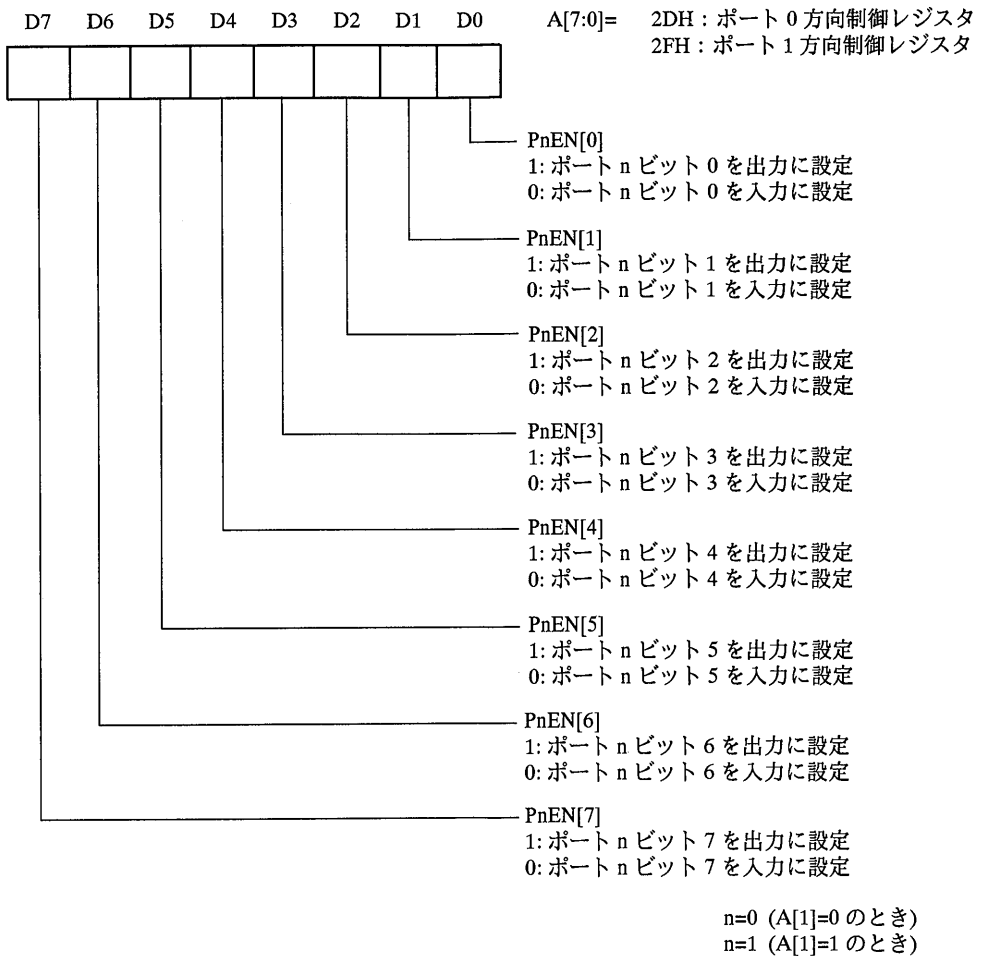
I/O アドレス	ブロック名	ライト時	リード時
2CH	パラレルポートA	ポート0	ポート0
2DH		ポート0方向制御レジスタ	ポート0方向制御レジスタ
2EH		ポート1	ポート1
2FH		ポート1方向制御レジスタ	ポート1方向制御レジスタ

### 10.2 ポート入出力設定

KP65 のポート入出力方向設定は I/O アドレス =2DH, 2FH の各ポート方向制御レジスタ書き込みで行います。

#### ポート方向制御レジスタ

KP65 は 16 本のパラレルポートの入出力の方向設定を各ポートの方向制御レジスタにより 1 ビット単位で設定できます。



### 10. 3 ポート入出力動作

#### ポート入力動作

入力にモード設定されたポートでは双方向バッファからとりこんだデータをデータバスに出力します。

#### ポート出力動作

出力に設定されたポートではデータバスから各ポートに書き込まれたデータが双方向バッファから出力されます。

### 10. 4 リセット

リセット時には以下のレジスタが初期化されます。

ポート 0 全ビット	リセット
ポート 1 全ビット	リセット
ポート 0 方向制御レジスタ全ビット	リセット
ポート 1 方向制御レジスタ全ビット	リセット

### 10. 5 使用上の注意

他の信号とマルチプレクスされているパラレルポートをパラレルポートとして使用しない場合、そのパラレルポートは入力方向に設定して下さい。

### 11. パラレルポートB

本チップの平行ポートBはKP66マクロセルで構成されています。KP66はKC80あるいはKC82をベースとしたASICマイコンの平行ポート用に開発されたマクロセルです。ポート数は24本で4ビット単位に入力／出力の方向制御が可能で、この設定はプログラムによるレジスタ設定で行なわれます。またプログラムによるビット単位のセット／リセット機能を用意しています。同様のマクロセルとしてKP65がありますが、KP65との違いは下記の点です。

#### 特徴

##### KP66

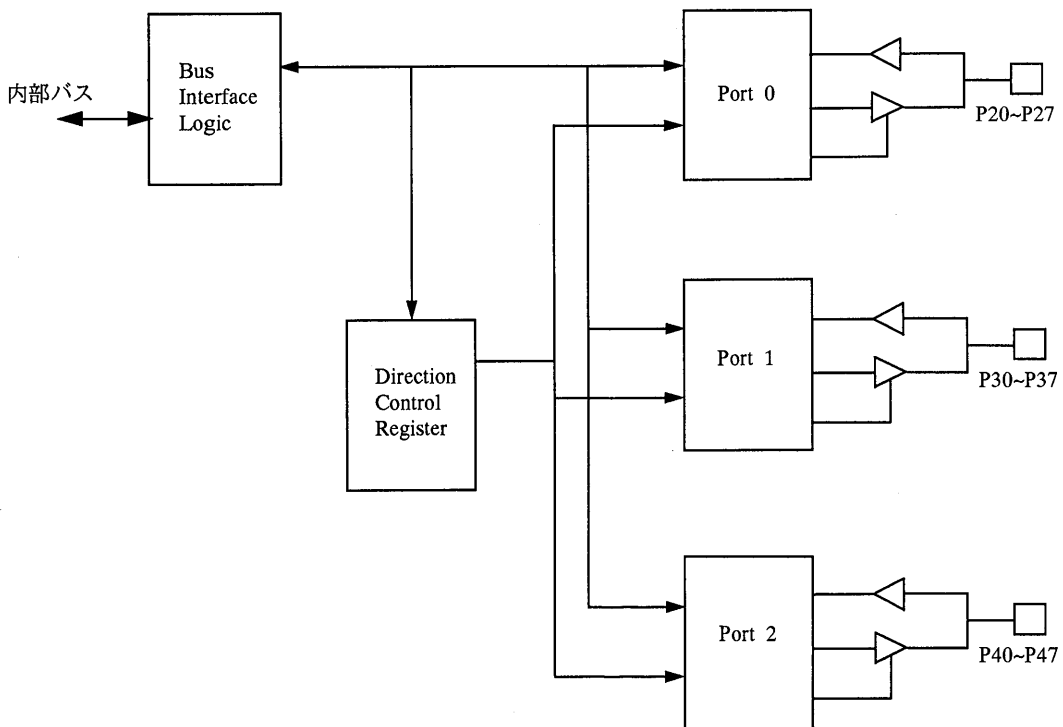
- ・24本の平行ポート搭載
- ・4ビット単位の入力／出力の方向制御が可能
- ・ソフトによりビット単位のセット／リセットが可能

##### KP65

- ・16本の平行ポート搭載
- ・1ビット単位の入力／出力の方向制御が可能

#### 11.1 ブロック図

KP66 の全体ブロック図と I/O レジスタマッピングを以下に示します。



### I/O レジスタマッピング

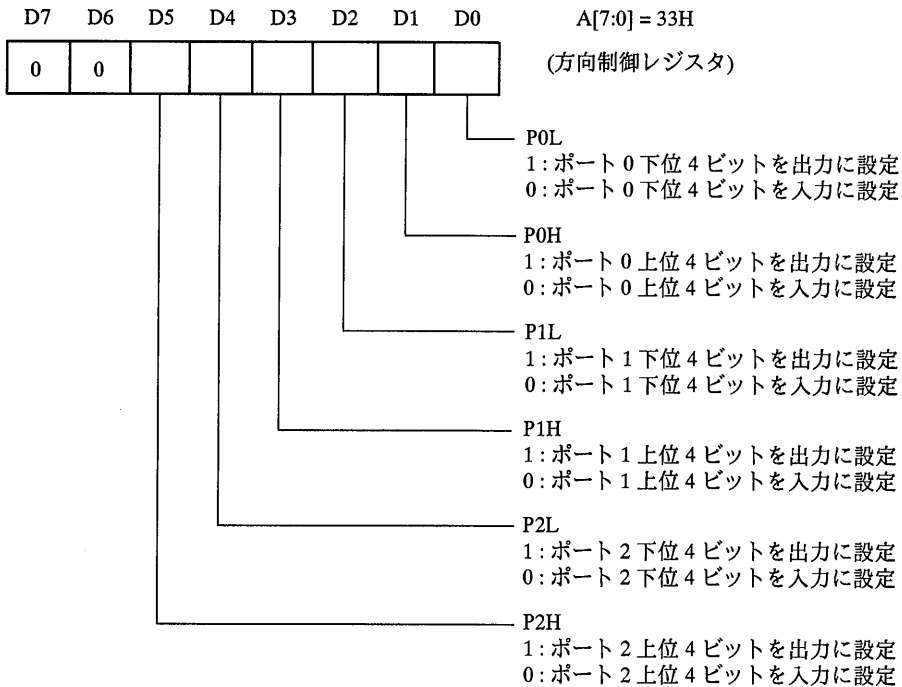
I/O アドレス	ブロック名	ライト時	リード時
30H	パラレルポートB	ポート0	ポート0
31H		ポート1	ポート1
32H		ポート2	ポート2
33H		コントロール・コマンド	方向制御レジスタ

### 11.2 コントロールコマンド

KP66 のポート入出力方向設定は I/O アドレス = 33H のコントロールコマンドで行います。コントロールコマンドはデータバス上位 2 ビットの値で動作が異なります。

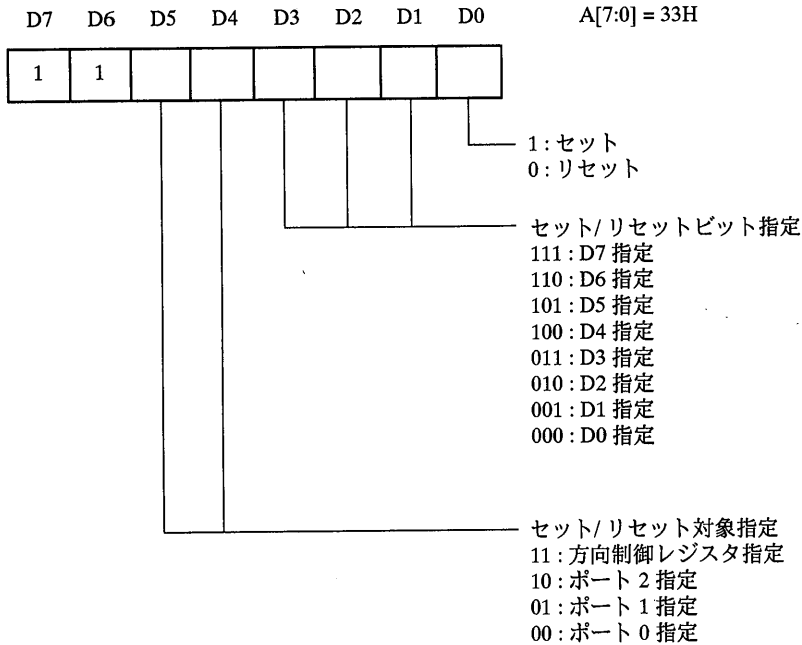
#### ポート方向設定コマンド (方向制御レジスタの設定)

KP66 は 24 本のパラレルポートの入出力の方向設定を方向制御レジスタにより 4 ビット単位で設定できます。本コマンドは方向制御レジスタの設定を直接行います。



### ビット操作コマンド

KP66 は全ポート 24 本と方向制御レジスタの 6 ビットのすべてを 1 ビット単位でセット/リセットできます。



### 11.3 ポート入出力動作

#### ポート入力動作

入力にモード設定されたポートでは双方向バッファからとりこんだデータをデータバスに出力します。

#### ポート出力動作

出力に設定されたポートではデータバスから各ポートに書き込まれたデータが双方向バッファから出力されます。

### 11.4 リセット

リセット時には以下のレジスタが初期化されます。

ポート 0	全ビット	リセット
ポート 1	全ビット	リセット
ポート 2	全ビット	リセット
方向制御レジスタ	全ビット	リセット

### 11.5 使用上の注意

他の信号とマルチプレクスされているパラレルポートをパラレルポートとして使用しない場合、そのパラレルポートは入力方向に設定して下さい。

### 12. 動作モード設定

本チップの動作モードは外部入力端子 MODE[1:0] とシステムコントロールレジスタにより指定されます。

#### 12.1 入力端子によるモード設定

MODE [1:0] = '00'	:	禁止
'01'	:	バグ・ファインダ Boot-on-RAM モード
'10'	:	マキシマムモード。ピン 68, 69 から A18, A17 を出力します。
'11'	:	ノーマルモード

#### ノーマルモード

ノーマルモードでは端子 68, 69 から M1CS<sub>1</sub>, M0CS<sub>1</sub> を出力します。この 2 つの信号は外部メモリのチップセレクト信号として使えます。このとき、外部に接続できるメモリは合計最大 256K バイトになります。

#### マキシマムモード

マキシマムモードでは端子 68, 69 から A[18], A[17] を出力します。このとき外部に最大 512K バイトのメモリが接続できます。

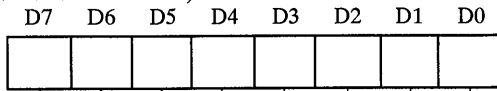
#### バグ・ファインダ Boot-on-RAM モード

バグ・ファインダ Boot-on-RAM モードはバグ・ファインダ・アダプタを BFSIO 端子に接続することでバグ・ファインダが外部 RAM 上で立ち上がります。端子 68, 69 から M1CS<sub>1</sub>, M0CS<sub>1</sub> を出力します。詳しくはバグ・ファインダのマニュアルを参照して下さい。本チップの動作はノーマルモードと同じです。外部に接続できるメモリは合計最大 256K バイトになります。

マキシマムモード、ノーマルモードに関係なく、内部アドレス空間は 1M バイト存在し、内部 RAM は物理アドレス空間 FFE00H~FFFFFH (512byte) にマップされています。

### 12.2 システムコントロールレジスタ

SCR0 (I/O アドレス =3AH)



シリアルポート TXC, RXC 入力  
 1: 端子 73, 74 からの入力を使います。  
 0: タイマ B チャンネル 1 の OUTBP 出力を内部的に接続します。端子 73, 74 は P14, P15 として機能します。

端子 90, 91, 94, 95 の機能  
 1: DSR\_, CTS\_, DTR\_, RTS\_ として機能します。  
 0: P41, P42, P45, P46 として機能します。DSR\_, CTS\_ には “L” が入力されます。

端子 89, 93 の機能  
 1: SYNDBD, SYDTIN として機能します。  
 0: P43, P47 として機能します。SYDTIN には “L” が入力されます。

端子 1, 4, 5, 6 の機能  
 1: OUTBS0, OUTBP1, OUTBP2, SYNC として機能します。  
 0: P30 ~ 33 として機能します。

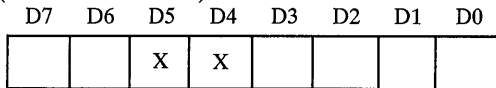
端子 97 ~ 100 の機能  
 1: OUTBS2, OUTBP0, OUTA1, OUTA0 として機能します。  
 0: P34 ~ 37 として機能します。

タイマ A チャンネル 0 GATE 入力 (GATEA0)  
 1: 端子 79 から入力されます。  
 0: 常に “H” が入力されます。端子 79 は P10 として機能します。

タイマ A チャンネル 1 GATE 入力 (GATEA1)  
 1: 端子 76 から入力されます。  
 0: 常に “H” が入力されます。端子 76 は P12 として機能します。

タイマ A の動作  
 1: タイマ A のチャンネル 1, 0 がカスケード接続され、32 ビットカウンタとして動作します。  
 0: タイマ A のチャンネル 1, 0 が独立に動作します。

SCR1 (I/O アドレス = 3BH)



端子 72 の機能

1: KC82 の M1\_ を出力します。  
0: P16 として機能します。

端子 71 の機能

1: KC82 の HALT\_ を出力します。  
0: P17 として機能します。

端子 92, 96 の機能

1: KC82 の BREQ\_, BACK\_ として機能します。  
0: P40, 44 として機能します。  
KC82 の BREQ\_ には常に "H" が入力されます。

端子 85 の機能

1: KC82 の NMI\_ 入力として機能します。  
0: P03 もしくは IR3 として機能します。  
KC82 の NMI\_ には常に "H" が入力されます。

外部バス・ウェイト・コントロール

	外部メモリ (00000 ~ 7FFFF)	外部メモリ (80000 ~ FFDFE)	外部 I/O
00	1 ウェイト	1 ウェイト	2 ウェイト
01	1 " *	1 " *	2 "
10	1 "	0 "	1 "
11	0 "	0 "	1 "

\*ワイド・ライト・ストロープ・オプション

注) 割り込みコントローラの外部割り込み要求入力 IR0 ~ IR7 を使用するときは対応するパラレル・ポートを入力方向に設定して下さい。IR0 ~ IR7 の使い方については割り込みコントローラの章を参照して下さい。

### 端子 75, 78 の使い方

端子 75, 78 の入出力方向の設定はシステムコントロールレジスタの設定には関係なく、パラレルポート A の方向制御レジスタによってのみ決定されます。

端子 75, 78 を XCLK0,1 として使用する場合は P11, P13 を入力方向に設定して下さい。

また端子 75, 78 をパラレルポートとして使用する場合は対応するタイマ/カウンタ A のチャネルはカウンタクロックはシステムクロックを使用するモードに設定して下さい。

### 端子 80, 81, 82, 83 の使い方

端子 80, 81, 82, 83 の入出力方向の設定はシステムコントロールレジスタの設定には関係なくパラレルポート A の方向制御レジスタによってのみ決定されます。したがって、端子 80, 81, 82, 83 からの外部入力を IR 4, 5, 6, 7 として使用する場合はパラレルポート A の P04, P05, P06, P07 は入力方向に設定して下さい。

逆にこれらの端子をパラレルポート A の P04, P05, P06, P07 として使用する場合は割り込みコントローラの設定により IR 4, 5, 6, 7 をマスクする必要があります。

### 端子 85 の使い方

端子 85 をシステムコントロールレジスタの設定により P03 もしくは IR3 として使用する場合 (SCR1 で D3= '0')、入出力方向の設定はパラレルポート A の方向制御レジスタによってのみ決定されます。したがって、端子 85 からの外部入力を IR 3 として使用する場合はパラレルポート A の P03 は入力方向に設定して下さい。

逆にこれらの端子をパラレルポート A の P03 として使用する場合は割り込みコントローラの設定により IR 3 をマスクする必要があります。

### 端子 86, 87, 88 の使い方

端子 86, 87, 88 の入出力方向の設定はシステムコントロールレジスタの設定には関係なくパラレルポート A の方向制御レジスタによってのみ決定されます。したがって、端子 86, 87, 88 からの外部入力を IR 0, 1, 2 あるいは GATEB 0, 1, 2 として使用する場合はパラレルポート A の P00, P01, P02

は入力方向に設定して下さい。

このとき、IR0, 1, 2 と GATEB 0, 1, 2 には共に外部入力が常に入っています。いずれかの機能を無効にしたいときは各ブロックのモード設定により無効にして下さい。例えば端子 88 を GAREB0 として使用する場合はパラレルポート A の設定により P00 を入力方向に設定し、割り込みコントローラの設定により IR0 をマスクする必要があります。

また端子 86, 87, 88 をパラレルポートとして使用する場合は対応する外部割り込み入力はマスク状態にし、対応するゲート入力 (GATEB 0, 1, 2) のカウンタチャネルは GATE 機能なしで使用されることをおすすめします。

### 12.3 リセット時

リセット時には SCR0, SCR1 ともに '00H' にリセットされます。

### 13. アドレス・マッピング

#### 内部I/Oマッピング

表 13-1 内部I/Oマッピング

I/Oアドレス	ブロック名	ライト時	リード時
00H	KC82(MMU)	BBR1 (境界/ベースレジスタ1)	BBR1 (境界/ベースレジスタ1)
01H		BR1 (ベースレジスタ1)	BR1 (ベースレジスタ1)
02H		BBR2 (境界/ベースレジスタ2)	BBR2 (境界/ベースレジスタ2)
03H		BR2 (ベースレジスタ2)	BR2 (ベースレジスタ2)
04H		BBR3 (境界/ベースレジスタ3)	BBR3 (境界/ベースレジスタ3)
05H		BR3 (ベースレジスタ3)	BR3 (ベースレジスタ3)
06H		BBR4 (境界/ベースレジスタ4)	BBR4 (境界/ベースレジスタ4)
07H		BR4 (ベースレジスタ4)	BR4 (ベースレジスタ4)
08H-1FH	川崎製鉄使用予約		
20H	タイマ/カウンタB	チャンネル0カウンタ	チャンネル0カウンタ
21H		チャンネル0コントロールワード	チャンネル0ステータス
22H		チャンネル1カウンタ	チャンネル1カウンタ
23H		チャンネル1コントロールワード	チャンネル1ステータス
24H		チャンネル2カウンタ	チャンネル2カウンタ
25H		チャンネル2コントロールワード	チャンネル2ステータス
26H		川崎製鉄使用予約	川崎製鉄使用予約
27H		川崎製鉄使用予約	川崎製鉄使用予約
28H	タイマ/カウンタA	チャンネル0カウンタ	チャンネル0カウンタ
29H		チャンネル0コントロールワード	チャンネル0ステータス
2AH		チャンネル1カウンタ	チャンネル1カウンタ
2BH		チャンネル1コントロールワード	チャンネル1ステータス
2CH	パラレルポートA	ポート0	ポート0
2DH		ポート0方向制御レジスタ	ポート0方向制御レジスタ
2EH		ポート1	ポート1
2FH		ポート1方向制御レジスタ	ポート1方向制御レジスタ
30H	パラレルポートB	ポート0	ポート0
31H		ポート1	ポート1
32H		ポート2	ポート2
33H		コントロール・コマンド	方向制御レジスタ
34H	割り込みコントローラ	LERL/PGRL	ISRL
35H		LERH/PGRH	ISRH
36H		IMRL	IMRL
37H		IVR/IMRH	IMRH
38H	シリアルポート	送信データ	受信データ
39H		コマンド/モード設定	ステータス
3AH	システム制御レジスタ	SCR0	SCR0
3BH		SCR1	SCR1
3CH-3FH	川崎製鉄使用予約		

注) 内部 I/O アドレスのデコードは、8 ビットデコードでアドレス上位 8 ビットは無視してデコードしています。

### メモリ・マッピング

#### ノーマル・モード時

内蔵RAM領域	物理アドレス空間	FFE00H~FFFFFH (512byte)
外部メモリ領域1	物理アドレス空間(M1CS <sub>1</sub> )	E0000H~FFDFH (128K-512byte)
外部メモリ領域0	物理アドレス空間(M0CS <sub>1</sub> )	00000H~1FFFFH (128Kbyte)

#### マキシマム・モード時

内蔵RAM領域	物理アドレス空間	FFE00H~FFFFFH (512byte)
外部メモリ領域	物理アドレス空間	00000H~7FFFFH (512Kbyte)

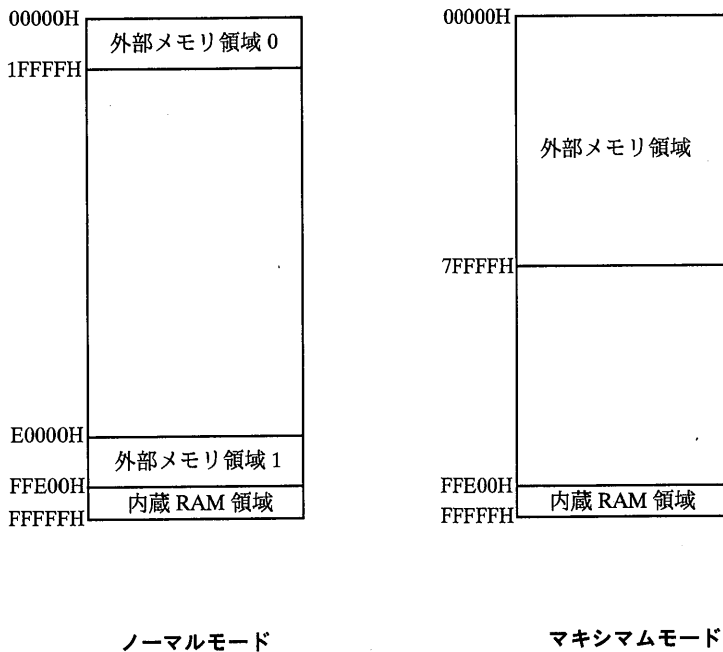


図 13-1 メモリマッピング

### 14. 発振回路

#### 14.1 概要

KL5C8012 はシステムクロックを発生させるための発振バッファを搭載しています。チップ内部のシステムクロックはこの発振回路が発生させた信号を2分周した信号です。

#### 14.2 回路構成

システムクロックを発生させるためには KL5C8012 の発振バッファ XIN, XOUT に外部部品として水晶振動子 (あるいはセラミック振動子)、フィードバック抵抗、制限抵抗、コンデンサを右図のように接続することで発振回路を構成できます。外部部品定数は使用する振動子、基板等によって異なります。外部部品定数の最適値は振動子メーカーの推奨値を御使用下さい。チップ内部のシステムクロックはこの発振回路が発生させた信号を2分周した信号です。分周回路はチップ内に搭載されています。

表 14-1 発振周波数

発振周波数	動作電源電圧
2~20 MHz	5V ± 5%, 5V ± 10%

#### 注意

外部ヘクロックを取り出す場合は CLK 端子から取り出して下さい。XIN, XOUT から直接信号を取り出さないでください。

外部からクロックを入力する場合は XIN からシステムクロックの2倍の周波数の信号を入力して下さい。このとき XOUT につく寄生容量はできるだけ小さくして下さい。

表 14-2 外部部品定数参考範囲

	Rd	Cl, Co
水晶	100 ~ 800 Ω	5 ~ 30pF
セラミック	30 ~ 300 Ω	5 ~ 100pF

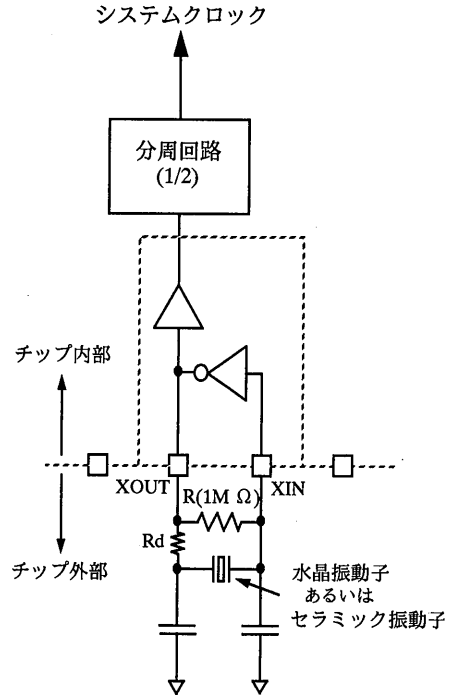


図 14-1 発振回路の構成

**15. その他**

本 LSI のソフト開発ツールとしては ICE と単型デバッグツール (バグ・ファインダ) 等多数ございます。それらにつきましては、別紙の開発ツール一覧表をご参照下さい。

### 16. 電気的特性

#### 16.1 絶対最大定格

表 16-1 絶対最大定格 (GND 基準)

項目	記号	定格値	単位
電源電圧	VDD	-0.6 ~ +7.0	V
入力電圧	VIN	-0.6 ~ VDD +0.6	V
保存温度	TSTG	-40 ~ +125	°C

#### 16.2 DC 特性 (5V ± 10%)

表 16-2 推奨動作条件

項目	記号	定格値	単位
電源電圧	VDD	4.5 ~ 5.5	V
動作周囲温度	TA	0 ~ +70	°C

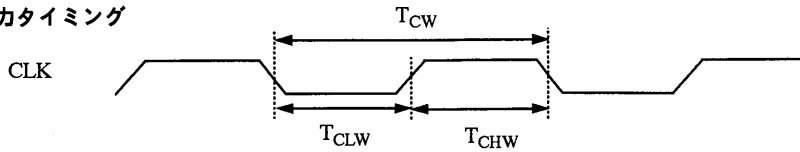
表 16-3 電気的特性 (推奨動作条件での特性)

項目	記号	規格値			単位	測定条件
		最小	標準	最大		
入力電圧 (RESET_ 以外の入力端子)	V <sub>IH</sub>	3.6		VDD	V	
	V <sub>IL</sub>	GND		1.4	V	
RESET_ 入力端子 (シュミットトリガ入力)	V <sub>+</sub>	2.4		4.0	V	
	V <sub>-</sub>	0.9		2.3	V	
	V <sub>h</sub>	0.9			V	
出力電圧	V <sub>OH</sub>	3.5			V	I <sub>OH</sub> = -4mA
	V <sub>OL</sub>			0.4	V	I <sub>OL</sub> = 4mA
出力電流	I <sub>OUT</sub>			±4	mA	
入力リーク電流	I <sub>IL</sub>	-10			μA	V <sub>IN</sub> = GND
	I <sub>IH</sub>			10	μA	V <sub>IN</sub> = VDD
出力リーク電流	I <sub>OZ</sub>	-10		10	μA	ハイ・インピーダンス出力時
プル・アップ電流	I <sub>PU</sub>	20	95	250	μA	V <sub>IN</sub> = GND
スタンバイ電流	I <sub>DDS</sub>		1.0*	100	μA	CLK 停止時、全端子 = H
動作時消費電流	I <sub>DDOP</sub>		25*		mA	f (CLK) = 10MHz

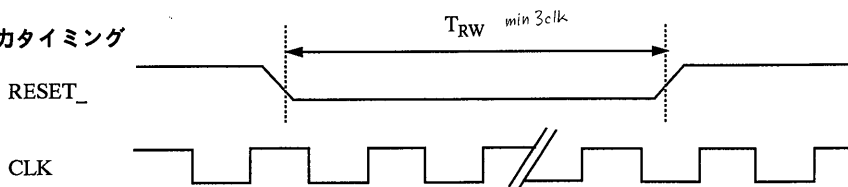
\* TA = 25 °C のとき

### 16.3 AC 特性

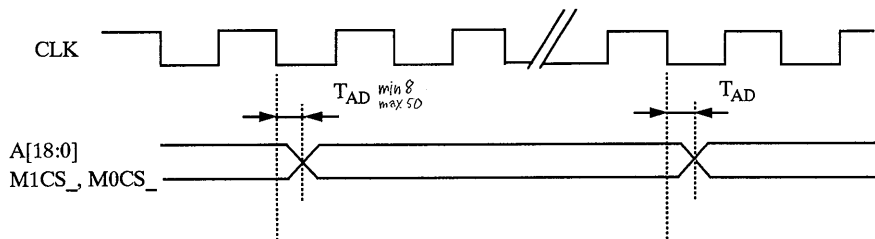
#### クロック出カタイミング



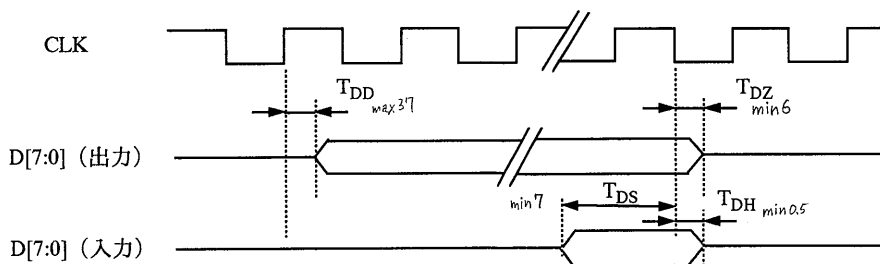
#### リセット入カタイミング



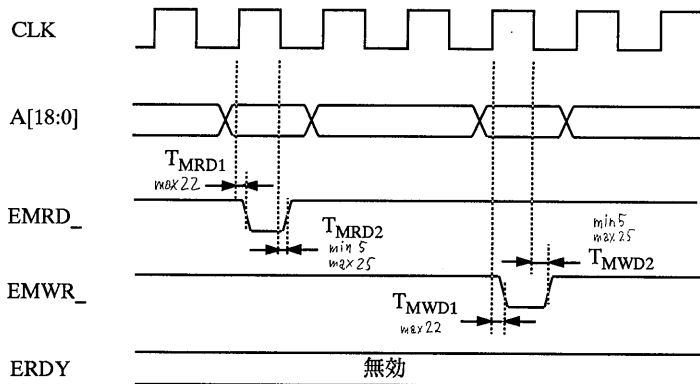
#### アドレス出カタイミング



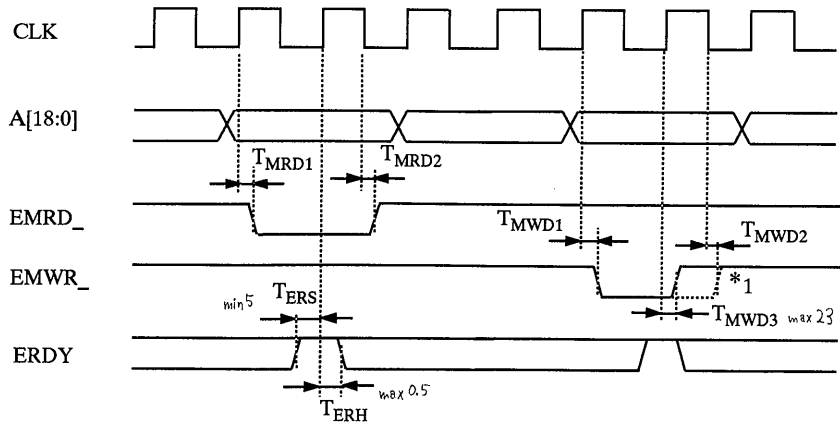
#### データ入出カタイミング



### 外部メモリアクセス (0 ウェイト)

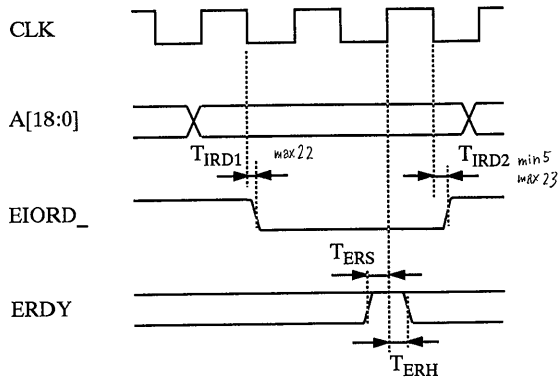


### 外部メモリアクセス (1 ウェイト)

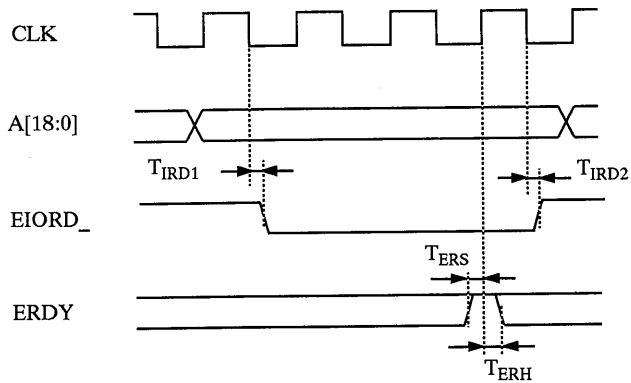


\*1 ワイド・ライト・ストロブ・オプション時の EMWR\_ 信号のタイミング

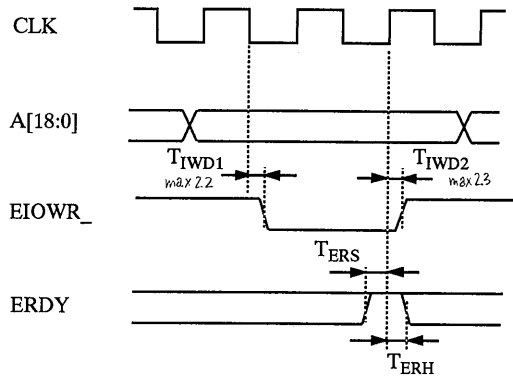
### 外部 I/O リード (1 ウェイト)



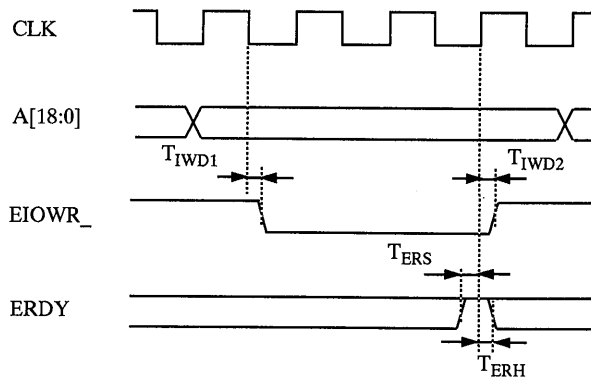
### 外部 I/O リード (2 ウェイト)



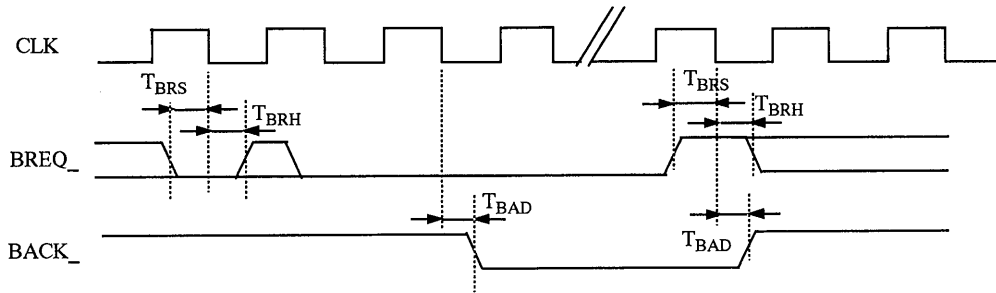
### 外部 I/O ライト (1 ウェイト)



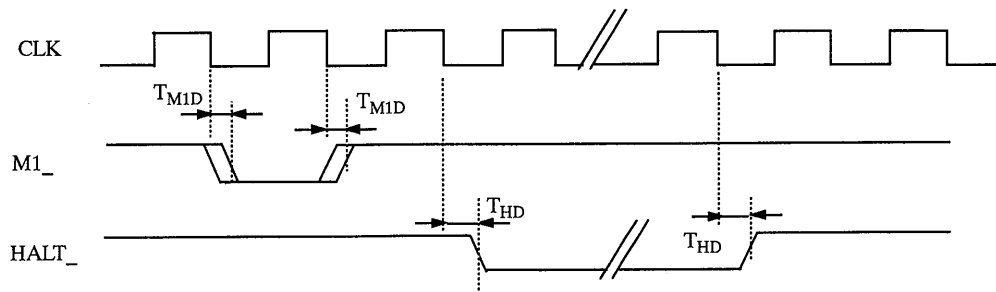
### 外部 I/O ライト (2 ウェイト)



### バスリクエスト入カタイミング

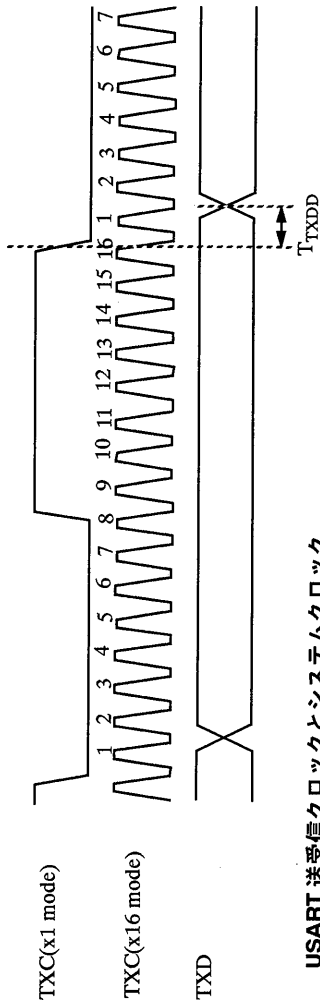


### CPU 制御出カタイミング

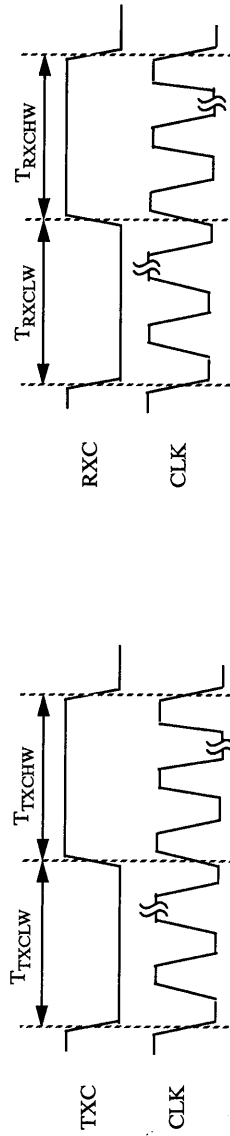


M1\_ 信号を利用する時は CLK の立ち上がりエッジでサンプルしてから使用して下さい。

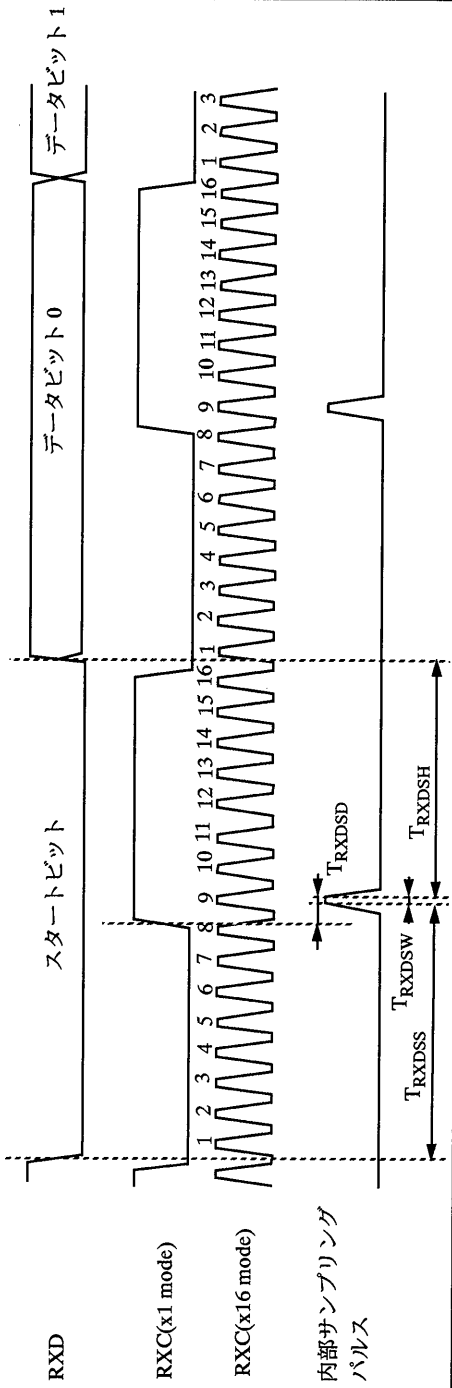
### USART 送信クロックと送信データ



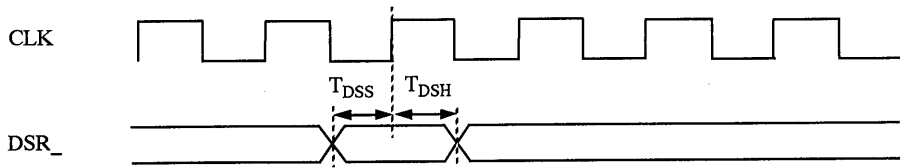
### USART 送受信クロックとシステムクロック



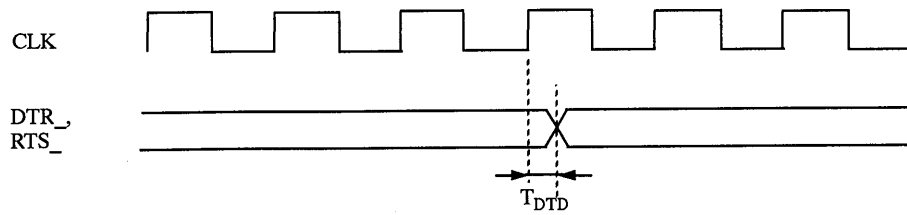
### USART 受信クロックと受信データ



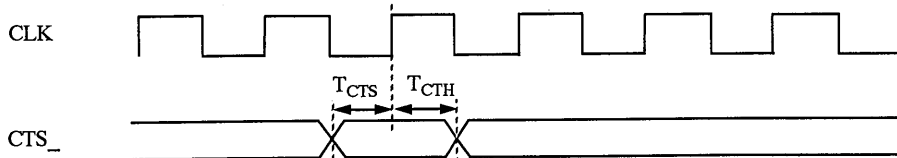
### DSR\_ 入カタイミング



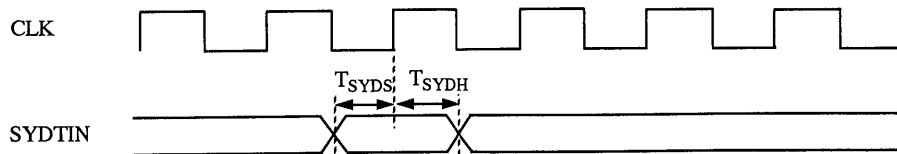
### DTR\_, RTS\_ 出カタイミング



### CTS\_ 入カタイミング

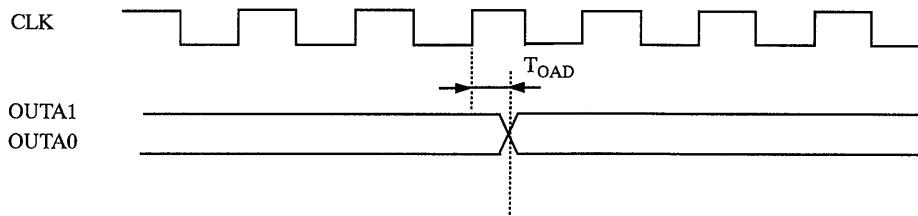


### SYDTIN 入カタイミング

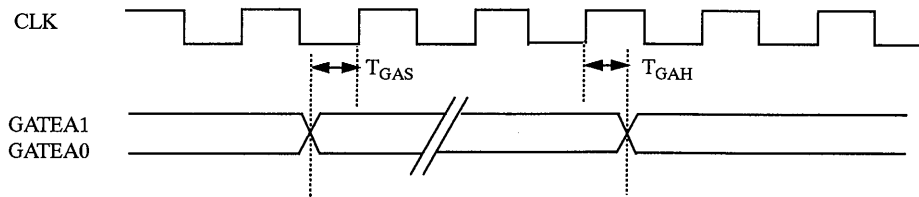


### タイマ/カウンタ A

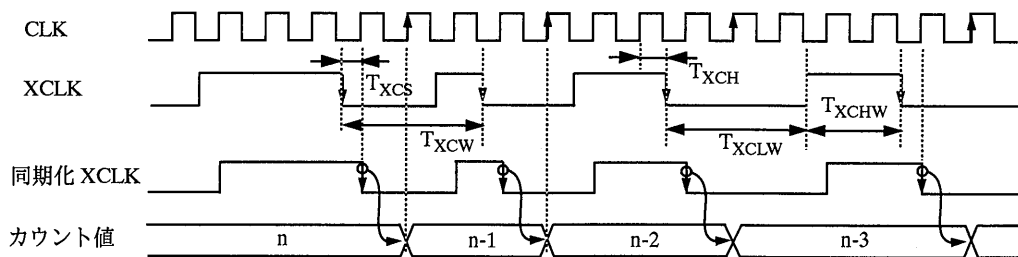
#### カウンタ出カタイミング



#### カウンタゲートタイミング



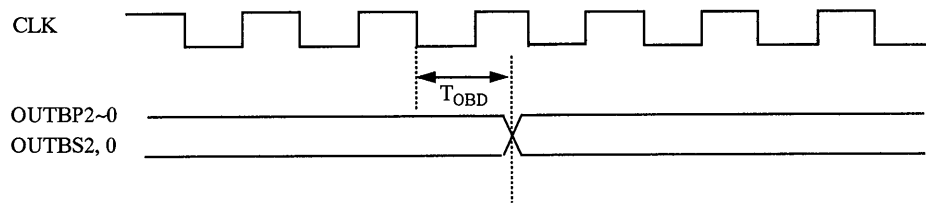
#### 外部カウンタクロック入力タイミング



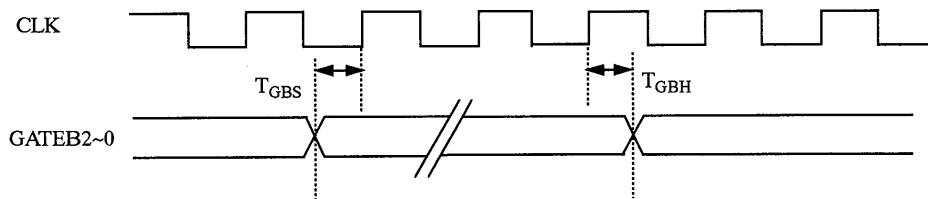
※ 外部クロックは、システムクロックの立ち上がりエッジでサンプルしてから使用します。

### タイマ/カウンタ B

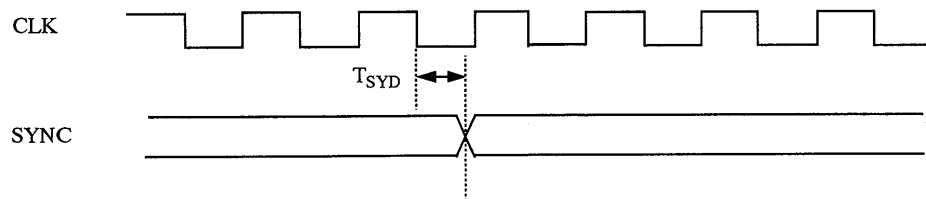
#### カウンタ出カタイミング



#### カウンタゲートタイミング

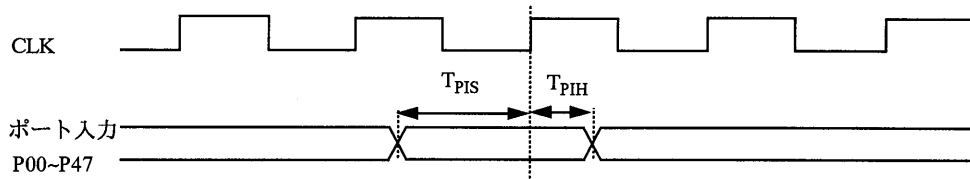


#### SYNC 出カタイミング

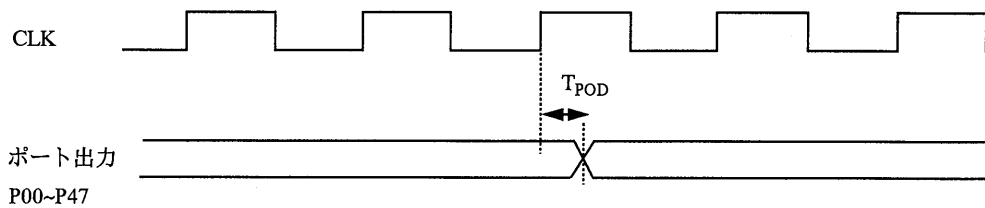


### パラレルポート A/B

#### ポート入力タイミング

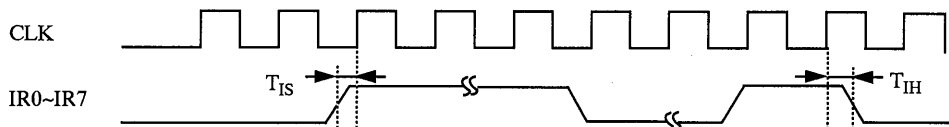


#### ポート出力タイミング



### 割り込みコントローラ

#### 外部割り込みタイミング



### KL5C80A12 AC 特性 (5V ± 10%)

番号	項目	最小	標準	最大	単位
T <sub>CYC</sub>	XINサイクル時間	50.0			ns
T <sub>CW</sub>	CLKサイクル時間	100.0			ns
T <sub>CLW</sub>	CLK “L” パルス幅		50.0		ns
T <sub>CHW</sub>	CLK “H” パルス幅		50.0		ns
T <sub>RW</sub>	RESET_パルス幅	3			clk
T <sub>AD</sub>	アドレス遅延時間	8.0		50.0	ns
T <sub>DD</sub>	CLK→データ出力遅延時間			37.0	ns
T <sub>DZ</sub>	CLK→データ出力off遅延時間	6.0			ns
T <sub>DS</sub>	データ入力セットアップ時間	7.0			ns
T <sub>DH</sub>	データ入力ホールド時間	0.5			ns
T <sub>MRD1</sub>	CLK立ち上がりエッジ→EMRD_ “L” 遅延時間			22.0	ns
T <sub>MRD2</sub>	CLK立ち下がりエッジ→EMRD_ “H” 遅延時間	5.0		25.0	ns
T <sub>MWD1</sub>	CLK立ち上がりエッジ→EMWR_ “L” 遅延時間			22.0	ns
T <sub>MWD2</sub>	CLK立ち下がりエッジ→EMWR_ “H” 遅延時間	5.0		25.0	ns
T <sub>MWD3</sub>	CLK立ち上がりエッジ→EMWR_ “H” 遅延時間			23.0	ns
T <sub>ERS</sub>	ERDYセットアップ時間	5.0			ns
T <sub>ERH</sub>	ERDYホールド時間	0.5			ns
T <sub>IRD1</sub>	CLK立ち下がりエッジ→EIORD_ “L” 遅延時間			22.0	ns
T <sub>IRD2</sub>	CLK立ち下がりエッジ→EIORD_ “H” 遅延時間	5.0		23.0	ns
T <sub>IWD1</sub>	CLK立ち下がりエッジ→EIOWR_ “L” 遅延時間			22.0	ns
T <sub>IWD2</sub>	CLK立ち上がりエッジ→EIOWR_ “H” 遅延時間			23.0	ns
T <sub>BRS</sub>	BREQ_セットアップ時間	5.0			ns
T <sub>BRH</sub>	BREQ_ホールド時間	1.6			ns
T <sub>BAD</sub>	BACK_遅延時間			43.0	ns
T <sub>M1D</sub>	M1_遅延時間			44.0	ns
T <sub>HD</sub>	HALT_遅延時間			45.0	ns
T <sub>TXDD</sub>	TXD遅延時間(TXC外部入力時)			21.0	ns
T <sub>TXCLW</sub>	TXC “L” パルス幅	2.5			clk
T <sub>TXCHW</sub>	TXC “H” パルス幅	2.5			clk

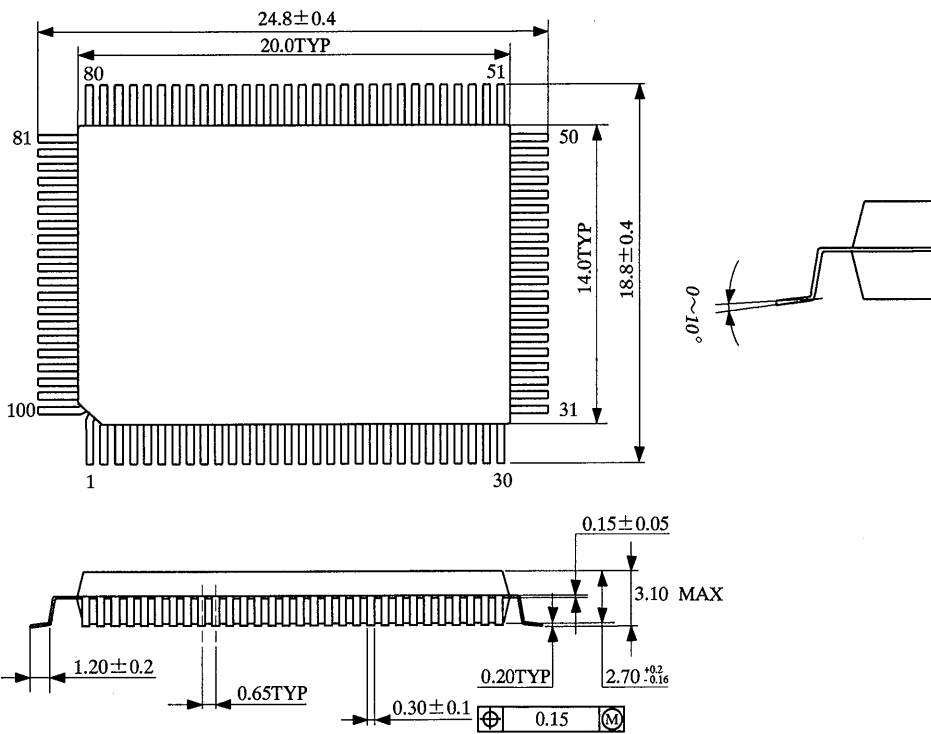
番号	項目	最小	標準	最大	単位
T <sub>RXCLW</sub>	RXC “L” パルス幅	2.5			clk
T <sub>RXCHW</sub>	RXC “H” パルス幅	2.5			clk
T <sub>RXDSD</sub>	RXD内部サンプリングパルス遅延時間	2		3	clk
T <sub>RXDSDW</sub>	RXD内部サンプリングパルス幅		1		clk
T <sub>RXDSS</sub>	RXDセットアップ時間		3		clk
T <sub>RXDSSH</sub>	RXDホールド時間		0		clk
T <sub>DSS</sub>	DSR_セットアップ時間	5.0			ns
T <sub>DSSH</sub>	DSR_ホールド時間	2.8			ns
T <sub>DTD</sub>	DTR_ RTS_遅延時間			25.0	ns
T <sub>CTS</sub>	CTS_セットアップ時間	5.0			ns
T <sub>CTH</sub>	CTS_ホールド時間	2.6			ns
T <sub>SYDS</sub>	SYDTINセットアップ時間	5.0			ns
T <sub>SYDH</sub>	SYDTINホールド時間	2.9			ns
T <sub>OAD</sub>	タイマ/カウンタA出力遅延時間			22.0	ns
T <sub>GAS</sub>	GATEA入力セットアップ時間	5.0			ns
T <sub>GAH</sub>	GATEA入力ホールド時間	0.5			ns
T <sub>XCS</sub>	XCLK入力セットアップ時間	5.0			ns
T <sub>XCH</sub>	XCLK入力ホールド時間	3.0			ns
T <sub>XCW</sub>	XCLK周期	2			clk
T <sub>XCLW</sub>	XCLK “L” パルス幅	1			clk
T <sub>XCHW</sub>	XCLK “H” パルス幅	1			clk
T <sub>OBD</sub>	タイマ/カウンタB出力遅延時間			42.0	ns
T <sub>GBS</sub>	GATEB入力セットアップ時間	5.0			ns
T <sub>GBH</sub>	GATEB入力ホールド時間	3.7			ns
T <sub>SYD</sub>	SYNC出力遅延時間			19.0	ns
T <sub>PIS</sub>	ポート入力セットアップ時間	5.0			ns
T <sub>PIH</sub>	ポート入力ホールド時間	0.5			ns
T <sub>POD</sub>	ポート出力遅延時間			30.0	ns
T <sub>IS</sub>	外部割り込み入力セットアップ時間	5.0			ns
T <sub>IH</sub>	外部割り込み入力ホールド時間	2.3			ns

注1) 出力端子の負荷容量 $C_L$ は70pFで測定しています。

注2) 単位の欄に clk とあるのはシステムクロック数を示しています。

### 17. 外形寸法図

KL5C8012 はプラスチック QFP100 パッケージに封止されています。下に QFP100 の外形寸法図を掲載します。



## 付録 A インストラクション表

この表は、命令の種類ごとに分類して、KC82 CPU コアのインストラクションをまとめたものです。表中の“ニーモニック”は各命令の名称を現わし、“命令動作”でその動作内容を簡潔に示しています。

命令動作の欄で“<-”とあるのは、右側の内容を左側に代入することを、カッコでくくられたものは、くくられたレジスタなどで示されるメモリの内容を、それぞれ意味しています。たとえば

r<-[HL]

とあるのは、HLレジスタで示されるアドレスのメモリの内容を、8ビットレジスタに代入するということです。ただし入出力命令の[n]と[C]は、対応する入出力ポートのデータを意味しています。

“フラグ”の欄は各フラグの動作を、“オペコード”はそれぞれの命令に対するマシン語コードを、2進数と16進数で記したものです。その右側の“B”“CL”は、各命令の長さ（バイト数）と、内部バスサイクルで命令を実行するのに要するクロック数を、それぞれ表わしています。ある命令の実行中に外部バスサイクルが存在するときには、外部バス・インターフェース・ユニットに設定されているウェイト数だけ“CL”に加算したものが実際にその命令を実行するのに要するクロック数になります。

このほか、インストラクション表に出てくる略号に関して、次の凡例にまとめておいたので参考にしてください。

A{7}	レジスタ A の最上位ビット
A{4..7}	レジスタ A のビット 4-7
;	動作の区切り
[IX+d]	IX に 8 ビットの符号つき変位を足した値が示すアドレスのデータ
C	キャリーフラグ
Z	ゼロフラグ
PV	パリティ・オーバーフローフラグ
S	サインフラグ
N	減算フラグ
H	ハーフキャリーフラグ
●	フラグは変化しない
⇔	フラグは実行結果により変化する
0	フラグは 0
1	フラグは 1
?	不定になる
V	オーバーフローフラグとして使われる
P	パリティフラグとして使われる
IF	割り込みフリップフロップの値が入る
r, r'	8ビットレジスタ、A, B, C, D, E, H, L
ss	16ビットレジスタ、BC, DE, HL, SP
pp	16ビットレジスタ、BC, DE, IX, SP
rr	16ビットレジスタ、BC, DE, IY, SP

qq	16 ビットレジスタ、BC, DE, HL, AF
e	JR 系の命令の飛び先アドレスへの差分、8 ビットの符号つき即値 (+127 ~ -128)
k	RST 命令の飛び先アドレス、00h, 08h, 10h, 18h, 20h, 28h, 30h, 38h
nn	16 ビットの即値、もしくは絶対アドレス
n	8 ビットの即値
b	ビット演算命令の第何ビットかを示す値
NOT	ビットを反転する
∨	ビットの OR をとる
∇	ビットの XOR をとる
∧	ビットの AND をとる
tmp	一時的に値を待避する
B	命令のバイト数
CL	命令の実行に必要な最小クロック数

分岐命令、コール命令でクロック数がふたつ書いてあるものは、上が条件が成立しないとき、下が条件が成立したときを意味します。

また、入出力命令でクロック数がふたつ書いてあるものは、上がまだ転送が終わらないとき、下が転送が終わったときをそれぞれ意味しています。

## 8 BIT LOAD GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
LD r, r'	r<-r'	● ● ● ● ● ● ● ●	01 r r'		1	1
LD r, n	r<-n	● ● ● ● ● ● ● ●	00 r 110		2	2
			<- n ->			
LD r, [HL]	r<-[HL]	● ● ● ● ● ● ● ●	01 r 110		1	2
LD r, [IX+d]	r<-[IX+d]	● ● ● ● ● ● ● ●	11011101	DD	3	5
			01 r 110			
			<- d ->			
LD r, [IY+d]	r<-[IY+d]	● ● ● ● ● ● ● ●	11111101	FD	3	5
			01 r 110			
			<- d ->			
LD [HL], r	[HL]<-r	● ● ● ● ● ● ● ●	01110 r		1	2
LD [IX+d], r	[IX+d]<-r	● ● ● ● ● ● ● ●	11011101	DD	3	5
			01110 r			
			<- d ->			
LD [IY+d], r	[IY+d]<-r	● ● ● ● ● ● ● ●	11111101	FD	3	5
			01110 r			
			<- d ->			
LD [HL], n	[HL]<-n	● ● ● ● ● ● ● ●	00110110	36	2	3
			<- n ->			
LD [IX+d], n	[IX+d]<-n	● ● ● ● ● ● ● ●	1011101	DD	4	5

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
LD [IY+d], n	[IY+d]<-n	● ● ● ● ● ●	00110110	36	4	5
			<- d ->			
			<- n ->			
			1111101	FD		
LD A, I	A<-I	⇔ ⇔ 0 IF 0 ● *1	00110110	36	2	2
			<- d ->			
LD A, R	A<-R	⇔ ⇔ 0 IF 0 ● *1	11101101	ED	2	2
			<- n ->			
LD I, A	I<-A	● ● ● ● ● ●	01010111	57	2	2
			11101101	ED		
LD R, A	R<-A	● ● ● ● ● ●	01000111	47	2	2
			11101101	ED		
LD A, [BC]	A<-[BC]	● ● ● ● ● ●	01001111	4F	1	3
			00001010	0A		
			00011010	1A		
LD A, [DE]	A<-[DE]	● ● ● ● ● ●	00111010	3A	1	3
LD A, [mn]	A<-[mn]	● ● ● ● ● ●	<- nnl ->		3	4
LD [BC], A	[BC]<-A	● ● ● ● ● ●	<- nnh ->		1	3
			00000010	02		
			00010010	12		
			00110010	32		
LD [DE], A	[DE]<-A	● ● ● ● ● ●	<- nnl ->		1	3
LD [mn], A	[mn]<-A	● ● ● ● ● ●	<- nnh ->		3	4

\*1 IF indicates IFF2.

	000	001	010	011	100	101	110	111
r	B	C	D	E	H	L		A

### 16 BIT LOAD GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
LD ss, nn	ss<-nn	● ● ● ● ● ●	00ss0001		3	3
LD IX, nn	IX<-nn	● ● ● ● ● ●	<- nnl ->		4	4
			<- nnh ->			
			11011101	DD		
			00100001	21		
LD IY, nn	IY<-nn	● ● ● ● ● ●	<- nnl ->		4	4
			<- nnh ->			
			11111101	FD		

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
			00100001	21		
			< -nnl ->			
			< -nnh ->			
LD SP, HL	SP<-HL	● ● ● ● ● ●	11111001	F9	1	1
LD SP, IX	SP<-IX	● ● ● ● ● ●	11011101	DD	2	2
			11111001	F9		
LD SP, IY	SP<-IY	● ● ● ● ● ●	11111101	FD	2	2
			11111001	F9		
LD ss, [nn]	ssh<-[nn+1] ssl<-[nn]	● ● ● ● ● ●	11101101	ED	4	6
			01ss1011			
			< -nnl ->			
			< -nnh ->			
LD HL, [nn]	H<-[nn+1] L<-[nn]	● ● ● ● ● ●	00101010	2A	3	5
			< -nnl ->			
			< -nnh ->			
LD IX, [nn]	IXH<-[nn+1] IXL<-[nn]	● ● ● ● ● ●	11011101	DD	4	6
			00101010	2A		
			< -nnl ->			
			< -nnh ->			
LD IY, [nn]	IYH<-[nn+1] IYL<-[nn]	● ● ● ● ● ●	11111101	FD	4	6
			00101010	2A		
			< -nnl ->			
			< -nnh ->			
LD [nn], ss	[nn+1]<-ssh [nn]<-ssl	● ● ● ● ● ●	11101101	ED	4	6
			01ss0011			
			< -nnl ->			
			< -nnh ->			
LD [nn], HL	[nn+1]<-H [nn]<-L	● ● ● ● ● ●	00100010	22	3	5
			< -nnl ->			
			< -nnh ->			
LD [nn], IX	[nn+1]<-IXH [nn]<-IXL	● ● ● ● ● ●	11011101	DD	4	6
			00100010	22		
			< -nnl ->			
			< -nnh ->			
LD [nn], IY	[nn+1]<-IYH [nn]<-IYL	● ● ● ● ● ●	11111101	FD	4	6
			00100010	22		
			< -nnl ->			
			< -nnh ->			

	00	01	10	11
ss	BC	DE	HL	SP

### EXCHANGE GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
EX DE, HL	DE<->HL	● ● ● ● ● ●	11101011	EB	1	1
EX AF, AF'	AF<->AF'	⇔ ⇔ ⇔ ⇔ ⇔ ⇔	00001000	08	1	1
EX [SP], HL	L<->[SP]	● ● ● ● ● ●	11100011	E3	1	5
	H<->[SP+1]					
EX [SP], IX	IXL<->[SP]	● ● ● ● ● ●	11011101	DD	2	6
	IXH<->[SP+1]		11100011	E3		
EX [SP], IY	IYL<->[SP]	● ● ● ● ● ●	11111101	FD	2	6
	IYH<->[SP+1]		11100011	E3		
EXX	BC<->BC'	● ● ● ● ● ●	11011001	D9	1	1
	DE<->DE'					
	HL<->HL'					

### PUSH / POP GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
PUSH qq	[SP-2]<-qql	● ● ● ● ● ●	11qq0101		1	4
	[SP-1]<-qqh					
	SP<-SP-2					
PUSH IX	[SP-2]<-IXL	● ● ● ● ● ●	11011101	DD	2	5
	[SP-1]<-IXH		11100101	E5		
	SP<-SP-2					
PUSH IY	[SP-2]<-IYL	● ● ● ● ● ●	11111101	FD	2	5
	[SP-1]<-IYH		11100101	E5		
	SP<-SP-2					
POP qq	qql<-[SP]	● ● ● ● ● ●	11qq0001		1	3
	qqh<-[SP+1]					
	SP<-SP+2					
POP IX	IXL<-[SP]	● ● ● ● ● ●	11011101	DD	2	4
	IXH<-[SP+1]		11100001	E1		
	SP<-SP+2					
POP IY	IYL<-[SP]	● ● ● ● ● ●	11111101	FD	2	4
	IYH<-[SP+1]		11100001	E1		
	SP<-SP+2					

	00	01	10	11
qq	BC	DE	HL	AF

All flags change on POP AF.

## BLOCK TRANSFER GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
LDI	[DE]<-[HL]	● ● 0 ⇔ 0 ●	11101101	ED	2	5
	DE<-DE+1	*1	10100000	A0		
	HL<-HL+1					
	BC<-BC-1					
LDD	[DE]<-[HL]	● ● 0 ⇔ 0 ●	11101101	ED	2	5
	DE<-DE-1	*1	10101000	A8		
	HL<-HL-1					
	BC<-BC-1					
LDIR	repeat;	● ● 0 0 0 ●	11101101	ED	2	6
	[DE]<-[HL]		10110000	B0		
	DE<-DE+1					
	HL<-HL+1					
LDDR	BC<-BC-1					
	until BC=0					
	repeat;	● ● 0 0 0 ●	11101101	ED	2	6
	[DE]<-[HL]		10111000	B8		
DE<-DE-1						
HL<-HL-1						
	BC<-BC-1					
	until BC=0					

\*1 If BC=0 PV=0, else PV=1

## BLOCK SEARCH GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
CPI	A-[HL];HL<-	⇔ ⇔ ⇔ ⇔ 1 ●	11101101	ED	2	4
	HL+1;BC<-BC-1	*2 *1	10100001	A1		
CPD	A-[HL];HL<-	⇔ ⇔ ⇔ ⇔ 1 ●	11101101	ED	2	4
	HL-1;BC<-BC-1	*2 *1	10101001	A9		
CPIR	repeat;A-[HL]	⇔ ⇔ ⇔ ⇔ 1 ●	11101101	ED	2	6
	HL<-HL+1	*2 *1	10110001	B1		
	BC<-BC-1					
	until BC=0 or A=[HL]					
CPDR	repeat;A-[HL]	⇔ ⇔ ⇔ ⇔ 1 ●	11101101	ED	2	6
	HL<-HL-1	*2 *1	10111001	B9		
	BC<-BC-1					
	until BC=0 or A=[HL]					

\*1 If BC=0 PV=0, else PV=1

\*2 If A=[HL] Z=1, else Z=0

ADD GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
ADD A, r	A<-A+r	⇔⇔⇔V 0⇔	1000 r		1	1
ADD A, n	A<-A+n	⇔⇔⇔V 0⇔	11000110 <- n ->	C6	2	2
ADD A, [HL]	A<-A+[HL]	⇔⇔⇔V 0⇔	10000110	86	1	2
ADD A, [IX+d]	A<-A+[IX+d]	⇔⇔⇔V 0⇔	11011101 10000110 <- d ->	DD 86	3	5
ADD A, [IY+d]	A<-A+[IY+d]	⇔⇔⇔V 0⇔	11111101 10000110 <- d ->	FD 86	3	5
ADC A, r	A<-A+r+C	⇔⇔⇔V 0⇔	10001 r		1	1
ADC A, n	A<-A+n+C	⇔⇔⇔V 0⇔	11001110 <- n ->	CE	2	2
ADC A, [HL]	A<-A+[HL]+C	⇔⇔⇔V 0⇔	10001110	8E	1	2
ADC A, [IX+d]	A<-A+[IX+d]+C	⇔⇔⇔V 0⇔	11011101 10001110 <- d ->	DD 8E	3	5
ADC A, [IY+d]	A<-A+[IY+d]+C	⇔⇔⇔V 0⇔	11111101 10001110 <- d ->	FD 8E	3	5
ADC HL, ss	HL<-HL+ss+C	⇔⇔ ? V 0⇔	11101101 01ss1010	ED	2	2
ADD HL, ss	HL<-HL+ss	● ● ? ● 0⇔	00ss1001		1	1
ADD IX, pp	IX<-IX+pp	● ● ? ● 0⇔	11011101 00pp1001	DD	2	2
ADD IY, rr	IY<-IY+rr	● ● ? ● 0⇔	11111101 00rr1001	FD	2	2
INC r	r<-r+1	⇔⇔⇔V 0●	00 r 100		1	1
INC [HL]	[HL]<-[HL]+1	⇔⇔⇔V 0●	00110100	34	1	4
INC [IX+d]	[IX+d]<-[IX+d]+1	⇔⇔⇔V 0●	11011101 00110100 <- d ->	DD 34	3	7
INC [IY+d]	[IY+d]<-[IY+d]+1	⇔⇔⇔V 0●	11111101 00110100 <- d ->	FD 34	3	7
INC ss	ss<-ss+1	● ● ● ● ● ●	00ss0011		1	1
INC IX	IX<-IX+1	● ● ● ● ● ●	11011101	DD	2	2

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
INC IY	IY<-IY+1	● ● ● ● ● ●	00100011	23	2	2
			11111101	FD		
			00100011	23		

	00	01	10	11
ss	BC	DE	HL	SP
pp	BC	DE	IX	SP
rr	BC	DE	<del>IX</del>	SP

IY

### SUBTRACT GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
SUB r	A<-A-r	⇔⇔⇔V 1⇔	10010 r		1	1
SUB n	A<-A-n	⇔⇔⇔V 1⇔	11010110	D6	2	2
			< - n - >			
SUB [HL]	A<-A-[HL]	⇔⇔⇔V 1⇔	10010110	96	1	2
SUB [IX+d]	A<-A-[IX+d]	⇔⇔⇔V 1⇔	11011101	DD	3	5
			10010110	96		
			< - d - >			
SUB [IY+d]	A<-A-[IY+d]	⇔⇔⇔V 1⇔	11111101	FD	3	5
			10010110	96		
			< - d - >			
SBC A, r	A<-A-r-C	⇔⇔⇔V 1⇔	10011 r		1	1
SBC A, n	A<-A-n-C	⇔⇔⇔V 1⇔	11011110	DE	2	2
			< - n - >			
SBC A, [HL]	A<-A-[HL]-C	⇔⇔⇔V 1⇔	10011110	9E	1	2
SBC A, [IX+d]	A<-A-[IX+d]-C	⇔⇔⇔V 1⇔	11011101	DD	3	5
			10011110	9E		
			< - d - >			
SBC A, [IY+d]	A<-A-[IY+d]-C	⇔⇔⇔V 1⇔	11111101	FD	3	5
			10011110	9E		
			< - d - >			
SBC HL, ss	HL<-HL-ss-C	⇔⇔ ? V 1⇔	11101101	ED	2	2
			01ss0010			
DEC r	r<-r-1	⇔⇔⇔V 1●	00 r 101		1	1
DEC [HL]	[HL]<-[HL]-1	⇔⇔⇔V 1●	00110101	35	1	4
DEC [IX+d]	[IX+d]<-[IX+d]-1	⇔⇔⇔V 1●	11011101	DD	3	7
			00110101	35		
			< - d - >			
DEC [IY+d]	[IY+d]<-[IY+d]-1	⇔⇔⇔V 1●	11111101	FD	3	7
			00110101	35		
			< - d - >			

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
DEC ss	ss<-ss-1	● ● ● ● ● ●	00ss1011		1	1
DEC IX	IX<-IX-1	● ● ● ● ● ●	11011101	DD	2	2
DEC IY	IY<-IY-1	● ● ● ● ● ●	00101011	2B	2	2
			11111101	FD		
			00101011	2B		

### COMPARE GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
CP r	A-r	⇔ ⇔ ⇔ V 1 ⇔	10111 r		1	1
CP n	A-n	⇔ ⇔ ⇔ V 1 ⇔	11111110	FE	2	2
			< - n - >			
CP [HL]	A-[HL]	⇔ ⇔ ⇔ V 1 ⇔	10111110	BE	1	2
CP [IX+d]	A-[IX+d]	⇔ ⇔ ⇔ V 1 ⇔	11011101	DD	3	5
			10111110	BE		
			< - d - >			
CP [IY+d]	A-[IY+d]	⇔ ⇔ ⇔ V 1 ⇔	11111101	FD	3	5
			10111110	BE		
			< - d - >			

### LOGICAL GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
AND r	A<-A ^ r	⇔ ⇔ 1 P 0 0	10100 r		1	1
AND n	A<-A ^ n	⇔ ⇔ 1 P 0 0	11100110	E6	2	2
			< - n - >			
AND [HL]	A<-A ^ [HL]	⇔ ⇔ 1 P 0 0	10100110	A6	1	2
AND [IX+d]	A<-A ^ [IX+d]	⇔ ⇔ 1 P 0 0	11011101	DD	3	5
			10100110	A6		
			< - d - >			
AND [IY+d]	A<-A ^ [IY+d]	⇔ ⇔ 1 P 0 0	11111101	FD	3	5
			10100110	A6		
			< - d - >			
OR r	A<-A ∨ r	⇔ ⇔ 0 P 0 0	10110 r		1	1
OR n	A<-A ∨ n	⇔ ⇔ 0 P 0 0	11110110	F6	2	2
			< - n - >			
OR [HL]	A<-A ∨ [HL]	⇔ ⇔ 0 P 0 0	10110110	B6	1	2
OR [IX+d]	A<-A ∨ [IX+d]	⇔ ⇔ 0 P 0 0	11011101	DD	3	5
			10110110	B6		
			< - d - >			

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	C L
			76543210	Hex		
OR [IY+d]	A<-A ∨ [IY+d]	⇔ ⇔ 0 P 0 0	11111101 10110110 < - d - >	FD B6	3	5
XOR r	A<-A ∨ r	⇔ ⇔ 0 P 0 0	10101 r		1	1
XOR n	A<-A ∨ n	⇔ ⇔ 0 P 0 0	11101110 < - n - >	EE	2	2
XOR [HL]	A<-A ∨ [HL]	⇔ ⇔ 0 P 0 0	10101110	AE	1	2
XOR [IX+d]	A<-A ∨ [IX+d]	⇔ ⇔ 0 P 0 0	11011101 10101110 < - d - >	DD AE	3	5
XOR [IY+d]	A<-A ∨ [IY+d]	⇔ ⇔ 0 P 0 0	11111101 10101110 < - d - >	FD AE	3	5

### BIT SET AND TEST GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	C L
			76543210	Hex		
BIT b, r	Z<-NOT r{b}	? ⇔ 1 ? 0 ●	11001011 01 b r	CB	2	2
BIT b, [HL]	Z<-NOT [HL]{b}	? ⇔ 1 ? 0 ●	11001011 01 b 110	CB	2	3
BIT b, [IX+d]	Z<-NOT [IX+d]{b}	? ⇔ 1 ? 0 ●	11011101 11001011 < - d - >	DD CB	4	5
BIT b, [IY+d]	Z<-NOT [IY+d]{b}	? ⇔ 1 ? 0 ●	01 b 110 11111101 11001011 < - d - >	FD CB	4	5
SET b, r	r{b}<-1	● ● ● ● ● ●	01 b 110 11001011 11 b r	CB	2	2
SET b, [HL]	[HL]{b}<-1	● ● ● ● ● ●	11001011 11 b 110	CB	2	5
SET b, [IX+d]	[IX+d]{b}<-1	● ● ● ● ● ●	11011101 11001011 < - d - >	DD CB	4	7
SET b, [IY+d]	[IY+d]{b}<-1	● ● ● ● ● ●	11 b 110 11111101 11001011 < - d - >	FD CB	4	7
RES b, r	r{b}<-0	● ● ● ● ● ●	11 b 110 11001011 10 b r	CB	2	2

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	C L
			76543210	Hex		
RES b, [HL]	[HL]{b}<-0	● ● ● ● ● ●	11001011 10 b 110	CB	2	5
RES b, [IX+d]	[IX+d]{b}<-0	● ● ● ● ● ●	11011101 11001011 <- d -> 10 b 110	DD CB	4	7
RES b, [IY+d]	[IY+d]{b}<-0	● ● ● ● ● ●	11111101 11001011 <- d -> 10 b 110	FD CB	4	7

### ROTATE GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	C L
			76543210	Hex		
RLCA	C<-A{7};A<-A*2 A{0}<-C	● ● 0 ● 0 ⇔	00000111	07	1	1
RRCA	C<-A{0};A<-A/2 a{7}<-C	● ● 0 ● 0 ⇔	00001111	0F	1	1
RLA	tmp<-C;C<-A{7} A<-A*2 A{0}<-tmp	● ● 0 ● 0 ⇔	00010111	17	1	1
RRA	tmp<-C;C<-A{0} A<-A/2 A{7}<-tmp	● ● 0 ● 0 ⇔	00011111	1F	1	1
RLC r	C<-r{7} r<-r*2;r{0}<-C	⇔ ⇔ 0 P 0 ⇔	11001011 00000 r	CB	2	2
RLC [HL]	C<-[HL]{7} [HL]<-[HL]*2 [HL]{0}<-C	⇔ ⇔ 0 P 0 ⇔	11001011 00000110	CB 06	2	5
RLC [IX+d]	C<-[IX+d]{7} [IX+d]<-[IX+d]*2 [IX+d]{0}<-C	⇔ ⇔ 0 P 0 ⇔	11011101 11001011 <- d -> 00000110	DD CB 06	4	7
RLC [IY+d]	C<-[IY+d]{7} [IY+d]<-[IY+d]*2 [IY+d]{0}<-C	⇔ ⇔ 0 P 0 ⇔	11111101 11001011 <- d -> 00000110	FD CB 06	4	7
RRC r	C<-r{0} r<-r/2;r{7}<-C	⇔ ⇔ 0 P 0 ⇔	11001011 00001 r	CB	2	2
RRC [HL]	C<-[HL]{0} [HL]<-[HL]/2 [HL]{7}<-c	⇔ ⇔ 0 P 0 ⇔	11001011 00001110	CB 0E	2	5

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
RRC [IX+d]	C<-[IX+d]{0}	⇔ ⇔ 0 P 0 ⇔	11011101	DD	4	7
	[IX+d]<-[IX+d]/2		11001011	CB		
	[IX+d]{7}<-C		<- d ->			
			00001110	0E		
RRC [IY+d]	C<-[IY+d]{0}	⇔ ⇔ 0 P 0 ⇔	11111101	FD	4	7
	[IY+d]<-[IY+d]/2		11001011	CB		
	[IY+d]{7}<-C		<- d ->			
			00001110	0E		
RL r	tmp<-C;C<-r{7}	⇔ ⇔ 0 P 0 ⇔	11001011	CB	2	2
	r<-r*2;r{0}<-tmp		00010 r			
RL [HL]	tmp<-C;C<-[HL]{7}	⇔ ⇔ 0 P 0 ⇔	11001011	CB	2	5
	[HL]<-[HL]*2		00010110	16		
RL [IX+d]	tmp<-C	⇔ ⇔ 0 P 0 ⇔	11011101	DD	4	7
	C<-[IX+d]{7}		11001011	CB		
	[IX+d]<-[IX+d]*2		<- d ->			
	[IX+d]{0}<-tmp		00010110	16		
RL [IY+d]	tmp<-C	⇔ ⇔ 0 P 0 ⇔	11111101	FD	4	7
	C<-[IY+d]{7}		11001011	CB		
	[IY+d]<-[IY+d]*2		<- d ->			
	[IY+d]{0}<-tmp		00010110	16		
RR r	tmp<-C;C<-r{0}	⇔ ⇔ 0 P 0 ⇔	11001011	CB	2	2
	r<-r/2;r{7}<-tmp		00011 r			
RR [HL]	tmp<-C;C<-r{0}	⇔ ⇔ 0 P 0 ⇔	11001011	CB	2	5
	[HL]<-[HL]/2		00011110	1E		
RR [IX+d]	tmp<-C	⇔ ⇔ 0 P 0 ⇔	11011101	DD	4	7
	C<-[IX+d]{0}		11001011	CB		
	[IX+d]<-[IX+d]/2		<- d ->			
	[IX+d]{7}<-tmp		00011110	1E		
RR [IY+d]	tmp<-C	⇔ ⇔ 0 P 0 ⇔	11111101	FD	4	7
	C<-[IY+d]{0}		11001011	CB		
	[IY+d]<-[IY+d]/2		<- d ->			
	[IY+d]{7}<-tmp		00011110	1E		

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
RLD	tmp<-A{0..3} A{0..3} <-[HL]{4..7} [HL]{4..7}	⇔ ⇔ 0 P 0 ⇔	11101101 01101111	ED 6F	2	5
RRD	<-[HL]{0..3} [HL]{0..3}<-tmp tmp<-A{0..3} A{0..3} <-[HL]{0..3} [HL]{0..3}	⇔ ⇔ 0 P 0 ⇔	11101101 01100111	ED 67	2	5
	<-[HL]{4..7} [HL]{4..7}<-tmp					

### SHIFT GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
SLA r	C<-r{7} r<-r*2	⇔ ⇔ 0 P 0 ⇔	11001011 00100 r	CB	2	2
SLA [HL]	C<-[HL]{7} [HL]<-[HL]*2	⇔ ⇔ 0 P 0 ⇔	11001011 00100110	CB 26	2	5
SLA [IX+d]	C<-[IX+d]{7} [IX+d]<-[IX+d]*2	⇔ ⇔ 0 P 0 ⇔	11011101 11001011 <- d -> 00100110	DD CB 26	4	7
SLA [IY+d]	C<-[IY+d]{7} [IY+d]<-[IY+d]*2	⇔ ⇔ 0 P 0 ⇔	11111101 11001011 <- d -> 00100110	FD CB 26	4	7
SRL r	C<-r{0} r<-r/2	⇔ ⇔ 0 P 0 ⇔	11001011 00111 r	CB	2	2
SRL [HL]	C<-[HL]{0} [HL]<-[HL]/2	⇔ ⇔ 0 P 0 ⇔	11001011 00111110	CB 3E	2	5
SRL [IX+d]	C<-[IX+d]{0} [IX+d]<-[IX+d]/2	⇔ ⇔ 0 P 0 ⇔	11011101 11001011 <- d -> 00111110	DD CB 3E	4	7
SRL [IY+d]	C<-[IY+d]{0} [IY+d]<-[IY+d]/2	⇔ ⇔ 0 P 0 ⇔	11111101 11001011 <- d -> 00111110	FD CB 3E	4	7
SRA r	tmp<-r{7};C<-r{0} r<-r/2	⇔ ⇔ 0 P 0 ⇔	11001011 00101 r	CB	2	2

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
SRA [HL]	r{7}<-tmp	⇔ ⇔ 0 P 0 ⇔	11001011	CB	2	5
	tmp<-[HL]{7}		00101110	2E		
SRA [IX+d]	C<-[HL]{0}	⇔ ⇔ 0 P 0 ⇔	11011101	DD	4	7
	[HL]<-[HL]/2		11001011	CB		
SRA [IY+d]	[HL]{7}<-tmp	⇔ ⇔ 0 P 0 ⇔	< - d - >		4	7
	tmp<-[IX+d]{7}		00101110	2E		
SRA [IY+d]	C<-[IX+d]{0}	⇔ ⇔ 0 P 0 ⇔	11111101	FD	4	7
	[IX+d]<-[IX+d]/2		11001011	CB		
SRA [IY+d]	[IX+d]{7}<-tmp	⇔ ⇔ 0 P 0 ⇔	< - d - >		4	7
	tmp<-[IY+d]{7}		00101110	2E		
SRA [IY+d]	C<-[IY+d]{0}	⇔ ⇔ 0 P 0 ⇔	11111101	FD	4	7
	[IY+d]<-[IY+d]/2		11001011	CB		
SRA [IY+d]	[IY+d]{7}<-tmp	⇔ ⇔ 0 P 0 ⇔	< - d - >		4	7
	tmp<-[IY+d]{7}		00101110	2E		

### JUMP GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
JP nn	PC<-nn	● ● ● ● ● ● ●	11000011	C3	3	3
JP NZ, nn	if Z=0 PC<-nn	● ● ● ● ● ● ●	< - n n l - >		3	3
			< - n n h - >			
JP Z, nn	if Z=1 PC<-nn	● ● ● ● ● ● ●	11000010	C2	3	3
			< - n n l - >			
JP NC, nn	if C=0 PC<-nn	● ● ● ● ● ● ●	< - n n h - >		3	3
			11001010	CA		
JP C, nn	if C=1 PC<-nn	● ● ● ● ● ● ●	< - n n l - >		3	3
			11010010	D2		
JP PO, nn	if PV=0 PC<-nn	● ● ● ● ● ● ●	< - n n h - >		3	3
			11011010	DA		
JP PE, nn	if PV=1 PC<-nn	● ● ● ● ● ● ●	11100010	E2	3	3
			< - n n l - >			
JP P, nn	if S=0 PC<-nn	● ● ● ● ● ● ●	< - n n h - >		3	3
			11101010	EA		
JP M, nn	if S=1 PC<-nn	● ● ● ● ● ● ●	< - n n l - >		3	3
			11110010	F2		
JP M, nn	if S=1 PC<-nn	● ● ● ● ● ● ●	< - n n h - >		3	3
			11111010	FA		

Mnemonic	Operation	F l a g s						O p c o d e		B	CL	
		S	Z	H	P	V	N	C	76543210			Hex
	PC<-nn								< - nnl ->			
									< - nnh ->			
JP [HL]	PC<-[HL]	●	●	●	●	●	●	●	11101001	E9	1	1
JP [IX]	PC<-[IX]	●	●	●	●	●	●	●	11011101	DD	2	2
									11101001	E9		
JP [IY]	PC<-[IY]	●	●	●	●	●	●	●	11111101	FD	2	2
									11101001	E9		
JR e	PC<-PC+e	●	●	●	●	●	●	●	00011000	18	2	3
									< -e-2 ->			
JR NZ, e	if Z=0	●	●	●	●	●	●	●	00100000	20	2	2
	PC<-PC+e								< -e-2 ->			3
JR Z, e	if Z=1	●	●	●	●	●	●	●	00101000	28	2	2
	PC<-PC+e								< -e-2 ->			3
JR NC, e	if C=0	●	●	●	●	●	●	●	00110000	30	2	2
	PC<-PC+e								< -e-2 ->			3
JR C, e	if C=1	●	●	●	●	●	●	●	00111000	38	2	2
	PC<-PC+e								< -e-2 ->			3
DJNZ e	B<-B-1; if B><0	●	●	●	●	●	●	●	00010000	10	2	3
	PC<-PC+e								< -e-2 ->			
CALL nn	[SP-2]<-PCL	●	●	●	●	●	●	●	11001101	CD	3	5
	[SP-1]<-PCH								< - nnl ->			
	SP<-SP-2								< - nnh ->			
	PC<-nn											
CALL NZ, nn	if Z=0	●	●	●	●	●	●	●	11000100	C4	3	3
	[SP-2]<-PCL								< - nnl ->			5
	[SP-1]<-PCH								< - nnh ->			
	SP<-SP-2											
	PC<-nn											
CALL Z, nn	if Z=1	●	●	●	●	●	●	●	11001100	CC	3	3
	[SP-2]<-PCL								< - nnl ->			5
	[SP-1]<-PCH								< - nnh ->			
	SP<-SP-2											
	PC<-nn											
CALL NC, nn	if C=0	●	●	●	●	●	●	●	11010100	D4	3	3
	[SP-2]<-PCL								< - nnl ->			5
	[SP-1]<-PCH								< - nnh ->			
	SP<-SP-2											
	PC<-nn											
CALL C, nn	if C=1	●	●	●	●	●	●	●	11011100	DC	3	3
	[SP-2]<-PCL								< - nnl ->			5
	[SP-1]<-PCH								< - nnh ->			
	SP<-SP-2											
	PC<-nn											
CALL PO, nn	if PV=0	●	●	●	●	●	●	●	11100100	E4	3	3

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
CALL PE, nn	[SP-2]<-PCL		< - nnl - >			5
	[SP-1]<-PCH		< - nnh - >			
CALL P, nn	SP<-SP-2					
	PC<-nn	● ● ● ● ● ●	11101100	EC	3	3
CALL M, nn	if PV=1		< - nnl - >			5
	[SP-2]<-PCL		< - nnh - >			
RET NZ	[SP-1]<-PCH					
	SP<-SP-2					
RET Z	PC<-nn	● ● ● ● ● ●	111100100	F4	3	3
	if S=0		< - nnl - >			5
RET C	[SP-2]<-PCL		< - nnh - >			
	[SP-1]<-PCH					
RET NC	SP<-SP-2					
	PC<-nn	● ● ● ● ● ●	11111100	FC	3	3
RET PE	if S=1		< - nnl - >			5
	[SP-2]<-PCL		< - nnh - >			
RET PO	[SP-1]<-PCH					
	SP<-SP+2					
RET NZ	PCL<-[SP]	● ● ● ● ● ●	11001001	C9	1	3
	PCH<-[SP+1]					
RET Z	SP<-SP+2					
	if Z=0	● ● ● ● ● ●	11000000	C0	1	2
RET C	PCL<-[SP]					4
	PCH<-[SP+1]					
RET PE	SP<-SP+2					
	if Z=1	● ● ● ● ● ●	11001000	C8	1	2
RET NC	PCL<-[SP]					4
	PCH<-[SP+1]					
RET PO	SP<-SP+2					
	if C=0	● ● ● ● ● ●	11010000	D0	1	2
RET PE	PCL<-[SP]					4
	PCH<-[SP+1]					
RET PE	SP<-SP+2					
	if C=1	● ● ● ● ● ●	11011000	D8	1	2
RET PE	PCL<-[SP]					4
	PCH<-[SP+1]					
RET PE	SP<-SP+2					
	if PV=0	● ● ● ● ● ●	11100000	E0	1	2
RET PE	PCL<-[SP]					4
	PCH<-[SP+1]					
RET PE	SP<-SP+2					
	if PV=1	● ● ● ● ● ●	11101000	E8	1	2
RET PE	PCL<-[SP]					4

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
RET P	PCH<-[SP+1] SP<-SP+2 if S=0	● ● ● ● ● ●	11110000	F0	1	2 4
	PCL<-[SP] PCH<-[SP+1] SP<-SP+2					
RET M	if S=1	● ● ● ● ● ●	11111000	F8	1	2 4
	PCL<-[SP] PCH<-[SP+1] SP<-SP+2					
RETI	interrupt return	● ● ● ● ● ●	11101101 01001101	ED 4D	2	7
RETN	non maskable interrupt return	● ● ● ● ● ●	11101101 01000101	ED 45	2	4
RST k	[SP-2]<-PCL [SP-1]<-PCH SP<-SP-2;PCL<-k PCH<-0	● ● ● ● ● ●	11k/8111		1	4

### INPUT AND OUTPUT GROUP

Mnemonic	Operation	F l a g s S Z H P V N C	Opcode		B	CL
			76543210	Hex		
IN A, [n]	A<-[n]	● ● ● ● ● ●	11011011 <- n ->	DB	2	4
IN r, [C]	r<-[C]	⇔ ⇔ 0 P 0 ●	11101101 01 r 000	ED	2	4
INI	[HL]<-[C] B<-B-1 HL<-HL+1	? ⇔ ? ? 1 ● *1	11101101 10100010	ED A2	2	5
IND	[HL]<-[C] B<-B-1 HL<-HL-1	? ⇔ ? ? 1 ● *1	11101101 10101010	ED AA	2	5
INIR	repeat;[HL]<-[C] B<-B-1 HL<-HL+1 until B=0	? 1 ? ? 1 ●	11101101 10110010	ED B2	2	6 6
INDR	repeat;[HL]<-[C] B<-B-1 HL<-HL-1 until B=0	? 1 ? ? 1 ●	11101101 10111010	ED BA	2	6 6
OUT [n], A	[n]<-A	● ● ● ● ● ●	11010011	D3	2	4

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
OUT [C], r	[C]<-r	● ● ● ● ● ●	< - n - > 11101101 01r 001	ED	2	4
OUTI	[C]<-[HL] B<-B-1 HL<-HL+1	? ⇔ ? ? ? 1 ● *1	11101101 10100011	ED A3	2	5
OUTD	[C]<-[HL] B<-B-1 HL<-HL-1	? ⇔ ? ? ? 1 ● *1	11101101 10101011	ED AB	2	5
OTIR	repeat:[C]<-[HL] B<-B-1 HL<-HL+1 until B=0	? 1 ? ? ? 1 ●	11101101 10110011	ED B3	2	7 7
OTDR	repeat:[C]<-[HL] B<-B-1 HL<-HL-1 until B=0	? 1 ? ? ? 1 ●	11101101 10111011	ED BB	2	7 7

\*1 If B=0, Z=1, else Z=0

#### GENERAL-PURPOSE ARITHMETIC AND CPU CONTROL GROUPS

Mnemonic	Operation	F l a g s S Z H P V N C	O p c o d e		B	CL
			76543210	Hex		
DAA	adjust to decimal	⇔ ⇔ ⇔ P ● ⇔	00100111	27	1	1
CPL	A<-NOT A	● ● 1 ● 1 ●	00101111	2F	1	1
NEG	A<-NOT A+1	⇔ ⇔ ⇔ V 1 ⇔	11101101 01000100	ED 44	2	2
CCF	C<-NOT C	● ● ? ● 0 ⇔	00111111	3F	1	1
SCF	C<-1	● ● 0 ● 0 1	00110111	37	1	1
NOP	NO operation	● ● ● ● ● ●	00000000	00	1	1
HALT	HALT	● ● ● ● ● ●	01110110	76	1	2
DI	IFF<-0	● ● ● ● ● ●	11110011	F3	1	2
EI	IFF<-1	● ● ● ● ● ●	11111011	FB	1	2
IM 0	interrupt mode 0	● ● ● ● ● ●	11101101 01000110	ED 46	2	2
IM 1	interrupt mode 1	● ● ● ● ● ●	11101101 01010110	ED 56	2	2
IM 2	interrupt mode 2	● ● ● ● ● ●	11101101 01011110	ED 5E	2	2



- 本製品、および本書に記載された、情報・回路の使用に対して、当社は第三者の工業所有権、知的財産権、および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。
- 本製品は、外国為替および外国貿易管理法に定める戦略物資に該当しますので、本製品を輸出する場合、同法に基づく輸出許可が必要です。
- 本製品を、極めて高い信頼性が要求される用途にご使用の場合は、事前に当社営業までご連絡下さい。

## 川崎製鉄株式会社

LSI事業部 営業部

東京都千代田区内幸町2-2-3 〒100

TEL (03) 3597-4619(ダイヤルイン) FAX (03) 3597-3634