MSX-TIMER

仕様書

目次

1. 概要

本書は、MSX-TIMER の仕様書である。制御方法、動作タイミングを規定するものである。

MSX-TIMER には、大まかに下記の機能がある。

(1) ワンショットタイマー
(2) コンティニューアスタイマー
(3) フリーランカウンター
(4) 割り込み
(5) 精度指定
(6) 4つのタイマー

各機能付いて、次章以降で詳細に説明する。

MSX-TIMER は、Y8960 Cartridge、MSX2++以降の本体に搭載される。

基本的に、インクリメントカウンターによるシンプルなタイマー回路である。

2. I/O ポート

I/O ポートは、4ポート占有する。下記に表 1 I/O ポート一覧を示す。

表 1 I/O ポート一覧

ポート番号	機能
B0h	レジスタ番号を指定する。
B1h	B0h で指定したレジスタを読み書きする。
B2h	現在発生中の割り込みフラグの読み出しとクリア。
B3h	カウンターの読みだし。

2.1. レジスタの読み書き

レジスタへの書き込みは、下記のように B0h に書き込みたいレジスタ番号を書き込み後、B1h にそのレジスタへ書き込みたい値を書き込む。

```
DI
LD    A, register_number
OUT   (0B0h), A
LD    A, value
OUT   (0B1h), A
EI
```

レジスタの読みだしは、下記のように B0h に読みだしたいレジスタ番号を書き込み後、B1h を読みだす。

```
DI
LD          A, register_number
OUT         (0B0h), A
IN          A, (0B1h)
EI
```

B0h に書いた値は、B0h を読みだすことで読むことができるため、例えば割り込み処理の中で、もとの値を読みだして保持しておき、割り込みから抜けるときにもとの値へ戻すことも出来る。

```
interrupt_handler:
        (レジスタの待避など)
        IN      A, (0B0h)
        PUSH    AF
        (割り込み内の処理)
        POP     AF
        OUT     (0B0h), A
        (レジスタの復帰など)
        RET
```

2.2. 割り込み要因のチェック

MSX は、標準的には Z80 の IM1 で動作する。割り込みが発生すると 0038h がコールされ、そこから BIOS の割り込み処理ルーチンへジャンプする。そして、真っ先にフック H.KEYI (FD9Ah) にジャンプする。通常は、H.KEYI に割り込み処理ルーチンをフックすることで、割り込みを受けるが、VDP の割り込みも、MSX-TIMER の割り込みも、全てここを通過するため、割り込み処理ルーチンは「自身が処理すべき割り込みなのか？」を判定して、そうで無い場合は速やかに戻る必要がある。

MSX-TIMER は、独立した 4つのタイマーを持っているが、この４つのどの割り込みが発生したかを、I/O B2h を一回読むだけで判断する情報を得ることができる。

読みだした値(8bit) の各ビットには、 T0～T3 の名前が付けられている。

図1. B2h のビットマップ

読みだした値のうち、bit4-7 の Reserve には 0 が入っている。

bit0 が T0, bit1 が T1, bit2 が T2, bit3 が T3 に対応しており、それぞれ Timer0～Timer3 の割り込み要因フラグに対応している。つまり「T0=1 なら Timer0 の割り込みが発生している状態」を示しているのである。「TN=1 なら TimerN の割り込みが発生している状態」である。

このどれか１つでも 1 である状態の時に、CPU が割り込みを受けられる状態になると、割り込みが発生することになる。割り込みハンドラーは、EI する前に「自身が処理したタイマー割り込み」をクリアする必要がある。

クリアは、T0～T3 のクリアしたい bit に 1 を立てた値を B2h へ書き込む事により実現する。

従って、MSX-TIMER の割り込み処理を全て引き受ける割り込みハンドラであれば、B2h から読んだ値をそのまま書き込んでやればクリア出来るのである。

2.3. カウンターの読みだし

B3h の下位 2bit に対して、0～3 の値を書き込み、その後に B3h を読みだすとカウント値を読みだすことができる。

3. レジスタ

４つのタイマーは、全て同じ機能を持つ。周期等を個別に指定できるため、目的に応じて４つのタイマーを使い分けることができる。

１つのタイマーモジュールには、表 1 レジスター一覧 に示す2個のレジスタが存在する。

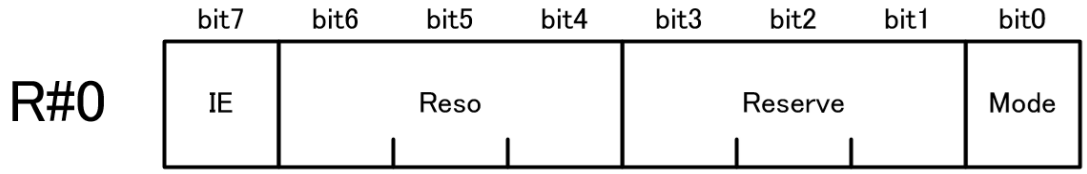
表 1 レジスター一覧

レジスタアドレス	機能
0	モードレジスタ。挙動を決めるレジスタである。
1	カウントレジスタ。
2	コントロールレジスタ。

3.1. モードレジスタ

モードレジスタ (0) のビットマップを図1 に示す。

図1. モードレジスタのビットマップ



Mode の設定値を、表 2 Mode レジスタの設定値一覧に示す。

表 2 Mode レジスタの設定値一覧

設定値	意味
0	one shot
1	repeat

one shot (0) にすると、所定の時間が経過したらそこで止まる。

repeat (1) にすると、所定の時間が経過したら、0 に戻る。

いずれの場合も、「所定の時間」はカウントレジスタを使って指定する。

Reso の設定値を、表 3 Reso レジスタの設定値一覧に示す。

表 3 Reso レジスタの設定値一覧

設定値	意味
0	11.92 [usec] 単位
1	23.84 [usec] 単位
中略	-
6	71.58 [usec] 単位
7	83.48 [usec] 単位

クロック源は、85.90908MHz である。内部に 32bit のカウンターを持っており、0～4,294,967,295 までカウントできる。

4,294,967,295 / 85,909,080 ≒ 49.99 [sec]

最大で、49.99[sec] までカウントできる。

しかし、MSX で32bit カウンターは扱いにくいことと、1カウント ≒ 11.64 [nsec] と、分解能が高すぎるため、これを 1024分周した 11.92 [usec] （内部 32bit カウンターの下位 10bit を無視）を単位としてカウントレジスタに指定する値の分解能を指定する。

IE の設定値を、表 4 IE レジスタの設定値一覧に示す。

表 4 IE レジスタの設定値一覧

設定値	意味
0	interrupt disable
1	interrupt enable

IE レジスタは、割り込みの発生を制御するレジスタである。

3.2. カウントレジスタ

タイマーの時間を指定するレジスタがカウントレジスタである。

1カウントに対応する時間は、モードレジスタ内の Reso レジスタで指定する。

カウント値としては、0～255 を指定できる。

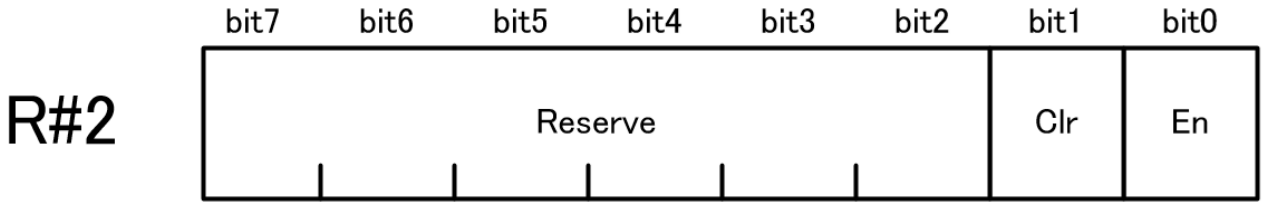
実際の待機時間は、(カウント値 + 1) * 単位時間 となる。

例えば、reso = 0 の場合、11.92[usec] 単位となるので、カウント値 が 0 なら、11.92[usec]後に割り込み要因が立つことになる。

3.3. コントロールレジスタ

コントロールレジスタ (2) のビットマップを図1に示す。

図1. コントロールレジスタのビットマップ



En の設定値を、表 5 En レジスタの設定値一覧に示す。

表 5 En レジスタの設定値一覧

設定値	意味
0	count disable
1	count enable

count disable (0) にすると、カウンターのインクリメントが停止する。一次停止。

count enable (1) にすると、カウンターがインクリメントする。稼働状態。

Clr は、1 を書き込むとカウンターをクリアする。読み出し時は常に 0 になる。

Reserve は、将来予約のスペースである。書き込み時には 0 を書き込むこと。読み出し時には 0 が返ってくる。

4. 詳細な動作説明

T.B.D.

更新履歴

2026/Jan/14th　v0.1　HRA!　初版