

第 5 章

サブCPU

5-1 サブ CPU

5-1-1 サブ CPU の構成

X1 シリーズでは、メイン CPU の負担を軽減するために、2 つのワンチップ CPU を搭載しています。これらの CPU は、キーボード、カセットなどのコントロールを行い、メイン CPU はサブ CPU に命令を送るだけで、各種の処理が行われるようになっています。

サブ CPU とメイン CPU 周りの構成を、図 5-1 に示します。但し、この図は X1turbo model -20 の場合で、機種によって構成は若干の違いがあります。

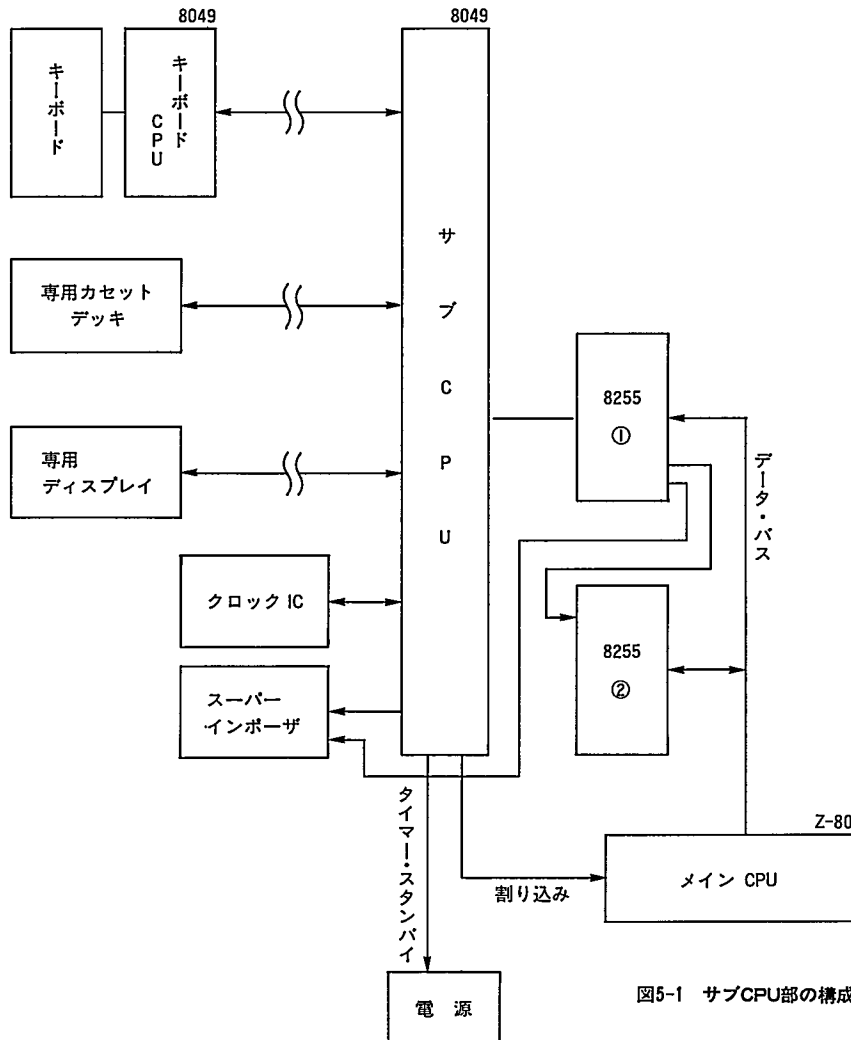


図5-1 サブCPU部の構成

ここで使われている CPU8049(80C49)は、8ビットのワンチップCPUで、2KバイトのROM、ワーク用のRAM、I/Oポートなどが内蔵されています。ROMには、サブCPUのコントロールプログラムが、ICの製造段階で書き込まれています。

ここでは、2つの8049を各々、キーボードCPU、サブCPUと呼ぶことにします(X1では、キーボードCPUは8048で、一部の機能に制限があります)。図5-1からもわかるように、これらのCPUは、以下のようなことを行っています。

(1) キーボードCPU

- ・キーボードをスキャンし、押されたキーのデータをサブCPUに送信

(2) サブCPU

- ・キーボードCPUからのデータ受信
- ・専用カセットデッキのコントロール
- ・タイマICの時刻データの読み書き
- ・メインCPUとのデータの受け渡し

サブCPUとキーボードCPUの電源は、メインCPU系統とは分離されています。フロントのスイッチがoffになっていても、後面パネルのスイッチが入っていれば、サブCPU系には電源が供給され、動作を続けます。このためフロントスイッチがoffであっても、キーボードを用いた各種のTVコントロールとTVタイマー機能は動作するのです。

サブCPUは、BASICを用いている限り、意識する必要はありません。しかし、マシン語を使って、I/Oを制御しようとするとしてもサブCPUとデータのやり取りを行わなければなりません。以降、サブCPUの使い方を説明していきます。

5-1-2 直接アクセスによるコマンドの送受信

サブCPUには、メインCPUと独立して動作しています。サブCPUに何かのコントロールを実行させたり、結果を受け取ったりするには、決められた手続きに従ってコマンドを送らなければなりません。サブとメインには、図5-1に示すように、2つの8255で接続されています。サブCPUとの通信に必要なI/Oポートは、以下に示す通りです。

・8255②ポートB(入力)

I/Oアドレス=1A01H

ビット6:IBF(Input Buffer Full)

Lの時8049にデータを送っても良い

ビット5: $\overline{\text{OBF}}$ (Output Buffer Full)

Lなら8049からのデータがある

・8255①ポートA(サブCPUの管理下にある)

I/Oアドレス=1900H

サブCPUとのデータの入出力

サブCPUと、直接データのやり取りをする場合は、データのぶつかり合いが起これないように、IBFと $\overline{\text{OBF}}$ の2本の制御線を用いて通信します。

(1) サブ CPU にコマンドやデータを渡す場合

- 1) IBF を読み込む
- 2) IBF が 1 なら, 0 になるまで待つ
- 3) データを 1 バイト 1900H に書き込む
- 4) まだ送るデータがあるなら 1)へ

(2) サブ CPU からデータを読み込む場合

- 1) $\overline{\text{OBF}}$ を読み込む
- 2) $\overline{\text{OBF}}$ が 1 なら, 0 になるまで待つ
- 3) データを 1 バイト 1900H から読み込む
- 4) まだ読み込むべきデータがあるなら 1)へ

サンプルプログラムをリスト 5-1 と 5-2 に示します。

リスト 5-1 Z-80 から 80C49 にデータを送る場合

```

GET49:  LD      BC, 01A01H
GET49_1: IN      A, (C)
        AND     040H
        JR      NZ, GET49_1
        LD      BC, 01900H
        LD      A, 0E3H
        OUT     (C), A
        RET
        ;
        END

```

80C49 がデータを受け取るまで待つ

ゲームキーデータ送信要求コマンド

リスト 5-2 Z-80 が 80C49 からデータを受け取る場合

```

RCV49:  LD      BC, 1A01H
R49_1:  IN      A, (C)
        AND     20H
        JR      NZ, R49_1
        LD      BC, 01900H
        IN      A, (C)
        RET
        ;
        END

```

80C49 からデータが送られるまで待つ

データを受け取る

サブ CPU との通信は、このように送る手順が決っており、バイト数はコマンドの種類によって変わってきます。従って、サブ CPU の実行状態と、メイン CPU の要求が何かのはずみで食い違っていると、サブ・メイン間の通信が不可能となり、システムがハング・アップする可能性があります。プログラムの起動直後など、サブ CPU の状態が不明の時は、サブ CPU の初期化を行う必要があります。このためのサブルーチンを、リスト 5-3 に示します。

リスト 5-3 80C49 を初期化する

```

S49RES EQU      13E5H
INT49:  LD      BC, 1A01H
        IN      A, (C)
        AND     10H
        JR      NZ, C49_1
        LD      A, 1EH
        JR      C49_2
C49_1:  LD      A, 1EH
C49_2:  PUSH    AF

```

BIOS ROM がアクティブかどうか調べ、その状態をセーブする

LD	A, 1DH	ROMをアクティブに
OUT	(00H), A	
CALL	S49RES	ROMの状態を元に戻す
POP	AF	
OUT	(00H), A	
RET		
:		
END		

5-2 キー入力

5-2-1 キー入力の概要

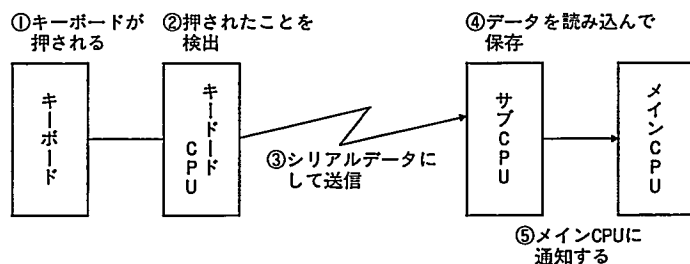


図5-2 キー・データの流れ

どのキーが押されたかというキーデータは、上の図に示すように伝達されます。キーボード用CPUは、すべてのキーを順次スキャンしており、新しく押されたキーや離されたキーを判定します。そして、キーの情報を直列のビット列にして、サブCPUに送ります。送る信号には、一般のキー入力に用いるモードA型の信号と、一度に複数のキーを読み取ることができるモードB型の信号があります。どちらの信号を使うかは、キーボード横のスイッチで選択します(X1には、モードBはありません)。サブCPUでは、送られてきた信号のパルスの幅から、モードを自動的に判別し、読み込みます。そして、その信号を基に現在のキーの状態を把握し、メインCPUからの要求があると、そのデータをメインCPUに渡します。

モードBで同時読み込みができるキーは、テンキーやスペースキーなどの24個のキーです。これを「ゲームキー」と呼びます。また、モードBでは、キーボードのカナ文字の配列が、JIS配列から五十音配列になります。

キーデータをメインCPUで受け取る方法には2通りあります。割り込み(インターラプト)を使用する方法と、使用しない方法です。割り込みを使うと、キーを押した瞬間に、データを受け取ることができるので、キー入力に対しすぐに応答しなければならない場合に適します。逆に、割り込みを使わない方法では、必要なときだけキーの情報を見に行けば良いので、不必要なキー入力を無視することができます。また、プログラムも若干簡単になります。

5-2-2 キーボードの信号

キーボードから送られて来る信号は、何種類もの幅を持ったパルス列による特殊なものです。この信号は、メインCPUから直接読むことはできないので、その概要だけ説明します。

(1) モード A 信号

キーボードからの信号は、図 5-3 のようになっています。ファンクションコード(1バイト目)は、データ(2バイト目)のデータの種類や各種シフトキーの内容を表しています。このファンクションコードの内容を、表 5-1 に示します。

モード A のキーデータ構成

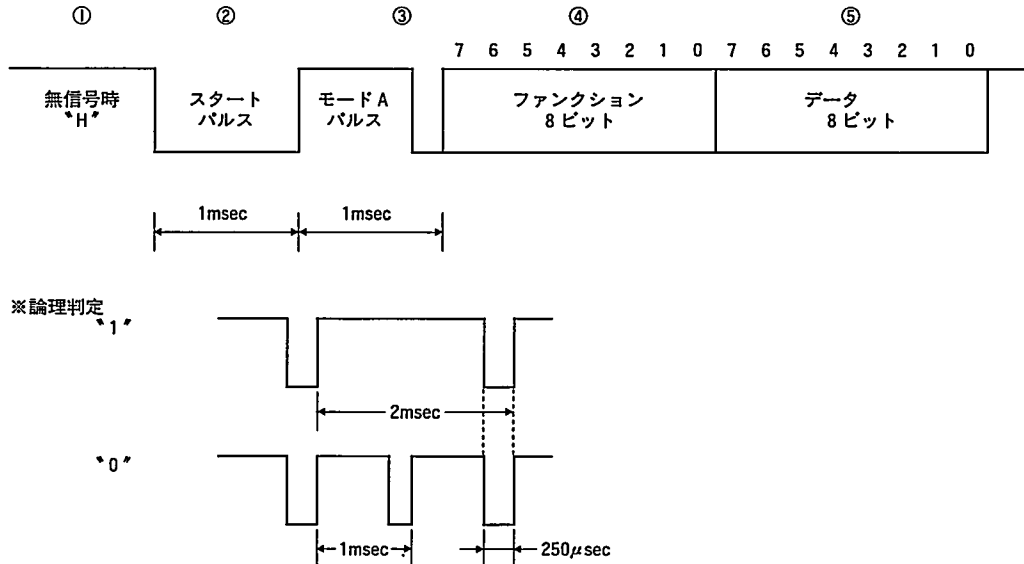


図 5-3 モード A 信号

	(MSB) 7	6	5	4	3	2	1	(LSB) 0
	ファンク ション	キーデータ 有効/無効	リピート	GRAPH	CAPS	カナ	SHIFT	CTRL
0	・テンキー ・ファンク ションキー ・TVキー ・カセット キー	・データ・ コード (8 ビット) が 有効である。	・リピート・ データであ る。	・GRAPHキ ーが押され ている。	・CAPSキー が押されて いる。 (LOCKされ ている)	・カナキー が押されて いる。 (LOCKされ ている)	・SHIFT キ ーが押され ている。	・CTRLキー が押されて いる。
1	・上記以外	・データ・ コード (8 ビット) が 無効である。	・1 回目の データであ る。	・GRAPHキ ーがはなさ れている。	・CAPSキー がはなされ ている。	・カナキー がはなされ ている。	・SHIFT キ ーがはなさ れている。	・CTRLキー がはなされ ている。

表 5-1 ファンクションコード

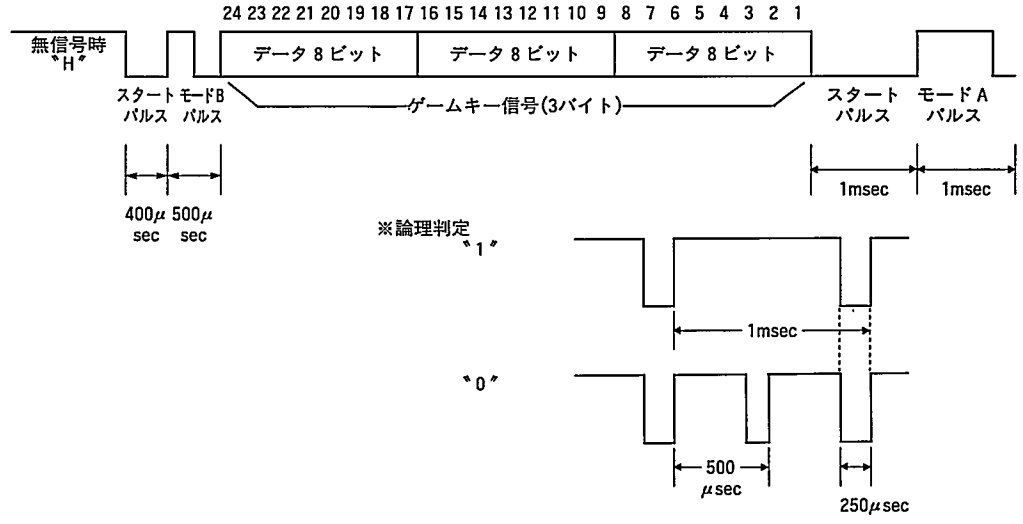
(2) モード B 信号

モード B では、モード A で送られてくるのと同じキーデータの前に、3 バイトのゲームキーデータが付くことがあります。この 3 バイト(24 ビット)のゲームキーデータは、ゲームキーのいずれかが押されるか、離されたときに送られます。このデータによって、24 種類のキーに限って同

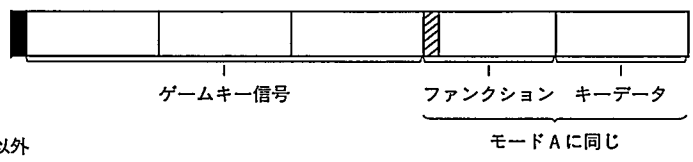
時に読み取ることができます。

図5-4にゲームキー信号の内容を示します。信号の各ビットは、24種類のそれぞれのキーに対応していて、そのキーが押されている時は、そのビットが1に、押されていないときは0になります。各ビットとキーの対応表を、表5-2に示します。

●ゲーム・キー信号の形式



●モードB信号の形式 (1)ゲームキーが押された時/離された時



(2)それ以外

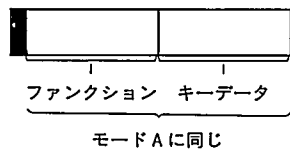


図5-4 モードB信号

ビット	1	2	3	4	5	6	7	8
キ ー	Q	W	E	A	D	Z	X	C

ビット	9	10	11	12	13	14	15	16
キ ー	7	4	1	8	2	9	6	3

テンキー

ビット	17	18	19	20	21	22	23	24
キ ー	ESC	/	-	+	*	HTAB	スペース	RET

テンキー

表5-2 ゲームキー信号の各ビットとキーの対応

5-2-3 割り込みを用いないキー入力

割り込みを用いるモードでも用いないモードでも、どちらのモードを使うかを、最初にサブCPUに宣言しておかなければなりません。このためのサブCPUのコマンドコードが、[E4H]です。

・サブCPU コマンド[E4H]

(キー入力割り込みのベクタアドレスセット)

メイン→サブ：E4, <アドレス>

サブ→メイン：なし

<アドレス>=0の時、割り込みを使わないモードになる。

割り込みを使わない場合は、0E4Hに続いて00Hを送ります。そうするとサブCPUは、それ以降キー入力があっても、割り込みを起こしません。

実際にキーデータを読み込むには、コマンド[E6H]をサブCPUに送り、続いて2バイトサブCPUから読み込みます。この2バイトは、ファンクションコード1バイトと、押されたキーのASCII(アスキー)コード1バイトです。

・サブCPU コマンド[E6H]

(キーデータの読み込み)

メイン→サブ：E6

サブ→メイン：<ファンクションコード>, <キーコード>

<ファンクションコード>の内容は、表5-1に同じ

<キーコード>はASCIIコードに変換されている

リスト5-4 キー割り込みによらないキーデータ読み込みのサンプルプログラム

OT49SB	EQU	1413H	
COMOUT	EQU	1432H	
TAK49S	EQU	143BH	
NKYIN:	LD	BC, 1A01H] BIOS ROMの状態を調べ、その状態をセーブする
	IN	A, (C)	
	AND	10H	
	JR	NZ, NKIN1	
	LD	A, 01H	
	JR	NKIN2	
NKIN1:	LD	A, 1EH	
NKIN2:	PUSH	AF	
	LD	A, 1DH] ROMアクティブ
	OUT	(00H), A	
	LD	A, 0E4H	
	CALL	COMOUT] キー割り込みを禁止する
	XOR	A	
	CALL	OT49SB	
	LD	A, 0E6H	
	LD	DE, KEYBUF] キー入力をバッファに受け取る
	CALL	TAK49S	

```

      POP      AF      ] ROMの状態を元に戻す
      OUT      (00H), A
      RET
      ;
KEYBUF: DS      2
      ;
      END

```

キーボードがモードBにセットされているときは、ゲームキーの状態を得ることができます。このときはコマンド[E3H]を使います。

・サブCPU コマンド[E3H]

(ゲームキーデータの読み込み)

メイン→サブ: E3

サブ→メイン: <ゲームキーデータ(3バイト)>

<ゲームキーデータ>の内容は、表5-2を参照

このコマンドは、キーボードがモードAでも動作しますが、返ってくるデータには意味がありません。必ずキーボードをモードBにして使用して下さい。

5-2-4 割り込みを用いたキー入力

X1シリーズでは、一般にZ-80CPUの割り込みモードをモード2にして使用します。モード2割り込みでは、割り込みが発生したときの処理ルーチンの先頭番地(エントリーアドレス)が書かれているアドレス(ベクタアドレス)を、CPUのIレジスタを上位8ビット、各I/Oデバイスの出力する値を下位8ビットとして指定します。そこでキー入力を割り込みで使うときは、割り込みベクタアドレスの下位8ビットを、あらかじめサブCPUにセットしておきます。このためのコマンドは、先ほど説明した[E4H]です。このコマンドに続いて、ベクタアドレスの下位8ビットを送ります。この値は0であってはなりません。0の場合、割り込みを使わない、という意味になってしまうからです。また、アドレスの最下位ビットは0でなければなりません。ベクタは、偶数番地から始めることになっているからです。

・サブCPU コマンド[E4H]

(キー入力割り込みのベクタアドレスセット)

メイン→サブ: E4, <アドレス>

サブ→メイン: なし

<アドレス> = 割り込みベクタアドレスの下位8ビット

最下位ビットは必ず0でなければならない

<アドレス> = 0の時、割り込みを使わないモードになる

割り込み処理ルーチンでは、5-2-3で示したのと同様、コマンド[E6H]を使ってキーデータ

を読み込みます。普通のプログラムでは、得られたキーデータをメモリ上のバッファにセットし、必要とあればキー入力フラグを立て、EI(割り込み許可)命令を実行してリターンします。Z-80では、割り込みがかかると、自動的に割り込み禁止になりますから、EIを忘れると二度と割り込みがかからなくなります。サンプルプログラムをプログラム5-5に示します。

リスト5-5 キー入力割り込みによるキーデータ読み込みのサンプルプログラム

```

IN49SB EQU      1408H
IKYIN:  PUSH     BC
        PUSH     DE
        PUSH     HL
        PUSH     AF
        LD       BC, 1A01H
        IN       A, (C)
        AND      10H
        JR       NZ, IKIN1
        LD       A, 1DH
        JR       IKIN2
IKIN1:  LD       A, 1EH
IKIN2:  PUSH     AF
        LD       A, 1DH
        OUT      (00H), A
        CALL     IN49SB
        LD       L, A
        CALL     IN49SB
        LD       H, A
        LD       (KEYBUF), HL
        POP      AF
        OUT      (00H), A
        POP      AF
        POP      HL
        POP      DE
        POP      BC
        EI
        RETI
;
KEYBUF: DS       2
;
END

```

BIOS ROMの状態をセーブ

ROMアクティブ

80C49よりキーデータを受け取り、バッファにセーブする

割り込み処理ルーチン内でゲームキーを読み込むときは、まず[E6H]のキー入力コマンドを実行し、キー入力データを受け取ってから[E3H]のゲームキー入力を行って下さい。

リスト5-6 キー割り込み中のゲームキー読み込み

```

IN49SB EQU      1408H
TAK49S EQU      143BH
GKYIN:  PUSH     BC
        PUSH     DE
        PUSH     HL
        PUSH     AF
        LD       BC, 1A01H
        IN       A, (C)
        AND      010H
        JR       NZ, GKIN1
        LD       A, 01DH
        JR       GKIN2
GKIN1:  LD       A, 1EH
GKIN2:  PUSH     AF

```

ROMの状態をセーブ

```

LD      A, 1DH      ] ROMアクティブ
OUT     (00H), A    ]
CALL    IN49SB      ] キーファンクションとASCIIコードを読み込む
CALL    IN49SB      ]
LD      A, 0E3H      ]
LD      DE, GKBUF    ] ゲームキーデータをバッファにセットする
CALL    TAK49S      ]
POP     AF           ] ROMを元の状態に戻す
OUT     (00H), A    ]
POP     AF
POP     HL
POP     DE
POP     BC
EI
RETI
;
GKBUF:  DS          3
;
END

```

5-3 専用モニター TV のコントロール

5-3-1 モードの切り換え

X1シリーズの専用モニターには次の4つのモードがあり、X1本体からのリモコン信号によって切り換えることができます。

- 1) テレビ画面のみ
- 2) コンピュータ画面のみ
- 3) スーパーインポーズ1 (テレビとコンピュータ画面を重ねて表示. テレビのコントラストをダウンさせる)
- 4) スーパーインポーズ2 (テレビとコンピュータ画面を重ねて表示. テレビのコントラストはダウンさせない)

これらのモード切り換えも、サブCPUにコマンドを送ることによって行うことができます。このためのコマンドが[E7H]です。

・サブCPU コマンド[E7H]

(専用モニターのコントロール)

メイン→サブ: E7, <コード>

サブ→メイン: なし

モード切り換えのためのコードには、X1シリーズ共通の1～5バイトのものと、X1turbo以降追加された1バイトのものがあります。これらのコードを表5-3に示します。1～5バイトのコードでは、最初に0E7H, 05Hを送って、テレビ画面に切り換えてからモード切り換えをするので、一瞬テレビ画面が写ります。

(a) X1(1~5バイト)

画面モード \ 送信コード(バイト数)	1	2	3	4	5	6
T V 画面	E7	05				
コンピュータ画面	E7	05	E7	08		
スーパーインポーズ1 (コントラストダウン)	E7	05	E7	0F	E7	0A
スーパーインポーズ2 (コントラストノーマル)	E7	05	E7	0F		

コードはすべて16進数

(b) X1turbo以降(1バイト)

画面モード \ 送信コード(バイト数)	1	2
T V 画面	E7	1C
コンピュータ画面	E7	1D
スーパーインポーズ1 (コントラストダウン)	E7	1E
スーパーインポーズ2 (コントラストノーマル)	E7	1F

コードはすべて16進数

表5-3 モニターのモード切り換えコード

リスト5-7 モニター画面をコンピュータ画面にするためのアクセス手順

OT49SB EQU	1413H	
COMOUT EQU	1432H	
CMDSP: LD	BC, 1A01H	ROMの状態をセーブ
IN	A, (C)	
AND	10H	
JR	NZ, CMDP1	
LD	A, 1DH	
JR	CMDP2	
CMDP1: LD	A, 1EH	ROMをアクティブに
CMDP2: PUSH	AF	
LD	A, 1DH	
OUT	(00H), A	
LD	A, E7H	80C49にコマンドとデータを送る。画面モードをコンピュータ画面に変更する
CALL	COMOUT	
LD	A, 1DH	
CALL	OT49SB	
POP	AF	ROMを元の状態に戻す
OUT	(00H), A	
RET		
; END		

5-3-2 コントロール

専用モニターでは、モード切り換えの他、チャンネル、音量などのコントロールがコンピュータ側から可能です。この制御にも、コマンド[E7H]を使います。このコマンドの後に、表5-4に示すコードを送ります。

内 容	送信コード(バイト数)	
	1	2
ボ リ ュ ー ム ア ッ プ	E7	01
ボ リ ュ ー ム ダ ウ ン	E7	02
ボリュームノーマル(42/64階調)	E7	03
音 声 ミ ュ ー ト	E7	06
チャ ン ネ ル ア ッ プ	E7	0B
チャ ン ネ ル ダ ウ ン	E7	0C
パ ワ ー オ フ	E7	0D
パワーオン/オフ(トグル動作)	E7	0E
チャ ン ネ ル 1	E7	10
チャ ン ネ ル 12		1B
パ ワ ー オ ン	E7	80

コードはすべて16進数

表5-4 モニターのコントロールコード

リスト5-8 モニター画面をスーパーインポーズ1モードにするためにアクセス手順

```

OT49SB EQU      1413H
COMOUT EQU      1432H
-11DSP: LD       BC, 1A01H
        IN       A, (C)
        AND     010H
        JR      NZ, 11DP2
        LD      A, 1DH
        JR      11DP2
11DP1: LD      A, 1EH
11DP2: PUSH    AF
        LD      A, 1DH
        OUT     (00H), A
        LD      A, E7H
        CALL    COMOUT
        LD      A, 1EH
        CALL    OT49SB
        POP     AF
        OUT     (00H), A
        RET
        ;
        END

```

ROMの状態をセーブ

ROMをアクティブに

画面モードをスーパーインポーズ1に変更

ROMを元の状態に戻す

5-4 専用カセットデッキのコントロール

5-4-1 カセットメカのコントロール

X1 シリーズの専用カセット (内蔵のものを含む) は、サブ CPU によってコントロールされています。メカのコントロールの他、カセットの有無、テープエンド、消去防止ツメなどのチェックもしています。カセットのコントロールをメイン CPU から行うには、コマンド[E9H]を使います。なお、サブ CPU が行うのは、メカニズム関係のみで、実際の信号の読み書きはメイン CPU から直接 I/O ポートを通じて行います。8255②の項(5-6-3)を参照して下さい。

・サブ CPU コマンド[E9H]

(カセットメカのコントロール)

メイン→サブ：E9, <コントロールコード>

サブ→メイン：なし

コントロールコードは1バイトです。その内容を表5-5に示します。ここで APSS とは頭出しのための状態で、ヘッドをテープにつけたまま早送りや巻き戻しを行います。このときの APSS 読み取り信号は、8255①(サブ CPU 側)に接続されています。

動 作	コントロールコード
EJECT	00
STOP	01
PLAY	02
FF	03
REW	04
APSS-FF	05
APSS-REW	06
REC	0A

表5-5 カセット・コントロール・コード

5-4-2 ステータスの読み出し

専用カセットを使用しているときは、サブ CPU から現在のカセットの状態を知ることができます。このためのコマンドとして、メカの状態を知る[EAH]とカセットの状態を知る[EBH]があります。

・サブCPU コマンド[EAH]

(カセットメカニズムの状態検出)

メイン→サブ: EA

サブ→メイン: <状態コード>

<状態コード>の見方は、表 5-5 を参照

・サブCPU コマンド[EBH]

(カセットの状態検出)

メイン→サブ: EB

サブ→メイン: <カセット状態コード>

<カセット状態コード>の見方は、図 5-5 を参照

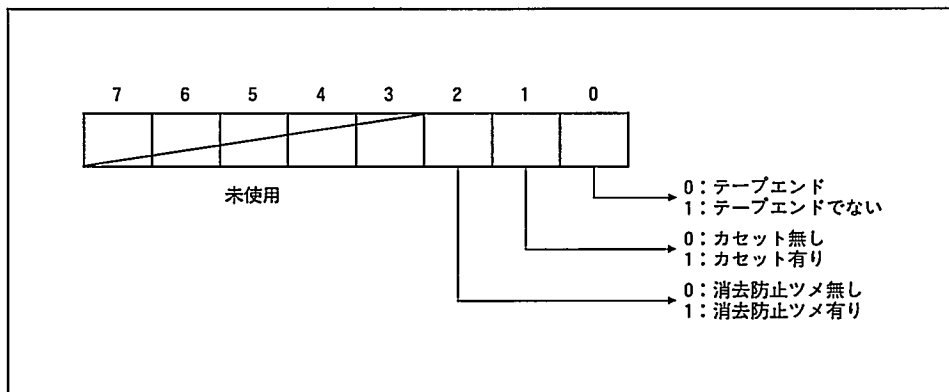


図5-5 カセット状態コード

カセットメカが動作中(PLAYやRECの状態など)に BREAK キー等が押されると、サブCPU は次のような動作を行います。

(1) BREAK キーが押された時

- ・カセットを STOP する
- ・メイン CPU の BREAK フラグ(8255② PB-0)を L にする
- ・キー入力割り込みをメイン CPU にかける(割り込みが許可されている時)
- ・メインからのキー入力に対しては、キーコード 03H を返す

(2) カセットコントロールキーが押された時

- ・押されたキーに対応するカセットの動作を行う
- ・メイン CPU の BREAK フラグ(8255② PB-0)を L にする

リスト5-10 カセットメカのコントロール

```

OT49BS EQU      1413H
COMOUT EQU      1432H
TCCTR:  LD      BC, 1A01H
        IN      A, (C)
        AND     10H
        JR      NZ, TCCT1
        LD      BC, 1D00H
        JR      TCCT2
TCCT1:  LD      BC, 1E00H
TCCT2:  LD      A, 1DH
        OUT     (00H), A
        LD      A, 0E9H
        CALL    COMOUT
        LD      A, (01H)
        CALL    OT49SB
        OUT     (C), A .....ROMを元に戻す
        RET
        ;
        END

```

ROMの状態をセーブ

ROMをアクティブ

カセットストップコード送信

リスト5-11 カセットメカの状態の読み出し

```

IN49SB EQU      1408H
COMOUT EQU      1432H
RCCTR:  LD      BC, 1A01H
        IN      A, (C)
        AND     10H
        JR      NZ, RCCT1
        LD      BC, 1D00H
        JR      RCCT2
RCCT1:  LD      BC, 1E00H
RCCT2:  LD      A, 1DH
        OUT     (00H), A
        LD      A, 0EAH
        CALL    COMOUT
        CALL    IN49SB
        LD      (RCCBF), A
        OUT     (C), A .....ROMを元に戻す
        RET
        ;
RCCBF:  DS      1
        ;
        END

```

ROMの状態をセーブ

ROMをアクティブに

カセット状態読み出し

5-4-3 カセットテープのフォーマット

カセットに記録されているデータの読み書きは、メインCPUがI/Oポートを通して直接行います。カセット関係のI/Oポートは、以下の通りです。

・8255②ポートB(入力)

I/Oアドレス=1A01H

ビット1: READ DATA

カセットからの読み込み信号

・8255②ポートC(出力)

I/Oアドレス=1A02H

ビット0: WRITE DATA

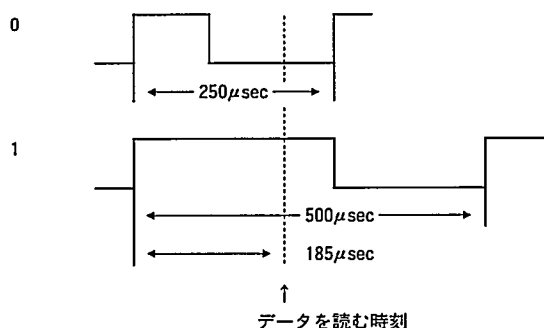
カセットへの書き込み信号

カセット書き込み用のポートC出力は、他のコントロール出力と共通ですから、カセット出力に必要なビット0以外は変更しないように注意しなければなりません。このためには8255のビットセット・リセット機能を使用します。

さて、X1シリーズの標準フォーマットでは、信号にシャープPWM方式という変調方式が使用されています。PWM(Pulse Width Modulation:パルス幅変調)では、パルスの幅を変化させて情報を記録します。パルスの1サイクルが $250\mu\text{s}$ なら0、 $500\mu\text{s}$ なら1となっています。読み込むときは、波形がHになってから $185\mu\text{s}$ 後の状態を調べ、このときHなら1、Lなら0と判断します。

読み書きは1バイトを単位として行われます。パルスの形式と、1バイトの構成を図5-6に示します。

● 1 ビットの信号



● 1 バイトの構成

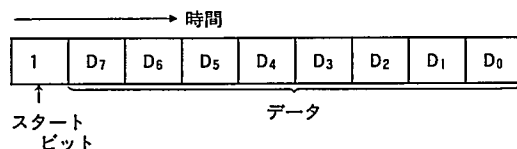
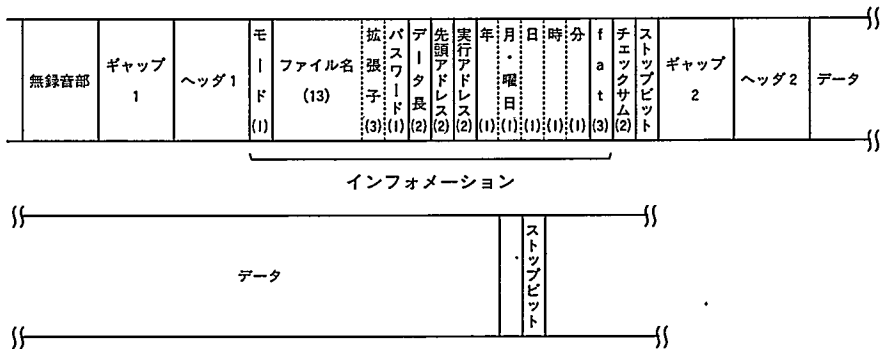


図5-6 シャープPWM方式

X1シリーズのテープフォーマット(論理フォーマット)は、インフォメーションブロックとデータブロックの2つに大きく分けられます。各ブロックの最後にはデータの総和を取った2バイトのチェックサムがつきます。全体では、まず8秒間の無録音部分、インフォメーションブロック、そしてデータブロックと続きます。また、データブロックについては、データが連続してベタで記録される「連続セーブ」と256バイトずつのブロック毎に区切って記録する「ブロッキングセーブ」があります。

BASICでは、プログラムのセーブに連続セーブ、データファイルやASCIIセーブにはブロッキングセーブを使っています。フォーマットの詳細を図5-7に示します。

(1) 連続セーブの場合



(2) ブロッキングセーブの場合

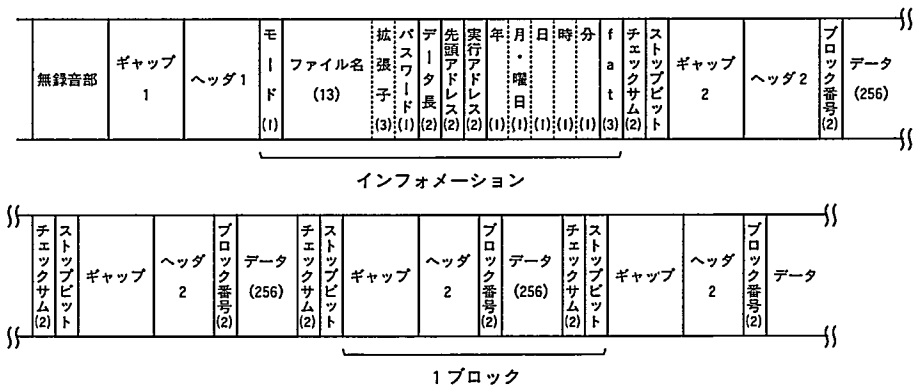


図5-7 テープの論理フォーマット

5-5 タイマーのコントロール

5-5-1 時刻の設定と読み出し

サブCPUには、タイマーIC(μPD1990)が接続されており、日付や時刻の読み出し、書き込みをメインCPUから行うことができます。タイマーICは、ニッカド(充電式)電池でバックアップされており、本体の電源が切れても時を刻み続けます。ただし、「年」の情報だけはサブCPUがカウントしているため、本体背面のメインスイッチを切ると「年」情報は無くなってしまいます。

日付、時刻の読み出し、書き込みのサブCPUコマンドは次の通りです。

・サブCPUコマンド[ECH]

(日付の設定)

メイン→サブ: EC, <日付データ(3バイト)>

サブ→メイン: なし

・サブCPU コマンド[EDH]

(日付データの読み出し)

メイン→サブ: ED

サブ→メイン: <日付データ(3バイト)>

・サブCPU コマンド[EEH]

(時刻の設定)

メイン→サブ: EE, <時刻データ(3バイト)>

サブ→メイン: なし

・サブCPU コマンド[EFH]

(時刻データの読み出し)

メイン→サブ: EF

サブ→メイン: <時刻データ(3バイト)>

日付データ, 時刻データはどちらも3バイトのブロックになっています。これらのデータの構成を次頁に示します。

リスト5-12 カレンダー時計から日付と日時を読み出してメモリーに書き込む

```

TAK49S EQU      143BH
RCCLD:  LD       BC, 1A01H
        AND      10H
        JR       NZ, RCCL1
        LD       BC, 1D00H
        JR       RCCL2
RCCL1:  LD       BC, 1E00H
RCCL2:  PUSH     BC
        LD       A, 1DH
        OUT      (00H), A
        LD       DE, CLBUF
        LD       A, 0EDH .....日付読み出しコマンド
RCCL3:  PUSH     AF
        CALL     TAK49S
        INC      DE
        INC      DE
        INC      DE
        POP      AF
        ADD      A, 02H .....=EF 時刻読み出しコマンド
        JR       NC, RCCL3
        POP      BC
        OUT      (C), A
        RET
        ;
CLBUF:  DS       6
        ;
        END
    
```

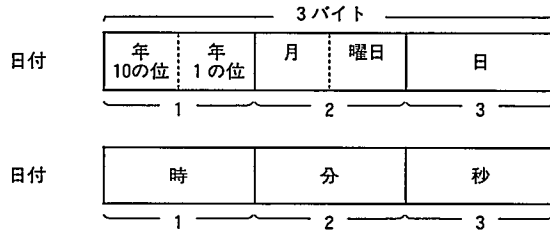
ROMの状態をセーブ

ROMをアクティブに

データをバッファにセーブ

ROMを元の状態に戻す

●データの形式



項 目	デ ー タ 説 明																		
年	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <div>10の位1の位</div>																		
(注1) 00年～99年の値をそのまま指定して下さい。																			
月， 曜日	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td></td><td></td><td></td><td></td><td>*</td><td></td><td></td><td></td><td></td></tr></table> <div>月曜日</div>													*					*印は無効ビット
					*														
(注2) 月：(1)H(1月)～(C)H(12月)までの値を指定して下さい。(0)Hを指定すると無効(XX)となります。 曜日：(0)H(日)～(6)H(土)までの値を指定して下さい。																			
日	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <div>10の位1の位</div>																		
(注3) 10の位：(0)～(3)Hまでの値を指定して下さい。 1の位：(1)～(9)Hまでの値を指定して下さい。																			
時	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <div>10の位1の位</div>																		
10の位：(0)～(2)Hまでの値を指定して下さい。 1の位：(0)～(9)Hまでの値を指定して下さい。																			
分	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <div>10の位1の位</div>									*									*印は無効ビット
	*																		
(注4) 10の位：(0)～(5)Hまでの値を指定して下さい。 1の位：(1)～(9)Hまでの値を指定して下さい。																			
秒	ビット	7	6	5	4	3	2	1	0										
	<table border="1"><tr><td>*</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> <div>10の位1の位</div>									*									*印は無効ビット
	*																		

図5-8 日付、時刻データの構成

5-5-2 テレビタイマーの設定と読み出し

X1シリーズには、最大7回まで設定できるテレビタイマーコントロール機能があります。これらはサブCPUが行っているため、本体のフロント電源がoffでも動作します。

テレビタイマーに関するサブCPUコマンドコードは、[D1H]～[D7H](設定)と[D9H]～[DFH](読み出し)の14個で、コードによって7つのタイマの選択を行います。

・サブCPU コマンド[D1H]～[D7H]

(テレビタイマーの設定)

メイン→サブ:[コマンド], <タイマーデータ(6バイト)>

サブ→メイン:なし

・サブCPU コマンド[D9H]～[DFH]

(テレビタイマーの設定状態の読み出し)

メイン→サブ:[コマンド]

サブ→メイン:<タイマーデータ(6バイト)>

コマンドの一覧表を表5-6に、タイマーデータの構成を図5-9と表5-7に示します。

TIMER番号	設定コード	読出しコード
1	D1	D9
2	D2	DA
3	D3	DB
4	D4	DC
5	D5	DD
6	D6	DE
7	D7	DF

コードはすべて16進数

図5-6 タイマー番号と送信コードの対応

6 バイト					
コントロール対象 及びインターバル	コントロール内容	分	時	月曜日	日
1	2	3	4	5	6

図5-9 タイマーデータの構成

項 目	デ ー タ 説 明																																									
コントロール 対 象 及 び インター バル	ビット 7 6 5 4 3 2 1 0 <table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td colspan="2">タイマー有無</td><td colspan="6">インターバル(1～59分)</td></tr></table>									タイマー有無		インターバル(1～59分)																														
	タイマー有無		インターバル(1～59分)																																							
	<table border="1"><tr><th colspan="2">ビット内容</th><th rowspan="2">機 能</th></tr><tr><th>7</th><th>6</th></tr><tr><td>0</td><td>0</td><td>タイマー無効</td></tr><tr><td>1</td><td>0</td><td rowspan="2">未 使 用</td></tr><tr><td>1</td><td>1</td></tr><tr><td>0</td><td>1</td><td>タイマー有効</td></tr></table>	ビット内容		機 能	7	6	0	0	タイマー無効	1	0	未 使 用	1	1	0	1	タイマー有効																									
ビット内容		機 能																																								
7	6																																									
0	0	タイマー無効																																								
1	0	未 使 用																																								
1	1																																									
0	1	タイマー有効																																								
	※インターバルタイマーとは、あるタイマーが動作してから指定された時間間隔 (1～59分)が経過したら、再び同じ動作を行うタイマーです。																																									
コントロール 内 容	7 6 5 4 3 2 1 0 <table border="1"><tr><td></td><td>0</td><td>0</td><td></td><td></td><td></td><td></td><td></td></tr><tr><td colspan="2">↓ システムビット</td><td colspan="6">TVコントロール内容コード</td></tr></table> <table border="1"><tr><td>0</td><td>TV ONのコードは出力しない。</td></tr><tr><td>1</td><td>TVパワーONしたのち、0～4ビットのコードをTVへ送ります。</td></tr></table>		0	0						↓ システムビット		TVコントロール内容コード						0	TV ONのコードは出力しない。	1	TVパワーONしたのち、0～4ビットのコードをTVへ送ります。																					
		0	0																																							
	↓ システムビット		TVコントロール内容コード																																							
	0	TV ONのコードは出力しない。																																								
	1	TVパワーONしたのち、0～4ビットのコードをTVへ送ります。																																								
		全コントロール内容とその設定データコードを下表に示します。 希望のコードを設定して下さい。(00)Hの場合、タイマー動作をしません。																																								
		<table border="1"><thead><tr><th>コントロール内容</th><th>データコード(Hex)</th></tr></thead><tbody><tr><td>ボリュームアップ</td><td>01</td></tr><tr><td>ボリュームダウン</td><td>02</td></tr><tr><td>ボリュームノーマル</td><td>03</td></tr><tr><td>音声ミュート</td><td>06</td></tr><tr><td>チャンネルアップ</td><td>0B</td></tr><tr><td>チャンネルダウン</td><td>0C</td></tr><tr><td>パワーオフ</td><td>0D</td></tr><tr><td>パワーオン/オフ (トグル動作)</td><td>0E</td></tr><tr><td>チャンネル 1</td><td>10</td></tr><tr><td>チャンネル 12</td><td>1B</td></tr><tr><td>パワーオン</td><td>80</td></tr><tr><td>パワーオン→ボリュームアップ</td><td>81</td></tr><tr><td>パワーオン→ボリュームダウン</td><td>82</td></tr><tr><td>パワーオン→ボリュームノーマル</td><td>83</td></tr><tr><td>パワーオン→音声ミュート</td><td>86</td></tr><tr><td>パワーオン→チャンネルアップ</td><td>8B</td></tr><tr><td>パワーオン→チャンネルダウン</td><td>8D</td></tr><tr><td>パワーオン→チャンネル 1</td><td>90</td></tr><tr><td>パワーオン→チャンネル 12</td><td>9B</td></tr></tbody></table>	コントロール内容	データコード(Hex)	ボリュームアップ	01	ボリュームダウン	02	ボリュームノーマル	03	音声ミュート	06	チャンネルアップ	0B	チャンネルダウン	0C	パワーオフ	0D	パワーオン/オフ (トグル動作)	0E	チャンネル 1	10	チャンネル 12	1B	パワーオン	80	パワーオン→ボリュームアップ	81	パワーオン→ボリュームダウン	82	パワーオン→ボリュームノーマル	83	パワーオン→音声ミュート	86	パワーオン→チャンネルアップ	8B	パワーオン→チャンネルダウン	8D	パワーオン→チャンネル 1	90	パワーオン→チャンネル 12	9B
	コントロール内容	データコード(Hex)																																								
	ボリュームアップ	01																																								
	ボリュームダウン	02																																								
ボリュームノーマル	03																																									
音声ミュート	06																																									
チャンネルアップ	0B																																									
チャンネルダウン	0C																																									
パワーオフ	0D																																									
パワーオン/オフ (トグル動作)	0E																																									
チャンネル 1	10																																									
チャンネル 12	1B																																									
パワーオン	80																																									
パワーオン→ボリュームアップ	81																																									
パワーオン→ボリュームダウン	82																																									
パワーオン→ボリュームノーマル	83																																									
パワーオン→音声ミュート	86																																									
パワーオン→チャンネルアップ	8B																																									
パワーオン→チャンネルダウン	8D																																									
パワーオン→チャンネル 1	90																																									
パワーオン→チャンネル 12	9B																																									

分	ビット	7	6	5	4	3	2	1	0	* 印は無効ビット
		*								
時	ビット	7	6	5	4	3	2	1	0	
月, 曜日	ビット	7	6	5	4	3	2	1	0	* 印は無効ビット
						*				
日	ビット	7	6	5	4	3	2	1	0	

(注1) 設定値として(00)H～(7F)Hまで取り得ますが、(00)H～(59)H(0分～59分)以外の値を指定した場合、そのタイマー動作について保障できません。この項を無効にはできません。

(注2) 設定値として(00)H～(FF)Hまで取り得ますが、(FF)Hを書き込んだ場合のみ、自動的に時間の項が無効(XX)となります。また25H(25時)以上の値を指定した場合、そのタイマー動作について保障できません。

(注3) 月: 13((D)H)未満のBCD値を指定する。(0)Hのときは無効(XX)となります。
曜日: 全ビットが“0”になる(7)H又は(F)Hを指定すると、自動的に曜日の項が無効(XXX)になります。

設定値 (H)	曜 日
0 or 8	SUN
1 or 9	MON
2 or A	TUE
3 or B	WED
4 or C	THU
5 or D	FRI
6 or E	SAT
7 or F	XXX

(注4) BCDの値を指定します。(32)H以上の値を書き込めますが、そのときのタイマー動作は保障できません。(00)Hの場合、日の項は無効(XX)となります。

表5-7 タイマーコントロールバイト(6バイト)の各ビットの内容

リスト5-13 テレビタイマーの設定例

```
TAK49S EQU 143BH
STTMR: LD BC, 1A01H
      AND 10H
      JR NZ, STTM1
      LD BC, 1D00H
      JR STTM2
STTM1: LD BC, 1E00H
STTM2: PUSH BC
      LD A, 1DH
      OUT (00H), A
```

BIOS ROMの状態をセーブ

ROMアクティブ

```

LD      DE, TMBUF
LD      A, 0D1H
CALL    TAK49S      ] 80C49へタイマーセットコマンドを送信
POP     BC           ] ROMを元の状態に戻す
OUT     (C), A
RET
;
TMBUF:  DB      1 .....タイマー番号
        DB      40H, 96H
        DB      17H, 00H, 7DH, 26H
        ;
END

```

5-5-3 タイマー用 IC

X1シリーズに使われているμPD1990は、カレンダ機能、クロック機能を持ったICです。サブCPUとは、40ビットのシリアルデータの形でデータの入出力を行います。このICは、充電式電池によってバックアップされており、本体の電源を切っても動作します。電池の充電は、サブCPUの電源系統を使って行われており、従ってフロントの電源スイッチがoffであっても、背面のメインスイッチが入っていれば、電池の充電が行われます。

このICは、データの読み書きをしていない時、DATA OUT 端子から1Hzの信号を出力しています。この信号は、画面のブリンク用に使われています。

5-6 PPI(8255)

X1シリーズでは、メインCPUとサブCPUに1つずつ、PPIと呼ばれるI/Oデバイスが接続されており、メイン↔サブCPU間の通信や、各種のI/O制御に使われています。サブCPUに接続された方のPPIを8255①、メインCPUに接続された方を8255②と呼びます。

5-6-1 PPIの概要

8255PPIは、インテル社が開発し、世界で広く使われているパラレル入出力用のLSIです。このLSIには24本の入出力端子(ポート)があり、これらは8本ずつ3組に分けてポートA、ポートB、ポートCと呼ばれます。各ポートは、入力にするか出力にするか、データ転送にハンドシェイク法を使うかどうか、などがCPUからのプログラムによって決められるようになっています。また、データが入力された時に割り込みをかけたり、ポートCの出力を1ビットずつセット、リセットする機能もあります。図5-10に、8255②の機能設定などに使われるコントロールレジスタの内容を示します。なお、8255①はサブCPUの管理下あるのでメインCPU側から操作することはできません。

8255の初期設定は、X1シリーズの場合、リセット時にIPLによって行われています。初期設定は、ハードウェアと密接な関係があり、設定を勝手に変更したりしてはいけません。

さて、8255には大きく分けて3つのモードがあります。

・モード0

単純な入力か出力しかしないモードです。8ビットの入出力ポート2個(ポートA、ポートB)と、4ビットの入出力ポート2個(ポートCの上位4ビット、下位4ビット)をそれぞれ、入力にするか出力にするか決めることができます。

・モード1

8ビットのデータを、3本のコントロールラインを使ったハンドシェイクと呼ぶ方法で転送します。コントロールラインにはポートCの一部が使われます。

・モード2

モード2はポートAだけが指定できます。このモードを指定するとポートAは、8ビットの入出力兼用線となります。ポートCの内5本がコントロールラインとなり、ハンドシェイクによるデータ転送を制御します。サブCPUとのコマンド送受信の項で、IBF(Input Buffer Full)、OBF(Output Buffer Full)というフラグが出てきましたが、実はこの5本のコントロールラインの一部なのです。

X1シリーズでは、サブCPUの管理下にある8255①のポートAがモード2に設定され、メインCPUとの通信に使われている他は、すべてモード0の単純入出力になっています。入力、出力の設定の様子を表5-9に示します。先に述べたように、これらの設定をプログラムで勝手に変更してはいけません。従ってコントロールワードのうちポートCのビットセット、リセット機能だけが使用できます。

グループ	ポート端子	アクティブ	コントロール内容	信号名
A	PA7	—	データ入出力	1D7
	PA6	—		1D6
	PA5	—		1D5
	PA4	—		1D4
	PA3	—		1D3
	PA2	—		1D2
	PA1	—		1D1
	PA0	—		1D0
	PC7	L	Z-80Aに対してデータ受取り	OBF
	PC6	L	Z-80AがポートAからデータを受取り信号	8049RD
	PC5	H	Z-80Aに対してデータ転送禁止信号	1BF
	PC4	L	Z-80AからのデータをポートAに入力/ラッチ指示信号	8255WE
B	PC3	—	未使用	
	PC2	H	PLAY時READ LED点灯します(L: WRITE LED)	
	PC1	L	Z-80AへのBREAK信号	BREAK
	PC0	L	カセットのEJECTソレノイドコントロール	EJECT SOL
	PB7	—	OBF信号	
	PB6	—	8049RD信号	
	PB5	—	APSS(無記録部検出)	APSS DATA(注)
	PB4	L	EJECT SW. センス	EJECT SW
	PB3	—	未使用	
	PB2	H	カセットテープの書き込み禁止用の爪がある状態	REC PROTECT
	PB1	H	カセットがセットされている状態	PACKAGE
	PB0	L	テープエンド検出	TAPE END

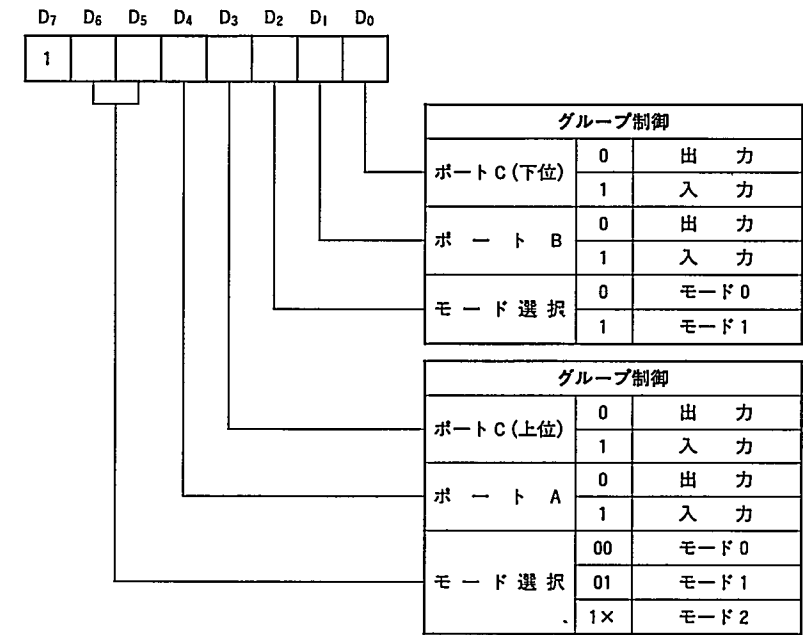
(注) READ DATA信号を、積分回路を通すことにより得た信号。

表5-8 X1 turbo model 10 における8255①のポートの割り当て

	CPU	ポート	モード	設 定
8255①	サ ブ	A B C	2 0 —	入出力 入 力 入出力
8255②	メイン	A B C	0 0 —	出 力 入 力 出 力

表5-9 X1シリーズにおける8255の設定

●モード設定



●ビット・セット/リセット

コントロールワード

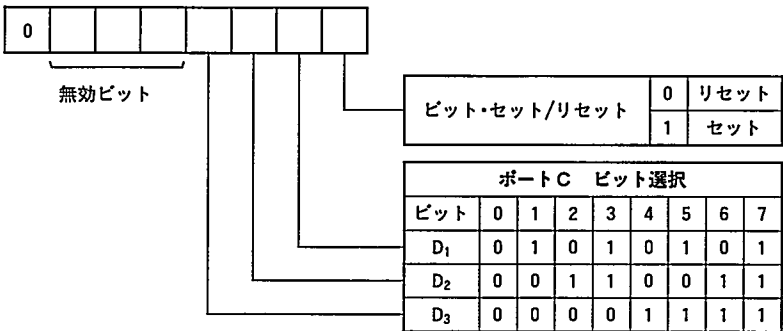


図5-10 8255のコマンドレジスタ