

3-9. クロック源選択

outp (BASE + 4, cks); /* cks : クロック源選択 */

ここで選択されたクロック源信号は次項で説明するプログラマブルカウンタにより分周されて目的のサンプリング（入力タイミング）クロックとなります。

表 3 - 9 A . 【BASE + 4】出力ポートの構成

ビット	各ビットの機能・意味	= 1 のとき	= 0 のとき	リセット時
B 7	外部クロック源（使用時）の有効極性指定	(+)	(-)	0
B 6	内部クロック源（使用時）の選択データ	【表 3 - 9 B】参照。		0
B 5				0
B 4	クロック源選択（外部 / 内部）	外部	内部	0
B 3	未使用			
B 2				
B 1				
B 0				

表 3 - 9 B . 選択される内部クロック源

ビット		選択される内部クロック源	
B 6	B 5	クロック源周波数	主な用途
1	1	無効	
1	0	オプション	
0	1	無効	
0	0	10.000 MHz	

オプションのクロック源素子（10 MHz 以下）は本ボード上のソケットに装着することにより使用可能となります。

素子名： JXO - 5 S - MHz（金石），または
DOC - 49 S 1 - MHz（大真空）

3-10. 読み書き対象カウンタの選択

```
outp(BASE + 7, ctr); /* ctr: 操作対象カウンタ */
```

この後に読み書きするカウンタ（またはカウンタの制御レジスタ）を選択します。
 カウンタ素子（ $\mu\text{PD71054} = \text{i8253A}$ 上位互換）は2個あり、各々16ビット×3本構成ですが、各素子共、このうち16ビット×2本を利用（残りの1本は不使用）しています。
 素子1はクロック源分周用に、素子2はFIFOバッファ入力時のデータ点数（語数）カウンタに使用します。

表3 - 10 A . 【BASE + 7】出力ポートの構成

ビット	各ビットの機能・意味	ビット時
B 7 B 6 B 5 B 4 B 3	未使用	
B 2 B 1 B 0	カウンタ（または制御レジスタ）選択データ【表3 - 10 B参照】	0 0 0

表3 - 10 B . カウンタ（または制御レジスタ）選択データ

B 2	B 1	B 0	選択されるカウンタ、または制御レジスタ
1	1	1	[素子 2] 制御レジスタ：1バイト
1	1	0	["] カウンタ# 2：不使用
1	0	1	["] カウンタ# 1：サンプリング点数カウンタ・上位ワード（2バイト）
1	0	0	["] カウンタ# 0：サンプリング点数カウンタ・下位ワード（2バイト）
0	1	1	[素子 1] 制御レジスタ：1バイト
0	1	0	["] カウンタ# 2：不使用
0	0	1	["] カウンタ# 1：クロック源分周・上位ワード（2バイト）
0	0	0	["] カウンタ# 0：クロック源分周・下位ワード（2バイト）

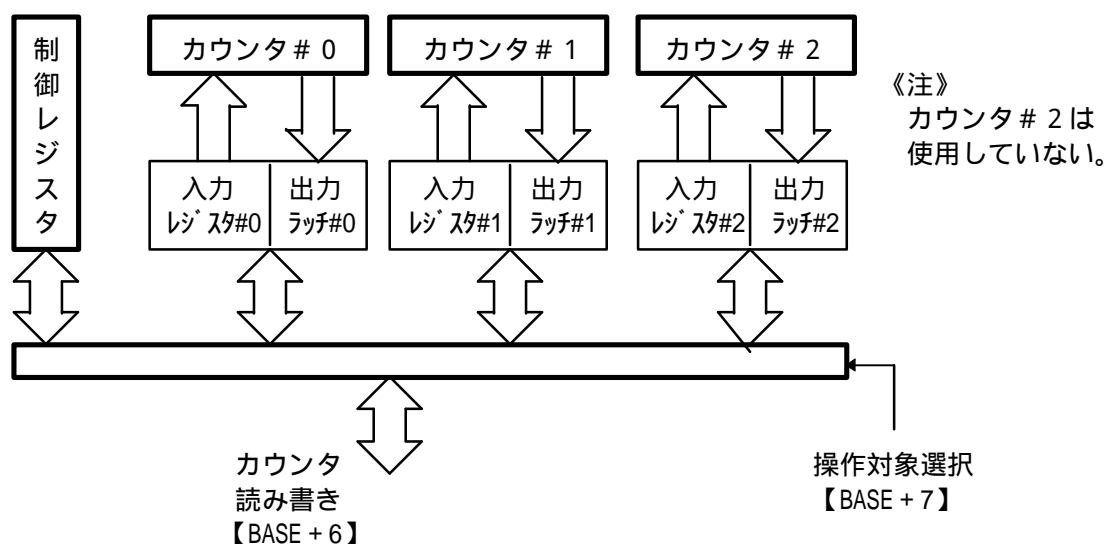
【注】 各カウンタ、制御レジスタの機能については次3 - 11項、3 - 12項を参照。

カウンタ素子の構造

各カウンタ素子（ μ PD71054 = i8253A上位互換）の構造を図3 - 10に記します。
 3本の各カウンタ（16BITバイナリ・ダウンカウンタ）は制御レジスタ（1バイト）により動作モードを指定されます。なお書き込まれるデータは実際には各カウンタ専用のレジスタに書き込み・保持されており、計数開始時等（動作モードに応じて）、必要なタイミングでカウンタ本体にロード（転送）されます。また、データをパソコン側から読み込むときは全カウンタを同時にラッチする構造となっています。

【3 - 11項・分周比（クロック値）の設定】および【3 - 12項・データ点数の設定】では当素子の特定動作モードのみを使用します。素子自体の詳細については素子メーカー発行の仕様書等を御参照ください。（本ボードの制御には不要です。）

図3 - 10 . μ PD71054（i8253A上位互換）の構造



3-11. クロック値（クロック源分周比）の設定

F I F Oバッファ入力モードで使用するクロック値は、前3 - 9項で選択したクロック源を分周して得ます。分周は32BITカウンタ（16BITバイナリカウンタを2本直列接続）で行います。

各16BITカウンタに設定できる値は1～65535で、その設定値を【DV#0】、【DV#1】とすると、

$$\text{分周比【DV】} = \text{【DV\#0】} \times \text{【DV\#1】} \quad \text{..... (a)}$$

$$\text{得られるクロック周波数【CKF】} = \text{【クロック源周波数】} \div \text{【DV】} \quad \text{..... (b)}$$

$$\text{得られるクロック周期【CKP】} = \text{【クロック源周期】} \times \text{【DV】} \quad \text{..... (c)}$$

となります。

《例》 内部クロック源10MHz（周期=100ns）を元に10μsのクロック値（周期）を得るには、

$$\text{式(c)より、} \quad \text{【DV】} = (10\mu s) \div (100ns) = 100$$

$$\begin{aligned} \text{式(a)から} \quad & \text{【DV\#0】} = 100 \quad \& \quad \text{【DV\#1】} = 1、 \\ & \text{【DV\#0】} = 50 \quad \& \quad \text{【DV\#1】} = 2、 \\ & \text{【DV\#0】} = 20 \quad \& \quad \text{【DV\#1】} = 5、 \quad \text{等々。} \end{aligned}$$

各データはさらに、2バイトに分割して適用カウンタに書き込みます。

操作手順

```
outp(BASE+7, 0x3) ; /* 素子1の制御レジスタを対象に指定 */
outp(BASE+6, 0x36) ; /* 制御データ【注2】: 固定値 */
```

```
outp(BASE+7, 0x0) ; /* 素子1のカウンタ#0を対象に指定 */
outp(BASE+6, DV#0L) ; /* 下位バイト書き込み */
outp(BASE+6, DV#0H) ; /* 上位バイト書き込み */
```

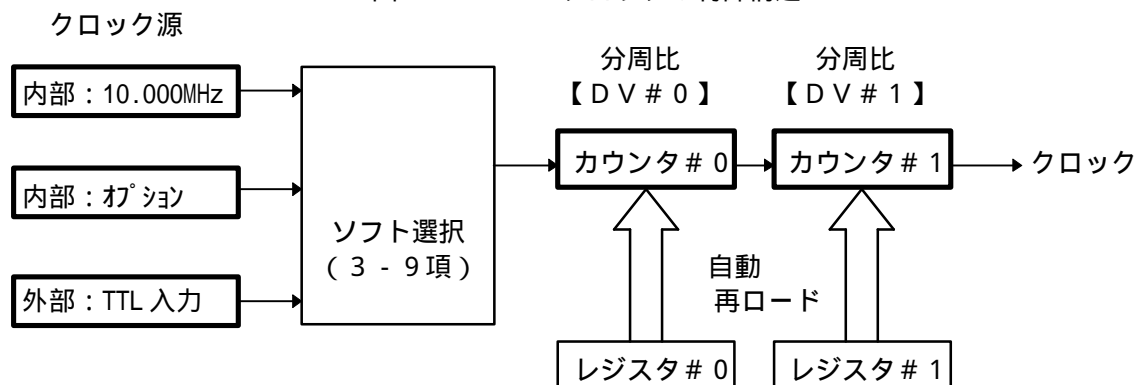
```
outp(BASE+7, 0x3) ; /* 素子1の制御レジスタを対象に指定 */
outp(BASE+6, 0x76) ; /* 制御データ【注3】: 固定値 */
```

```
outp(BASE+7, 0x1) ; /* 素子1のカウンタ#1を対象に指定 */
outp(BASE+6, DV#1L) ; /* 下位バイト書き込み */
outp(BASE+6, DV#1H) ; /* 上位バイト書き込み */
```

【注1】： 書き込み対象選択については前3 - 10項、参照。
また、各カウンタの書き込みは下位バイトを先に、続けて上位バイトの順とする。

【注2】 / 【注3】： カウンタ#0 / #1を“同素子のモード3”で使用する意味の固定データ。（無条件に適用してください。）

図 3 - 1 1 A . クロックの制御構造



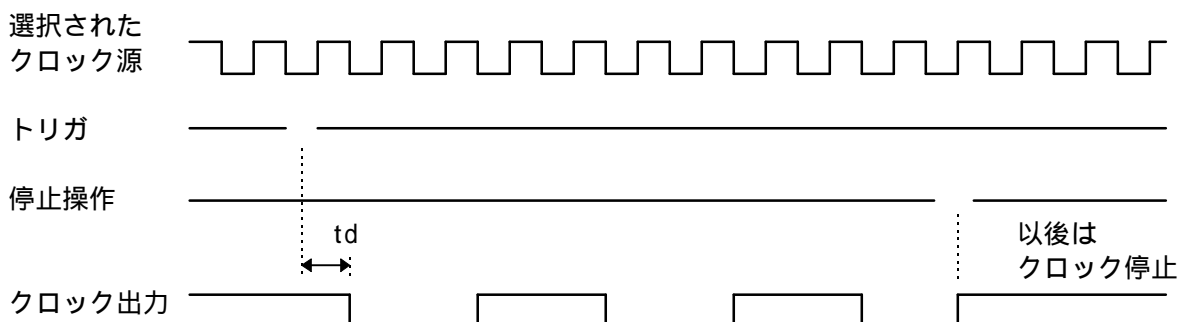
各カウンタ，レジスタは2バイト（16BIT）
 総合分周比【DV】＝【DV # 0】×【DV # 1】

（3 - 9 項で）選択されたクロック源信号は“素子1”内のカウンタ # 0，# 1 で分周されてクロック値となります。カウンタ # 0，# 1 は直列に接続された各16BITのバイナリDOWNカウンタで、当素子（μPD71054）のモード3で動作させます。

前記“操作手順”により書き込まれた分周比データは（実際には）、各カウンタ専用の入力レジスタ # 0，# 1 に保持されており、クロックスタート（トリガ発生）操作により各カウンタにロードされて計数が開始されます。総合32BITの当カウンタが“0”までカウントダウンすると1クロックを出力し、次のカウント・タイミングで入力レジスタ内のデータが再ロードされて再びカウントダウンが続けられます。この動作の様子を図3 - 1 1 Bに示します。

【注4】 クロックスタート（トリガ発生）時にも1クロック出力があります。

図 3 - 1 1 B . クロック発生タイミング



《動作説明》 上記はトリガ（クロックスタート＝出力開始）後、3クロック出力後に停止操作等で動作終了となった例。分周比は4。なお、

クロック源信号、およびクロック出力の有効エッジは常に立下り（ ）。最初のクロック出力遅れtdは最大で1クロック源周期。

3-12. サンプリングデータ点数の設定・認識

有限サンプリングモードでのサンプリングデータ点数は前3 - 11項で得たクロック出力をカウントして得ます。

当カウンタは32BIT(16BITバイナリカウンタを2本直列接続：カウンタ素子2)で、

指定データ点数【DC】=【W#0】×【W#1】

なお、カウンタ#0に書き込む値【W#0】：2～65535、
カウンタ#1に書き込む値【W#1】：1～65535、です。

《例》 指定データ点数【DC】=2000の場合、
各16BITカウンタに設定する値は上式から、

【W#0】=500 & 【W#1】= 4、
【W#0】=250 & 【W#1】= 8、
【W#0】=125 & 【W#1】=16、 等々。

各データはさらに、2バイトに分割して適用カウンタに書き込みます。

設定操作手順

```

outp(BASE+7, 0x7) ; /* 素子2の制御レジスタを対象に指定 */
outp(BASE+6, 0x34) ; /* 制御データ【注2】：固定値 */

outp(BASE+7, 0x4) ; /* 素子2のカウンタ#0を対象に指定 */
outp(BASE+6, W#0L) ; /* 下位バイト書き込み */
outp(BASE+6, W#0H) ; /* 上位バイト書き込み */

outp(BASE+7, 0x7) ; /* 素子2の制御レジスタを対象に指定 */
outp(BASE+6, 0x72) ; /* 制御データ【注3】：固定値 */

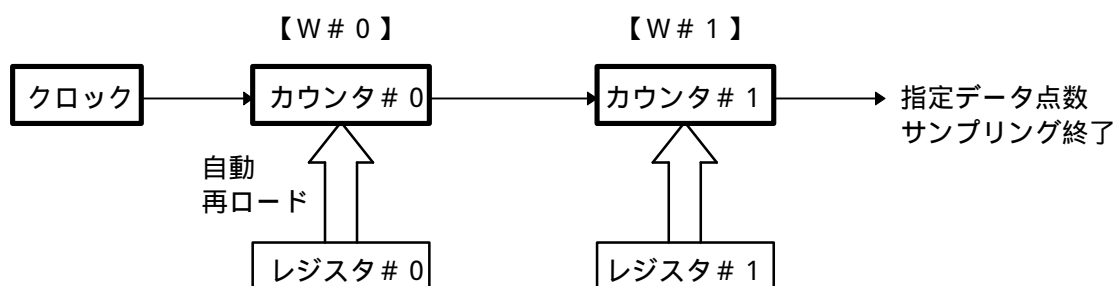
outp(BASE+7, 0x5) ; /* 素子2のカウンタ#1を対象に指定 */
outp(BASE+6, W#1L) ; /* 下位バイト書き込み */
outp(BASE+6, W#1H) ; /* 上位バイト書き込み */

```

【注1】：書き込み対象選択については前3 - 10項、参照。
また、各カウンタの書き込みは下位バイトを先に、続けて上位バイトの順とする。

【注2】/【注3】：カウンタ#0/#1を同素子の“モード2”/“モード1”で使用する意味の固定データ。(無条件に適用してください。)

図3 - 12 . (トリガ後) サンプリングデータ点数の制御構造



各カウンタ、レジスタは2バイト(16BIT)
 全データ点数【DC】=【W#0】×【W#1】

(3-11項で)得られたクロック信号はサンプリング動作に使用されると同時に“素子2”内のカウンタ#0, #1でサンプリング実行点数としてカウントされます。

カウンタ#0, #1は直列に接続された各16BITのバイナリDOWNカウンタで、当素子(μPD71054)のモード2またはモード1で動作させます。

前記“操作手順”により書き込まれた各データは(実際には)各カウンタ専用の入力レジスタ#0, #1に保持されており、クロックスタート(トリガ発生)操作により各カウンタにロードされて計数が開始されます。カウンタ#0が“0”までカウントダウンすると1パルスを出力し、次のカウント・タイミングで入力レジスタ内のデータが再ロードされて再びカウントダウンが続けられます。カウンタ#0からのパルス出力はボード内の制御タイミングに利用されると同時にカウンタ#1の入力ともなります。

#1のダウンカウント値が“0”に達すると全サンプリング終了なのでクロックを止めます。

なお【停止操作等まで無限に】クロック動作を続ける動作モードもあります。

(3-13項、参照)

(トリガ後) サンプリング済みデータ点数の読み込み

F I F Oバッファ入力モードで動作中、素子2のカウンタ# 0 , # 1を読むことにより現在までの(トリガ後)サンプリング済み点数を知ることができます。

動作開始時のカウンタ# 0 設定値【W# 0】

カウンタ# 1 設定値【W# 1】

現在時のカウンタ# 0 読み込み値【R# 0】

1 読み込み値【R# 1】

とすると、

入力済み点数

$$【DON】 = (【W# 1】 - 【R# 1】) \times 【W# 0】 + (【W# 0】 - 【R# 0】)$$

読み込み手順

```

o u t p ( B A S E + 7 , 0 x 7 )      ; /* 素子2の制御レジスタを対象に指定 */
o u t p ( B A S E + 6 , 0 x D 6 )    ; /* 制御データ【注2】: 固定値 */

o u t p ( B A S E + 7 , 0 x 4 )      ; /* 素子2のカウンタ# 0を対象に指定 */
R # 0 L = i n p ( B A S E + 6 )      ; /* 下位バイト読み込み */
R # 0 H = i n p ( B A S E + 6 )      ; /* 上位バイト読み込み */

o u t p ( B A S E + 7 , 0 x 5 )      ; /* 素子2のカウンタ# 1を対象に指定 */
R # 1 L = i n p ( B A S E + 6 )      ; /* 下位バイト読み込み */
R # 1 H = i n p ( B A S E + 6 )      ; /* 上位バイト読み込み */

```

【注1】: 読み込み対象選択については前3 - 10項、参照。
また、各カウンタの読み込みは下位バイトを先に、続けて上位バイトの順とする。

【注2】: カウンタ# 0 / # 1の値を同時ラッチ(出力用ラッチに読み出し)する意味の固定データ。(無条件に適用してください。) 当操作後は必ず続けてカウンタ(ラッチ)の読み込み操作を行うこと。 各ラッチは上位バイト読み出し直後からカウンタに追従を開始する。