

第3章．制御・操作

3-1. 制御・操作とデジタル入力の様子

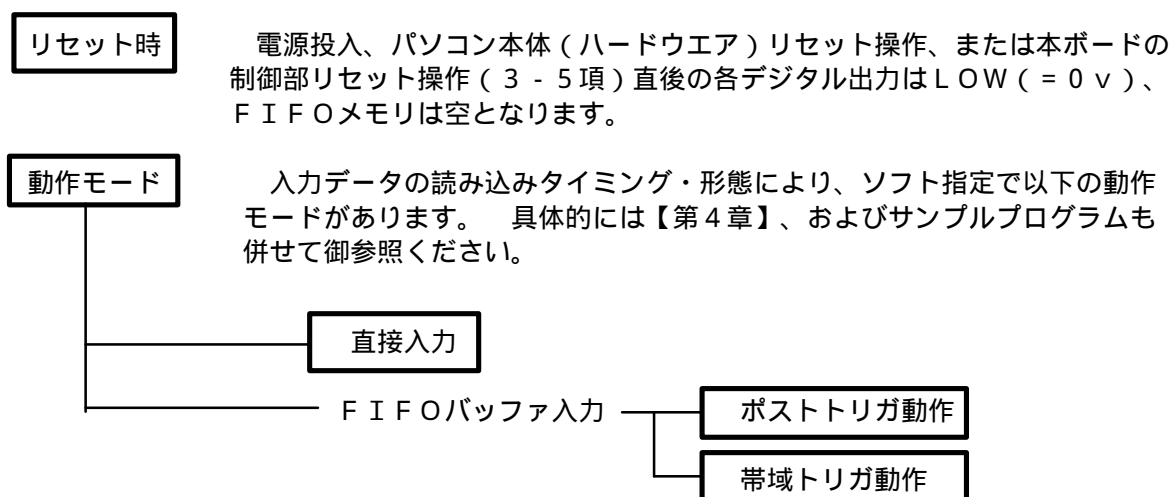
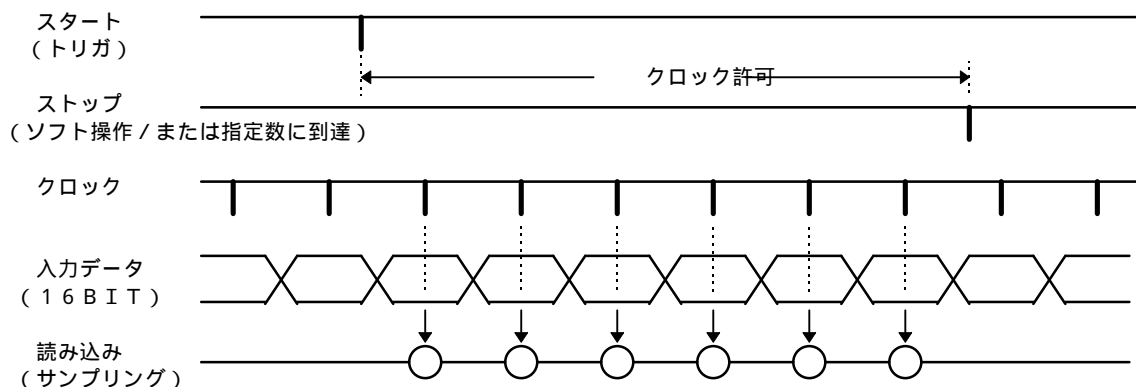


図3 - 1 A . FIFOバッファ入力動作（ポストアトリガの場合）



スタート： ソフトトリガ（即スタート）、または外部トリガ入力の指定エッジにより指定クロック（ここでは外部TTLクロック入力の指定エッジ）が有効となります。

読み込み： 16ビット（TTL）入力データは、外部TTLクロック入力の指定エッジに同期してFIFOバッファメモリに読み込まれます。この動作は指定データ数（最大65,536 / または無限）に達するか、ソフト上の停止操作まで続きます。

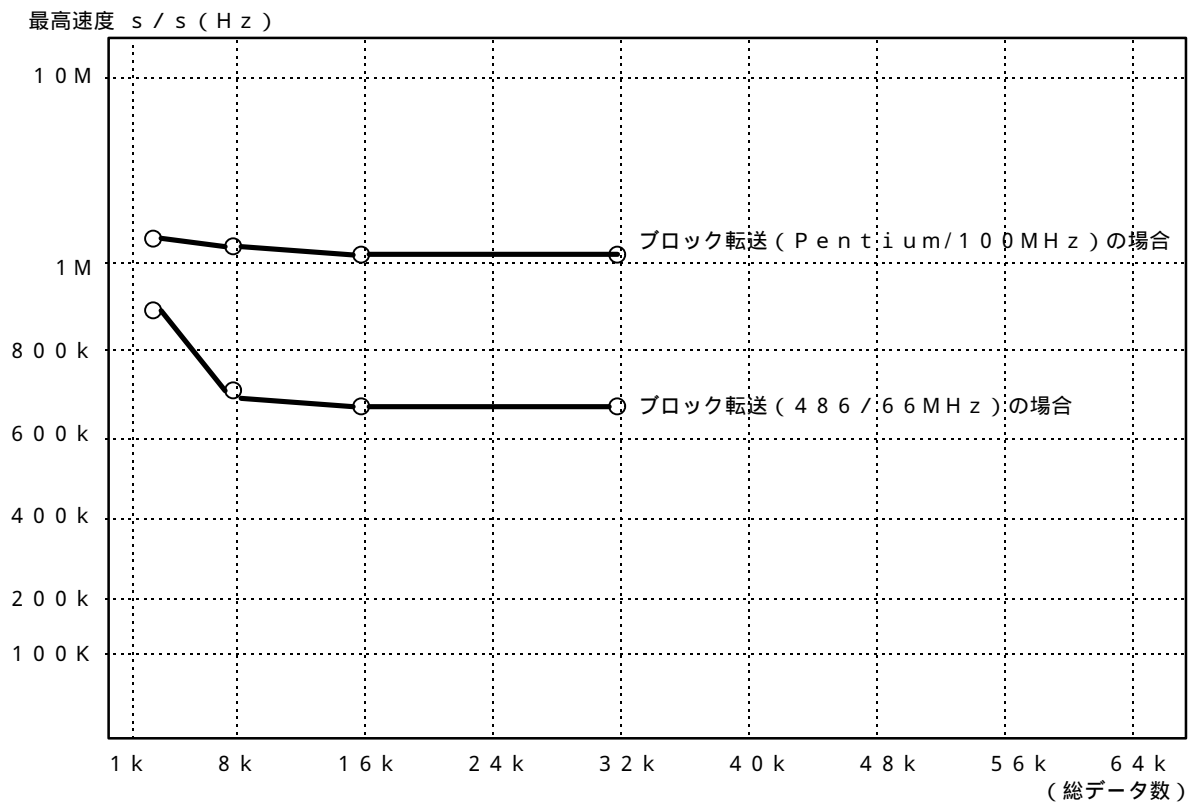
読み出し： パソコン側からはFIFOメモリの充満状態フラグを監視しながら、データが溢れないうちに読み出します。FIFOメモリの入力動作と出力動作は非同期、すなわち互いにタイミングを気を使う必要がありません。（同時に可能）また、データは古い順に出口側に詰めて位置しており、読み出された分だけ入口側に空領域が増えるので実質的にデータ数の制限はありません。

最高サンプリング速度

総データ点数がFIFOメモリ容量以内のときはパソコン側の読み出し速度に関係なく常に最高速度（10MHz）が可能です。

総データ点数がFIFOメモリ容量以上のときは、パソコン側からFIFOメモリ内のデータを（溢れないうちに）逐次読み出す必要があるため、パソコン側へのデータ転送速度・FIFOメモリ容量・総データ点数で可能な最高速度が決まります。（1-1項/表1-1参照）

図3-1B. ポストトリガ動作の最高速度（FIFO：標準1024語のとき）



【注】 本図は特定機種・システムでの実測値です。
 確実な値は各個別システムで実測する必要があります。

3-2. F I F Oメモリの動作

16BIT入力データ転送 (F I F O パソコン)

F I F Oバッファ入力モードで入力 (サンプリング) されたデータはF I F Oメモリ内にあり、パソコン側からの読み出しを待っています。ソフト上ではF I F Oメモリの充満状態を示すフラグを監視またはDMA・割り込み等を設定してアプリケーションに適した転送方法を探ります。

いずれの場合も、F I F Oバッファメモリがサンプリング実行とデータ転送のタイミング違いを吸収するので、マルチタスク・システムを容易に実現することができます。

ポーリング : 【Not-Empty】フラグを監視してデータをIN命令で1語ずつ読み込む方法、
【Half-Full】フラグを監視してデータをINSW命令でF I F Oメモリ容量の半分単位で読み込むブロックI/O転送がある。

割り込み : 【Not-Empty】【Half-Full】【クロック】【トリガ発生】【外部割り込み】等から選択した要因による割り込み処理ルーチン内で、ポーリングと同様のデータ転送を実行する。

DMA : 【Not-Empty】状態によりDMAを起動・実行する。

F I F Oメモリの動作

F I F O (first in first out) メモリは図3 - 2に示すパイプ状の構造をしており、入口から書き込まれたデータは自動的に最奥部まで転がり込んで行き、出口では最古のデータから順に読み出される構造となっている。

読み書きは非同期で同時も可能。すなわちデータの書き込み (入口) と読み出し (出口) は互いに相手側のタイミングに配慮する必要がない。

F I F Oメモリ内部は出口から読み出された分だけ入口側に空領域が増えるので満杯となる前にデータを読み出す動作であればサンプリング点数を制限しない。

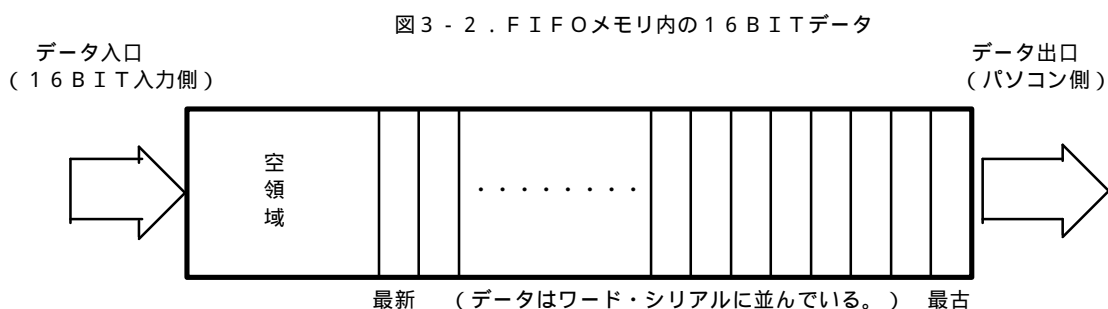


表3 - 2 . F I F Oメモリ関連のステータス・フラグ

フラグ名	フラグの意味 (標準1024語のとき)	
Not-EMPTY	1語以上。	(格納データ数 1)
Not-HALF-FULL	[容量の半分]以下。	(格納データ数 512)
ERR	サンプリング・エラー発生。	データ読み出しが遅いためメモリが溢れた、または10MHz以上の速度で書き込みした。

3-3. 制御・操作の手順

直接入力

最初に リセット・転送モード設定を行ったら、以後はプログラム上で任意に入力操作を行うことができます。

```
rst = inp (BASE + 7)      ; /*制御部リセット*/
outp (BASE + 8, trsm)    ; /*転送モード設定 (trsm = 0)*/
DL = inp (BASE + 0)      ; /*16BITデータ入力 (下位バイト)*/
DH = inp (BASE + 1)      ; /*16BITデータ入力 (上位バイト)*/
```

FIFOバッファ入力

～ までの手順・操作でサンプリング動作が即開始、またはトリガ待ち状態となります。ここまでは全んどの応用で同一手順となります。特に は割り込み/クロック/トリガの許可操作ですから最後に行います。

```
rst = inp (BASE + 7)      ; /*制御部リセット*/
outp (BASE + 8, trsm)    ; /*転送モード設定*/

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

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: 固定値 (3 - 11項参照)*/
outp (BASE + 7, 0x0)     ; /*素子1の#0カウンタを対象に指定*/
outp (BASE + 6, dv#0L)   ; /*クロック源分周比 (#0下位バイト)*/
outp (BASE + 6, dv#0H)   ; /*      "      "      (#0上位バイト)*/
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: 固定値 (3 - 11項参照)*/
outp (BASE + 7, 0x1)     ; /*素子1の#1カウンタを対象に指定*/
outp (BASE + 6, dv#1L)   ; /*      "      "      (#1下位バイト)*/
outp (BASE + 6, dv#1H)   ; /*      "      "      (#1上位バイト)*/
```

前記、クロック源分周比設定の要領でサンプリング点数カウンタに書き込む。(3 - 12項)

```
outp (BASE + 5, idc)     ; /*割り込みレベル&DMAチャンネルの設定*/
パソコン内 (ISAバス上) の割り込み、またはDMAコントローラ設定【前処理】
```

```
outp (BASE + 2, str)     ; /*トリガモード設定【START操作】*/
```

以後はステータスを評価して適切な以下の操作を行う。

```
sts = inp (BASE + 10)    ; /*ステータス取得*/
DL = inp (BASE + 0)      ; /*16BITデータ入力 (下位バイト)*/
DH = inp (BASE + 1)      ; /*16BITデータ入力 (上位バイト)*/
```

```
outp (BASE + 2, stp)     ; /*トリガモード設定【STOP操作】*/
パソコン内 (ISAバス上) の割り込み、またはDMAコントローラ設定【後処理】
```

【注1】 サンプリング点数の設定は有限サンプリング (: 3 - 12項) のときのみ必要。

【注2】 割り込み、DMAに関する設定は使用する場合のみ必要。