

## はじめに

本マニュアルは、同時サンプリング A/D ボード PVME-303 概要、取り扱いについて述べたものです。

PVME-303 は、同時サンプリング機能を備えた 12 ビット分解能 8 チャンネルの VME bus 対応の高速、高性能、低価格な A/D 入力ボードです。

独立型 A/D 変換方式の採用によりチャンネル数によってスループットレートが変わることはなく、しかも、最速サンプリングレート  $10\mu\text{s}$  での高速変換が可能です。

A/D 変換データは FIFO RAM へ蓄積されるため A/D 変換実行中にデータを読み込むことが可能です。

A/D 変換は内部インターバルタイマ/外部クロックのどちらにも同期できる他、最終チャンネル番号、及び変換回数が指定できるため FIFO RAM を有効に使用できます。

本ボードはマスタ/スレーブ構成にカスケード接続ができるため、ボード枚数を増やすことにより多チャンネルのアプリケーションに簡単に対応することができます。

ご使用になる前に、本マニュアルの内容をよくご理解いただき、有効にご利用ください。

# 目 次

1.概要	
1-1 特長	1
1-2 仕様	2
1-3 ブロック図	4
1-4 アドレス・マップ	5
2.取扱い方法	
2-1 部品の確認	6
2-2 御使用上の注意事項	6
3.機能	
3-1 アドレス空間	7
3-1-1 AMコード	7
3-1-2 ベース・アドレス設定	7
3-1-3 アドレス・モード設定	9
3-2 A/D変換モード	10
3-2-1 概要	10
3-2-2 モード設定	12
3-2-3 使用チャンネル数設定	12
3-2-4 変換回数設定	13
3-2-5 同期信号	15
3-2-6 トリガ信号	19
3-2-7 変換データの読み出し	21
3-2-8 マスタ/スレーブ構成	24
3-2-9 フィルタリング機能	27
4.割り込み	
4-1 割り込み	28
4-2 コントロール レジスタ	30
4-3 ベクタ・レジスタ	31
5.ポーリング	
5-1 ポーリング	32
6.レジスタ・フォーマット	
6-1 レジスタ・フォーマット	33

7.ディップスイッチ/ジャンパー設定	
7-1 初期設定	42
7-2 アドレス・モード設定	43
7-3 ベース・アドレス設定	44
7-4 モード/データ・フォーマット/F I F Oモード設定	45
7-5 インターバルタイマ単独/カスケード設定	45
7-6 入力レンジ設定	46
7-7 電圧/電流入力設定	47
7-8 SCK IN/SCK OUT/TRG INライン ターミネイト抵抗設定	47
8.ピン・アサイン	
8-1 フロントパネル外観	48
8-2 アナログ入力コネクタ (CN 1)	49
8-3 同期信号コネクタ (CN 2, CN 3)	50
8-4 VME busコネクタ (P 1, P 2)	52
9.アナログ調整	
9-1 調整方法	55
9-2 部品位置	59
9-3 調整プログラム (TEST303:C / ADRES303:H)	60
10.サンプル・ソフト	
10-1 ソフト概要	64
10-2 ソフト設定	64
10-3 プログラム構成	65
11.その他	
11-1 保証について	73
11-2 オーダリング・インフォメーション	73

# 第 1 章 概 要

## 1-1 特長

PVME-303 は VME bus (Rev C.1) 仕様の同時サンプリング A/D 入力ボードであり特長は次の通りです。

### ○ A/D 変換機能

- ・ 分解能 12 bit
- ・ 高スループットレート 10  $\mu$ s
- ・ 入力チャンネル 8 ch (差動)
- ・ 入力レンジ (ジャンパーセレクト)
  - ユニポーラ時
  - 1) 0 ~ + 2.5 [v]
  - 2) 0 ~ + 5.0
  - 3) 0 ~ + 10.0
  
  - バイポーラ時
  - 4) - 2.5 ~ + 2.5 [v]
  - 5) - 5.0 ~ + 5.0
  - 6) - 10.0 ~ + 10.0

- ・ 独立型 A/D 変換方式
- ・ マスタ/スレーブ構成により多チャンネル対応
- ・ オート・スキャン機能
- ・ 内部/外部トリガ
- ・ 内部インターバルタイマ/外部クロックに同期したサンプリング
- ・ 同時サンプリング回数 1 ~ 65536回/無限

- ・ FIFOメモリ容量 4092ロング・ワード/8192ワード

### ○ 割り込み機能

- ・ 68153 (Bus interrupt Modale) 使用
- ・ 割り込み要因

1) FIFOメモリ	FF	信号
2) FIFOメモリ	HF	信号
3) FIFOメモリ	EF	信号
4) スキャン終了	DONE	信号

### ○ ポーリング機能

- ・ ステータス

1) FIFOメモリ	FF	信号
2) FIFOメモリ	HF	信号
3) FIFOメモリ	EF	信号
4) スキャン終了	DONE	信号

## 1-2 仕様

インターフェース電氣的仕様（特に記述のない限り、TA=25°C、規定電源電圧）

バス型式	VMEバス規格 (Rev C.1) 準拠
アドレス	A 16 / A 24 / A 32 ビット、アドレス対応 連続256バイト (128ワード) 占有 上位アドレス、ロータリ・スイッチにてフルデコード
AMコード	特権、非特権 データ・アクセス ショート I/O・アクセス ロータリ・スイッチにて選択
転送データ幅	D 8 / D 16 / D 32 対応
モード	スレーブ
インターラプト	ベクタ方式 割込レベル：1～7レベルをソフトウェアセレクト 要求解除：応答時開放 (ROAK) 又は、 レジスタ・アクセス解放時 (RORA)
動作表示	スレーブ/マスタ・アクセス時及びA/D変換実行中に LED点灯

### A/D部

入力チャンネル	差動 8 ch
入力レンジ	ユニポーラ時 0 ~ + 2.5 [v] 0 ~ + 5.0 0 ~ +10.0 バイポーラ時 - 2.5 ~ + 2.5 [v] - 5.0 ~ + 5.0 -10.0 ~ +10.0 各チャンネルごとにジャンパー設定
最大入力電圧	± 15 [v] (電源 ON 時)
出力コード	ユニポーラ時 ストレート・バイナリ バイポーラ時 オフセット・バイナリ
入力インピーダンス	10 MΩ 以上
入力キャパシティ	± 30 pF 以下
入力リーク電流	50 nA 以下
分解能	12 ビット
スループット レート	10 μsec (チャンネル数に無関係)
オフセット エラー	± 60 mV (ゼロ調整可能)
オフセット ドリフト	55 μV (TYP)
フルスケール エラー	1% (ゼロ調整可能)
フルスケールドリフト	30 ppM/°C
リニアリティ エラー	± 0.013 %

## デジタル部

外部トリガ	入力レベル	TTLレベル	立下りエッジ (▽□)
	パルス幅 (min)		50 [nS]
	ディレイタイム (max)		50 [nS]
外部クロック	入力レベル	TTLレベル	立下りエッジ (▽□)
	パルス幅 (min)		50 [nS]
	ディレイタイム (max)		50 [nS]
スレーブ同期	出力波形		パルス (□□)
	出力レベル	TTLレベル	
	出力電流	IOL (mix)	48 [mA]
		IOH (max)	-15 [mA]
	パルス幅	(min)	100 [nS]
インターバル・タイム	10 [μs]	—	7 [s]
FIFO・メモリ			125 [ns] ステップ°
		8 1 9 2	[ワード]
	4 0 9 6		[ロング・ワード]

## 電源・外形

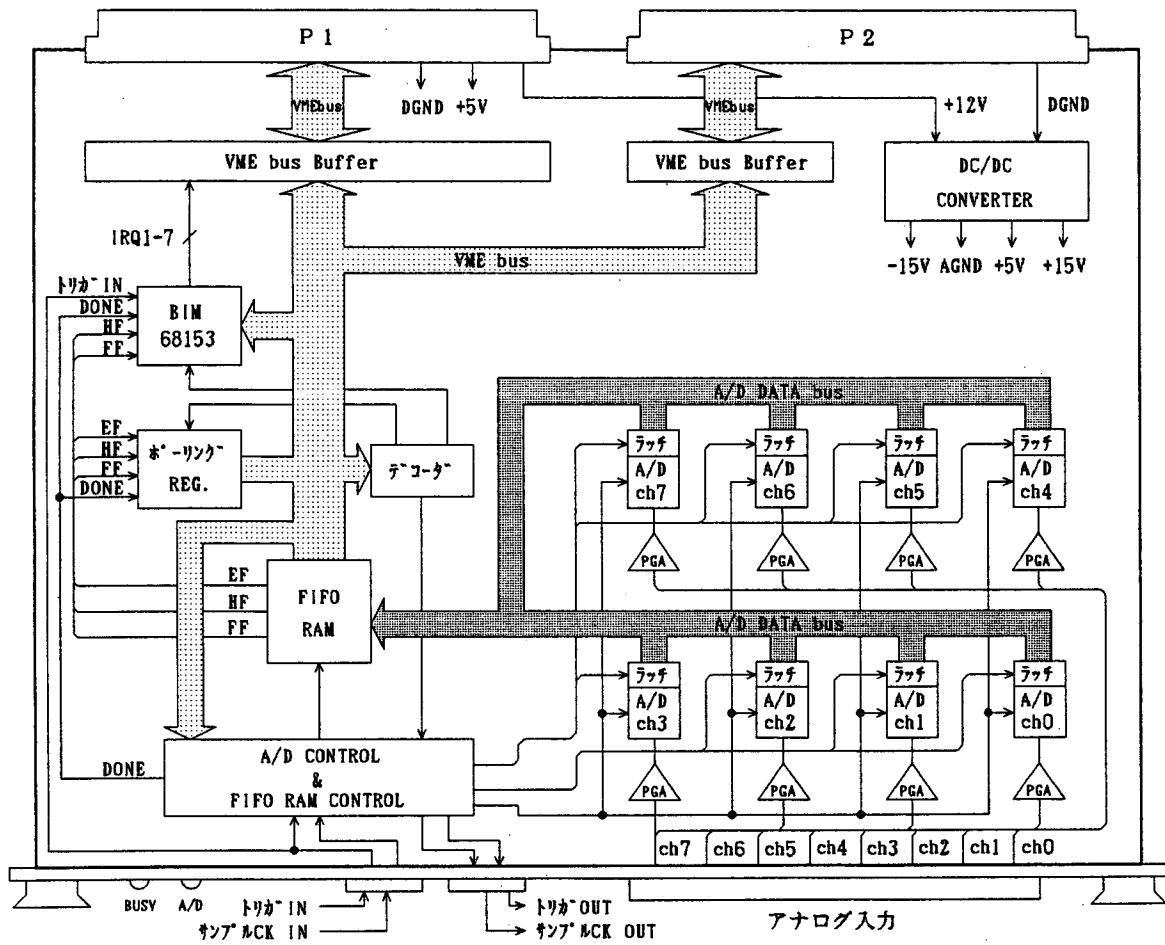
電 源	+ 5 [V] ± 5 %	3.0 [A]	MAX
ボードサイズ	ダブルハイト、	160 × 233	[mm]

## 温度・湿度規定

動作時	温 度	: 0 ~ + 60 °C
	湿 度	: 20 ~ 80 % 結露無
保存時	温 度	: -10 ~ + 70 °C
	湿 度	: 10 ~ 90 % 結露無

1-3 ブロック図

PVME-303 ブロック図





#### 1-4 アドレス・マップ

PVME-303 は、連続した256バイト（128ワード）空間を占有します。  
占有空間のアドレス・マップは次の通りです。

	UPPER BYTE	LOWER BYTE
Base+ 00h	*	コントロール レジスタ 0
02	*	コントロール レジスタ 1
04	*	コントロール レジスタ 2
06	*	コントロール レジスタ 3
08	*	ベクタ レジスタ 0
0A	*	ベクタ レジスタ 1
0C	*	ベクタ レジスタ 2
0E	*	ベクタ レジスタ 3
10	*	タイマー レジスタ 0
12	*	タイマー レジスタ 1
14	*	カウンタ レジスタ 0
16	*	コントロール レジスタ 4
18	*	*
1A	*	*
1C	*	*
1E	*	*
20	*	コントロール レジスタ 5
22	*	コントロール レジスタ 6
24	*	コントロール レジスタ 7
26	*	コントロール レジスタ 8
28	*	コントロール レジスタ 9
2A	*	コントロール レジスタ 10
2C	*	*
2E	*	*
30	データ レジスタ 0	
32	データ レジスタ 0	
34	*	*
36	*	*
38	*	*
3A	*	*
3C	*	*
3E	*	*
40	*	*
~	~	
FC	*	*
FE	*	*

注) ① D 8 / D 16 / D 32 アクセスに対して DTACK が発生されませんが、各 REGISTER  
 において、指定以外のアクセス時では、データが不確定となる場合があります。  
 (第 6 章レジスタ・フォーマットの参照を願います。)

② \* : 未使用

## 第 2 章 取扱い方法

## 2-1 部品の確認

お買い上げいただきました PVME-303 の梱包を解いたときに、下記の物があることを御確認下さい。

- ・ PVME-303 ユーザーズ・マニュアル ..... 1部
- ・ PVME-303 (部品面シールド板付) ..... 1枚
- ・ D-SUB 37P コネクタ及びハウジング ..... 1組
- ・ リボン・コネクタ (10P) メス側 ..... 2組
- ・ 保証書 ..... 1通

## 2-2 御使用上の注意事項

- ・ PVME-303 をジャンパー設定変更などで、ラックから抜き差しする際には、電源を切って行い、また、導電性のものの上に置き作業して下さい。このとき、人体アースを取ることも忘れないでください。
  
- ・ PVME-303 をラックに入れて使用する場合、振動により接触不良が起きることがありますので、パネル上下2箇所ネジを締めて御使用下さい。  
また、ネジを締めることにより、シールド板及び、入力チャンネル・コネクタ表面がフレーム・グラウンドと導通されます。
  
- ・ PVME-303 には、ノイズの影響を極力防げるため部品面側にシールド板が実装されていますが、はんだ面側には特にシールド対策を施しておりません。  
したがって、はんだ面側スロットは極力空けるか、もしくは、はんだ面側スロットに装着されるボードの部品面側にも、シールド板を実装し御使用されることを推奨いたします。
  
- ・ PVME-303 を保存する場合、保存環境条件を満たし、直射日光を避け、静電破壊を防ぐためには、はんだ面に導電性のものをつけて保存するようにして下さい。

# 第 3 章 機 能

### 3-1 アドレス空間

PVME-303 は、VMEシステムにて常にスレーブ・ボードとしてアクセスされます。

特権状態は、ユーザ定義によりスーパーバイザ/非特権のどちらか一方の状態、又は両方の状態を選択することができます。

アドレス形式は、ショート・アドレッシング、スタンダード・アドレッシング（標準）、エクステンディッド・アドレッシング（拡張）から選択可能です。また、各形式では、128ワード（A<sub>1</sub>～A<sub>7</sub>）を占有アドレスとして使用し、ベース・アドレス（上位アドレス）は自由に設定が可能です。

#### 3-1-1 AMコード

希望する状態をロータリ・スイッチ（LD7）にてユーザ定義することで、AMコードに対応することが可能です。

対応可能なAMコードを示します。

HEX コード	アドレス・モディファイヤ						機 能
	5	4	3	2	1	0	
3D	H	H	H	H	L	H	標準スーパーバイザ・データ・アクセス
39	H	H	H	L	L	H	標準非特権・データ・アクセス
2D	H	L	H	H	L	H	ショート・スーパーバイザ・アクセス
29	H	L	H	L	L	H	ショート・非特権・アクセス
0D	L	L	H	H	L	H	拡張スーパーバイザ・データ・アクセス
09	L	L	H	L	L	H	拡張非特権・データ・アクセス

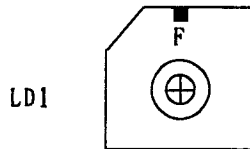
L = low 信号レベル      H = high 信号レベル

#### 3-1-2 ベース・アドレス設定

各アドレス形式においてロータリ・スイッチ（LD1～LD6）にて設定して下さい。

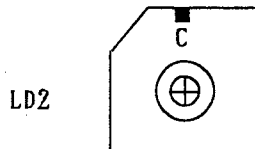
①アドレス A 3 1 - A 2 8 設定

LD1 にてHex コードにより設定します。



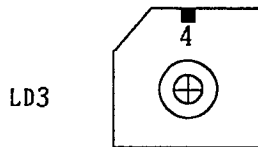
②アドレス A 2 7 - A 2 4 設定

LD2 にてHex コードにより設定します。



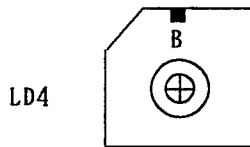
③アドレス A 2 3 - A 2 0 設定

LD3 にてHex コードにより設定します。



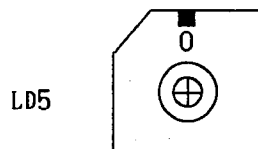
④アドレス A 1 9 - A 1 6 設定

LD4 にてHex コードにより設定します。



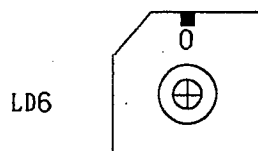
⑤アドレス A 1 5 - A 1 2 設定

LD5 にてHex コードにより設定します。



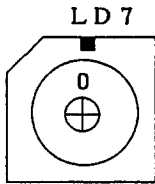
⑥アドレス A 1 1 - A 8 設定

LD6 にてHex コードにより設定します。



### 3-1-3 アドレス・モード設定

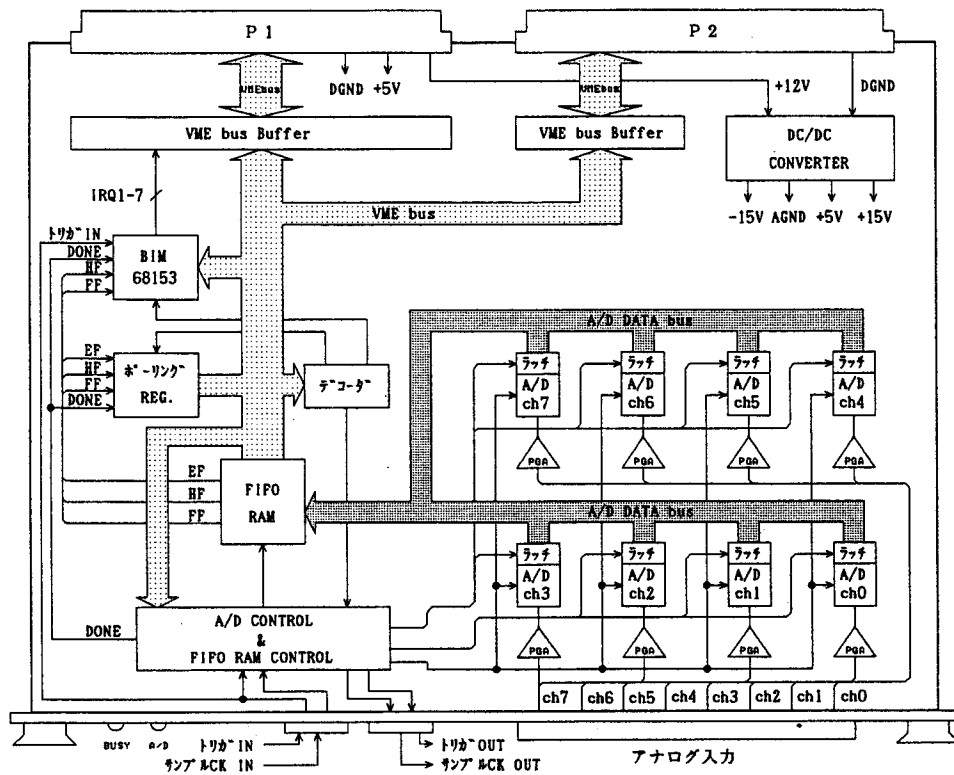
本モードのアドレス・モードはロータリ・スイッチ（LD7）にて設定が行えます。



LD7	特権／非特権／OR	拡張／標準／ショート
0	非特権	拡張
1	非特権	標準
2	非特権	ショート
3	*	*
4	特権	拡張
5	特権	標準
6	特権	ショート
7	*	*
8	OR	拡張
9	OR	標準
A	OR	ショート
B	*	*
C	*	*
D	*	*
E	*	*
F	*	*

\*：禁止

### 3-2 A/D変換モード



PVME-303 ブロック図

#### 3-2-1 概要

本ボードはブロック図の通り独立型A/D変換方式を採用することで、チャンネル数によるスループットレートの変化がありません。

変換データは順次FIFOメモリに貯えられ、フラグ(EF/HF/FF)状態を監視しながら、必要なときに読み取ることができます。又、変換データにはチャンネル番号が付加できますのでデータ管理に役立ちます。

変換シーケンスはトリガ信号(内部/外部)発生後、同期信号(インターバルタイマ/外部クロック)に同期しながら、設定に応じたチャンネル数、及び、変換回数にて変換を行います。変換終了時には、終了ステータス信号(DONE)を発生します。本ボードはマスタ/スレーブ構成のカスケード接続により多チャンネル同時サンプリングに対応することができます。



## モード

### ①モード1（内部トリガーインターバル タイマー）

内部トリガにてインターバル タイマーに同期した変換が設定チャンネル、及び変換回数で行われ、変換データは FIFO メモリに蓄積されます。FIFO メモリはデータ レジスタ 0 にて読み出しが行なえます。

### ②モード2（内部トリガー外部クロック）

内部トリガにて外部クロック信号に同期した変換が設定チャンネル、及び変換回数で行われ、変換データは FIFO メモリに蓄積されます。FIFO メモリはデータ レジスタ 0 にて読み出しが行なえます。

### ③モード3（外部トリガーインターバル タイマー）

外部トリガにてインターバル タイマーに同期した変換が設定チャンネル、及び変換回数で行われ、変換データは FIFO メモリに蓄積されます。FIFO メモリはデータレジスタ 0 にて読み出しが行なえます。

### ④モード4（外部トリガー外部クロック）

外部トリガにて外部クロックに同期したサンプリングが設定チャンネル、及び変換回数で行われ、変換データは FIFO メモリに蓄積されます。FIFO メモリはデータ レジスタ 0 にて読み出しが行なえます。

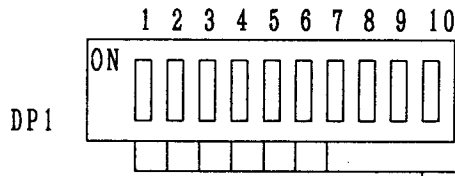
### ⑤モード5（スレーブ）

マスタ（モード1～4）に設定されたボードからの同期信号に同期したサンプリングを設定チャンネルに従い行います。変換データは FIFO メモリに蓄積されます。FIFO メモリはデータレジスタ 0 にて読み出しが行なえます。

注：各モードはボード上のディップ スイッチ（DPI）にて設定します。

### 3-2-2 モード設定

モード設定はディップ スイッチ (DP1) にて行います。



注) 10: 未使用

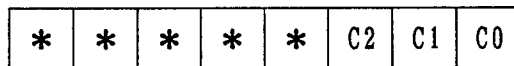
1	2	3	4	5	6	7	モード (マスタ)/スレーブ
ON	OFF	OFF	ON	OFF	ON	ON	1 (マスタ)
ON	OFF	OFF	OFF	ON	ON	OFF	2 (マスタ)
OFF	ON	OFF	ON	OFF	ON	ON	3 (マスタ)
OFF	ON	OFF	OFF	ON	ON	OFF	4 (マスタ)
OFF	OFF	ON	OFF	OFF	OFF	OFF	5 (スレーブ)

### 3-2-3 使用チャンネル数設定

コントロール レジスタ 6 に本ボードで使用するチャンネル数を所定のフォーマットに従いバイト・アクセスにて書き込みます。チャンネル数は最大 8チャンネルまで設定が行なえます。

コントロール レジスタ 6 (ベース・アドレス + 23h)

フォーマット, D7 D6 D5 D4 D3 D2 D1 D0



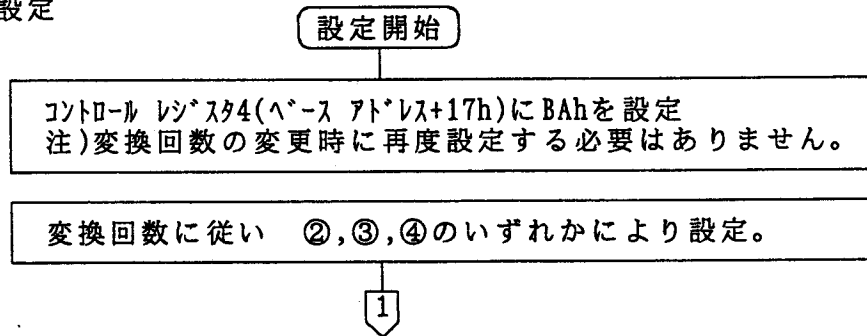
C2	C1	C0	チャンネル数
0	0	0	1
0	0	1	2
~	~	~	~
1	1	0	7
1	1	1	8

### 3-2-4 変換回数設定

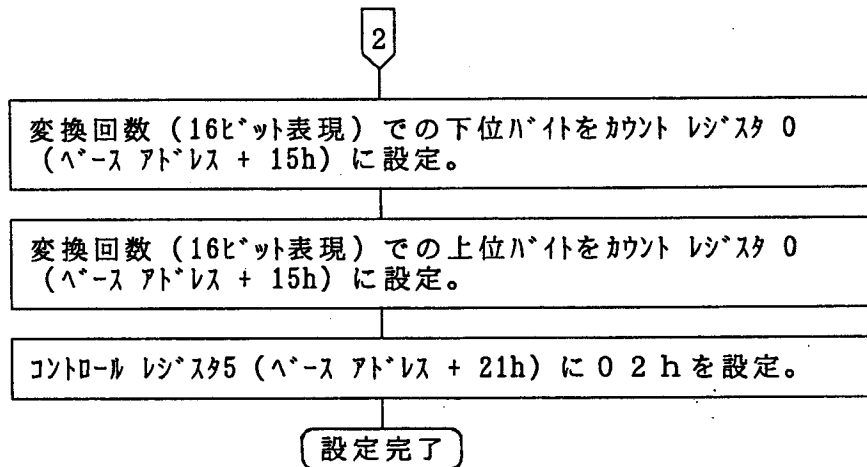
変換回数は1～65536回と無限回の範囲にて設定が行えます。変換回数は16ビット・カウンタによりカウントされますので、設定については、次のシーケンスに従い、カウントレジスタ0、コントロールレジスタ4、コントロールレジスタ5に値をバイト・アクセスにて書き込んで下さい。

#### ○設定シーケンス

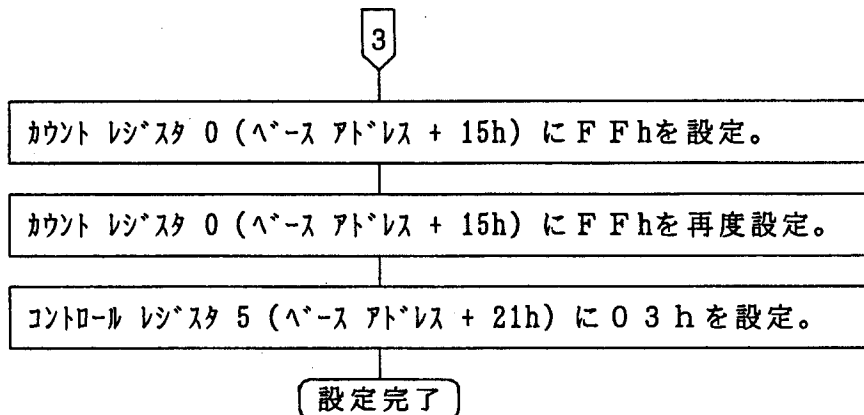
##### ①初期設定



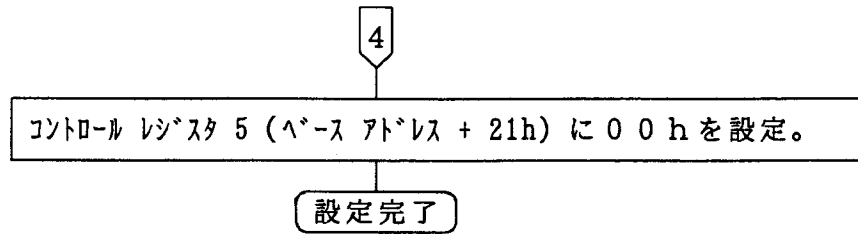
##### ②変換回数 = 1～65535回



##### ③変換回数 = 65536回の場合



④変換回数 = 無限回の場合



注) 無限回設定時は カウントレジスタ0の変換回数設定は必要ありません。

○ 変換終了

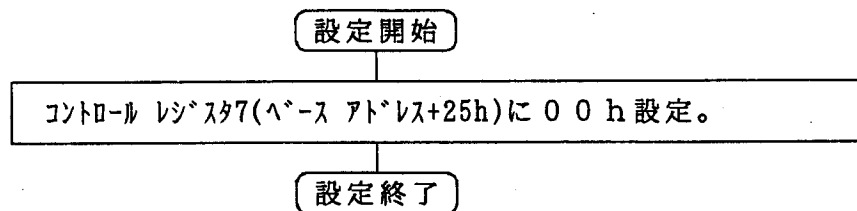
変換回数 = 1 ~ 65536回の場合

変換回数が設定値に達すると終了します。このとき、終了を知らせる DONE 信号が発生、コントロールレジスタ5の DONE ビット及び割り込み要因 DONE をアサートします。

変換回数 = 無限回の場合

変換回数を無限回に設定した時の変換中止とは コントロールレジスタ7に指定の値をバイト・アクセスにて書き込むことで設定します。

又、1 ~ 65535回の設定時にて変換を強制的に中止させることができます。ただし、この場合は、設定回数に達する前であれば、DONE 信号はアサートされません。



### 3-2-5 同期信号

#### ○ インターバルタイマー

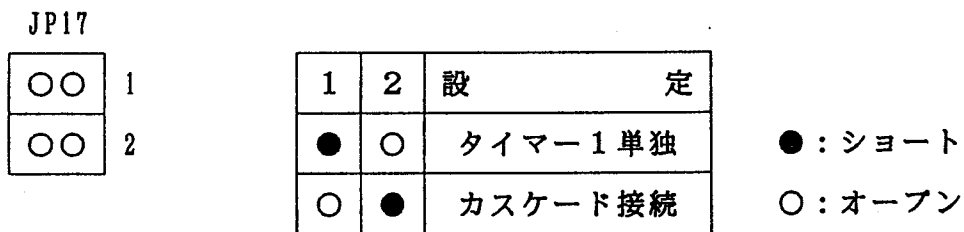
インターバルタイマーは、 $10\mu\text{s}$ ～約9分の範囲を最小  $125\text{ns}$  ステップにて設定が行えます。又、16ビットタイマー1及びタイマー2から構成されている為、設定時間に応じて単独でタイマー1を使用する場合とタイマー1及びタイマー2をカスケード接続で使用する場合の選択をジャンパー (JP17) にて行います。

インターバルタイマーの設定は次のシーケンスに従い、タイマーレジスタ0、タイマーレジスタ1、コントロールレジスタ4に値をバイト・ライトすることで行います。

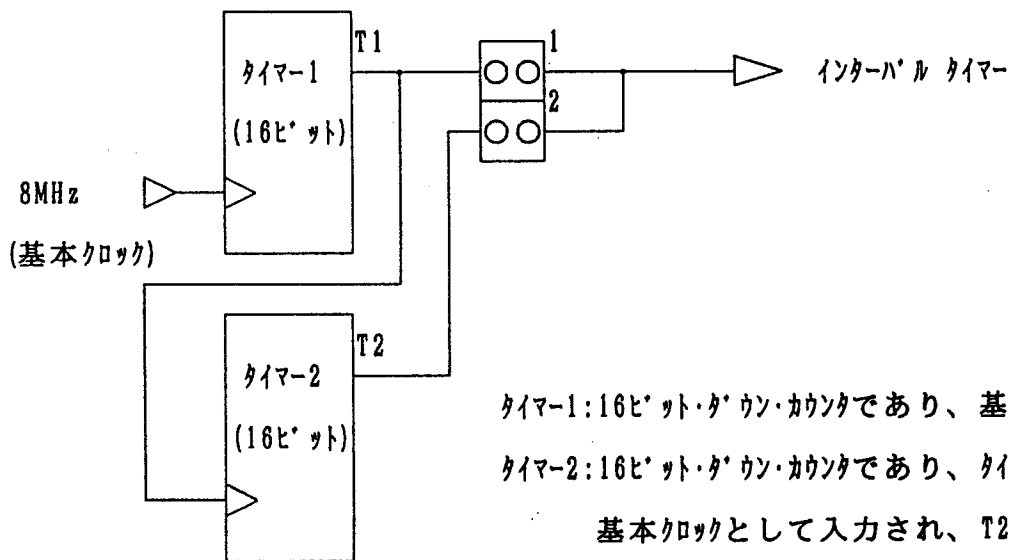
#### (設定シーケンス)

##### 1) 単独/カスケード選択

タイマー1単独/カスケード接続の選択を JP17 のフォーマットに従い設定します。



JP17 フォーマット



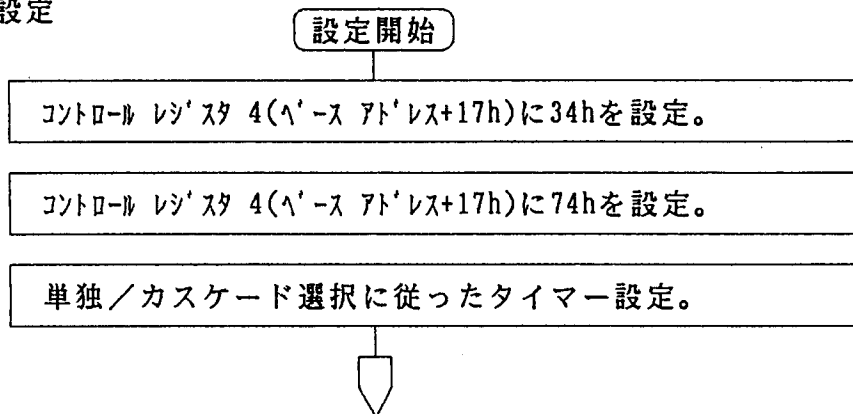
インターバルタイマ・ブロック図

## 2)タイマー設定

タイマーの設定は次のシーケンスに従いコントロールレジスタ4及びタイマーレジスタ0、タイマーレジスタ1に値をバイト・アクセスにて書き込むことを行います。

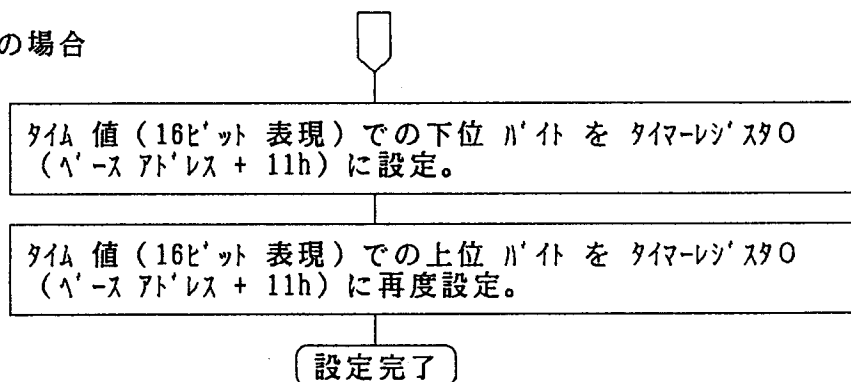
### ○ 設定シーケンス

#### ①初期設定



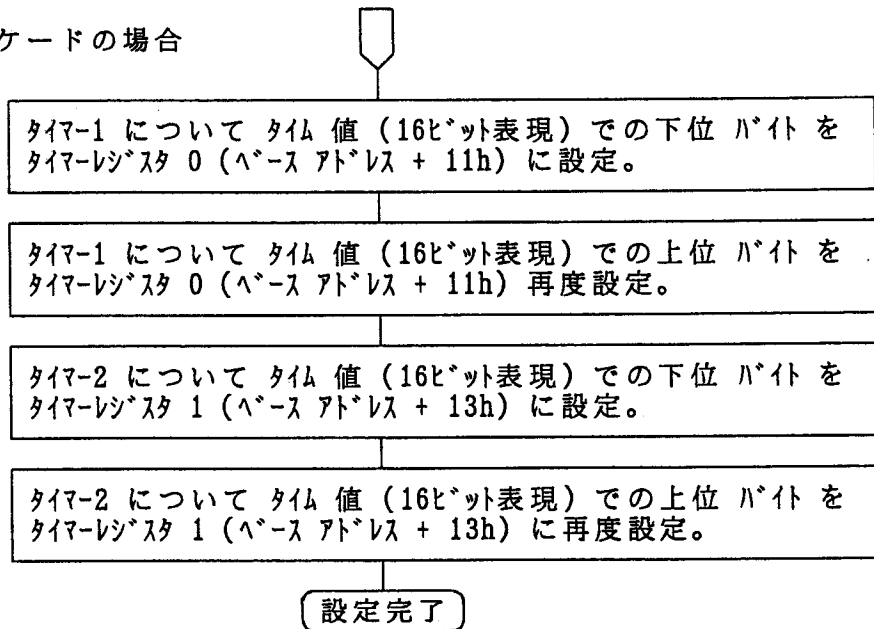
注) ファンクションレジスタはタイマー設定変更時に再設定の必要はありません。

#### ②単独の場合



注) ①最小設定値は80(0050h)です。

③カスケードの場合



注) ①タイマー 1 の最小設定値は 80 (0050h) です。

①タイマー 2 の最小設定値は 2 (0002h) です。

○ 外部クロック

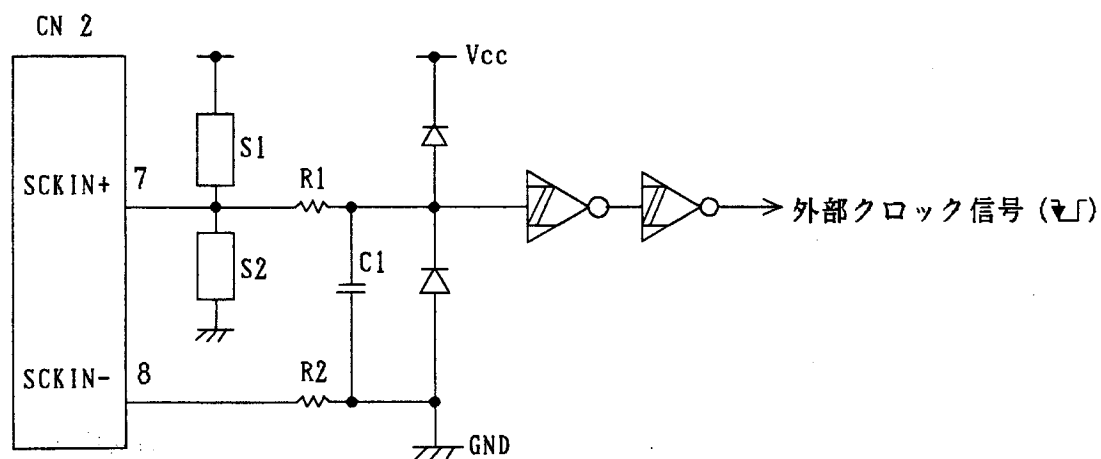
フロント側コネクタ (CN 2) の SCK IN+、SCK IN-間に入力されるクロック信号です。 (入力方法)

コネクタ	ピン番号	信号名	信号
CN 2	7	SCN IN+	外部クロック+側
	8	SCN IN-	外部クロック-側

電氣的仕様

入力レベル	立ち下りエッジ (↓) (↑)
パルス間隔 (min)	10 μs
パルス幅 (min)	50 ns
デレイタイム (max)	50 ns

外部クロック入力回路)



ターミネイトの抵抗値を変更の場合は、S 1、S 2をお取り換え下さい。又、出荷時は  
S 1 = 220  $\Omega$ 、S 2 = 330  $\Omega$ が実装されています。

又、ターミネイト抵抗値は本ボードをカスケード接続にて御使用の場合はマスタ及び最終  
スレーブボードのみ実装することを推奨致します。

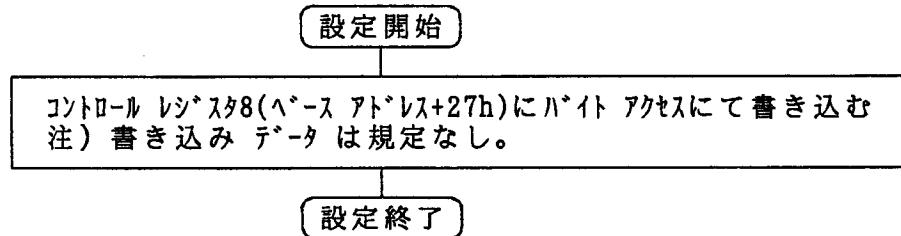


### 3-2-6 トリガ信号

トリガ信号には、内部トリガと外部トリガがあります。相方ともにコントロールレジスタ7を操作することで許可され有効となります。再トリガ時のトリガ入力モード1、2は毎回入力、モード3、4は強制停止実行後の再トリガ時のみ入力することになります。

#### ○ 内部トリガ

モード1、モード2で使用するトリガです。コントロールレジスタ8をバイトアクセスにて書き込みを行うことで内部トリガは発生します。



#### ○ 外部トリガ

モード3、モード4で使用するトリガです。フロント側コネクタ (CN2) の TRG IN+、TRG IN-間に入力される信号です。

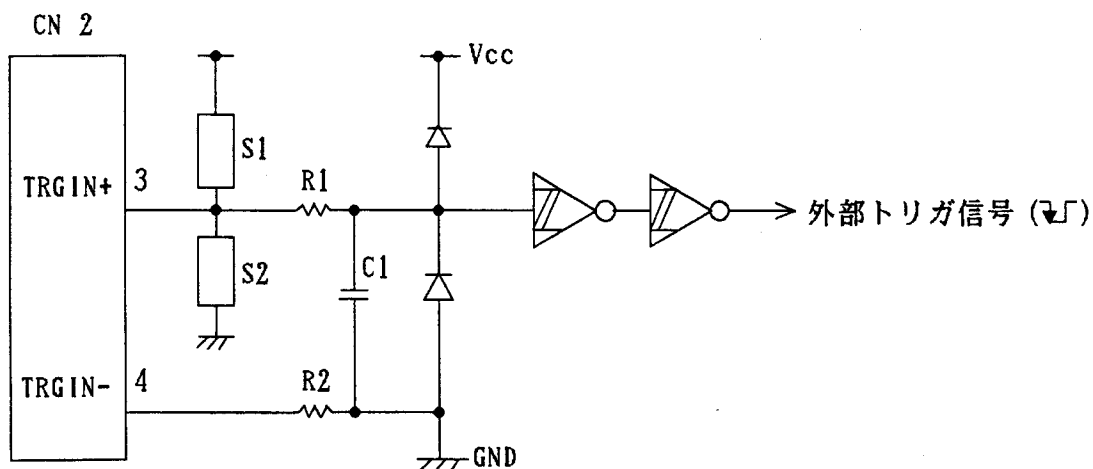
入力方法)

コネクタ	ピン番号	信号名	信号
CN 2	3	TRG IN+	外部トリガ +側
	4	TRG IN-	外部トリガ -側

電氣的仕様)

入力レベル	立ち下りエッジ (↓) (↑)
スレッシュホールドレベル	TTL レベル
パルス幅 (min)	50 [ns]
デレイタイム (max)	50 [ns]

外部トリガ入力回路)



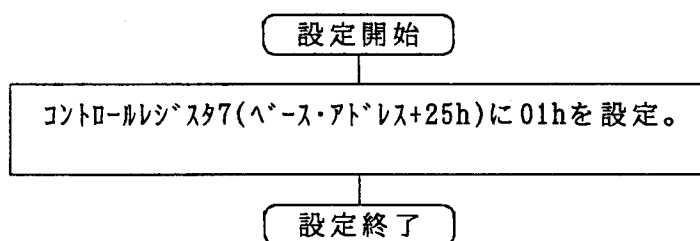
ターミネイトの抵抗値を変更の場合はS1、S2をお取り換え下さい。又、出荷時はS1=220Ω、S2=330Ωが実装されています。

○トリガ イネーブル/ディセーブル

内部トリガ及び外部トリガはコントロールレジスタ7のフォーマットに従い、イネーブルされて有効になります。

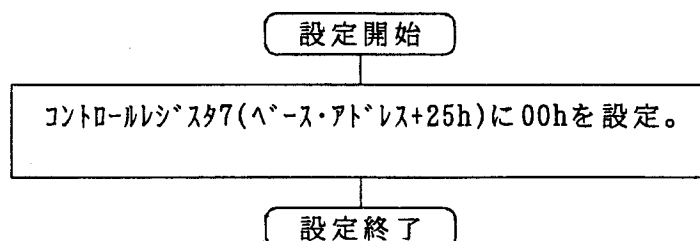
コントロールレジスタ7の設定はバイト・アクセスにて所定の値を書き込むことで行います。

① イネーブル設定



注)

② ディセーブル/変換中止設定



注) ディセーブル/変換中止設定の変換中止とは、トリガ信号の不許可に加えて、変換動作を強制的に停止できることを示します。

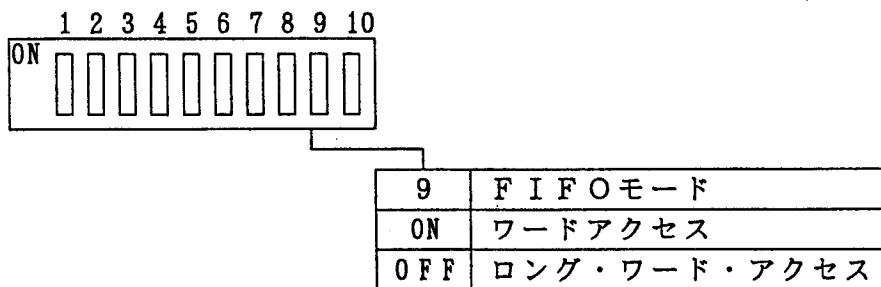
### 3-2-7 変換データの読み出し

サンプリングにて FIFO に蓄積された変換データは、データレジスタ 0 をワード・アクセス又はロング・ワード・アクセスにて読み出すことができます。

ただし、ワード又はロング・ワードのいずれか一方のアクセスに設定すること、及びデータについてチャンネル番号の有／無の設定が必要です。これらの設定はディップスイッチ (DP1) にて行います。

#### ○ FIFOモード設定

ディップスイッチ (DP1) の 9 にてフォーマットに従い設定して下さい。



●ワード設定時：① ワード・アクセスの読み出しのみデータ確定。

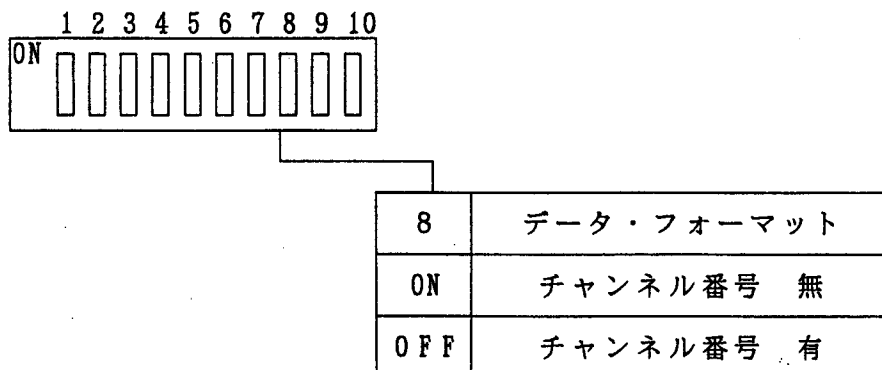
② FIFO 容量 8192ワード。

●ロング・ワード設定時：① ロング・ワードアクセスの読み出しのみデータ確認。

② FIFO 容量 4096ロング・ワード。

#### ○ データ・フォーマット設定

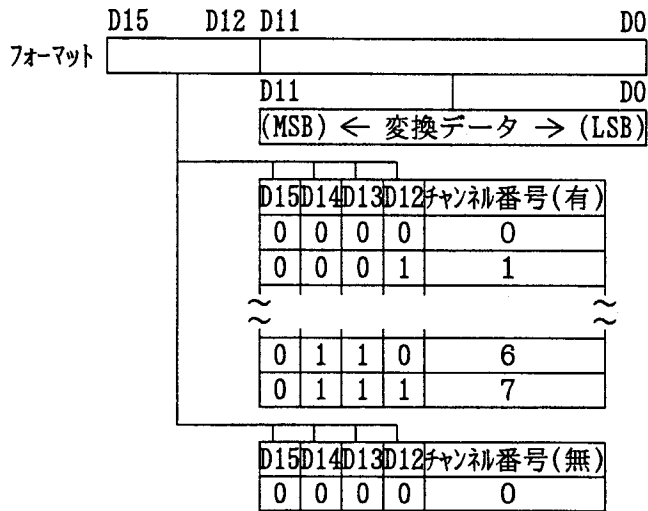
ディップスイッチ (DP1) の 8 にてフォーマットに従い設定して下さい。



○ データ レジスタ 0 データ・フォーマット

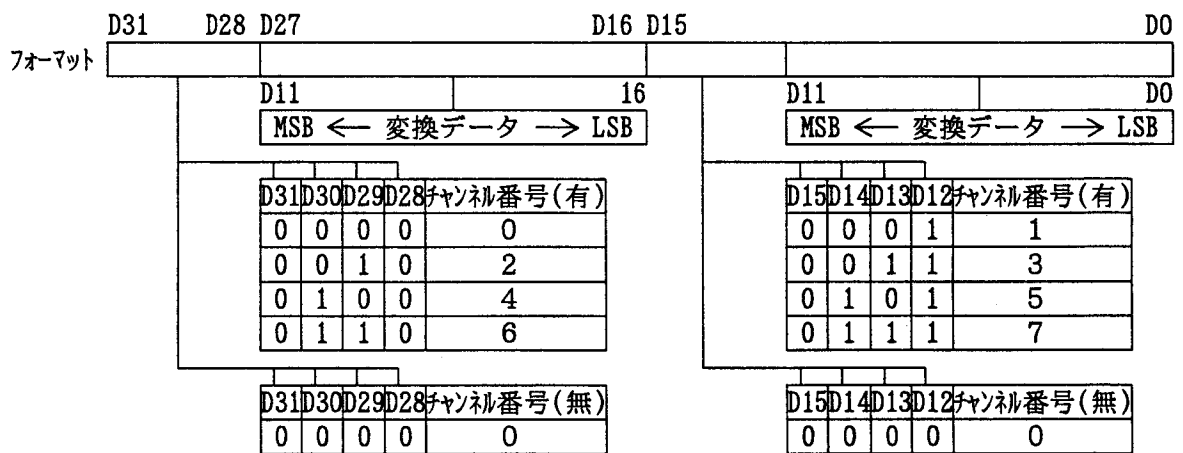
① ワード設定時

データ レジスタ 0 (ベースアドレス + 30h)



② ロング・ワード設定時

データ レジスタ 0 (ベースアドレス + 30h)



○ FIFO メモリ データ マップ

○ ワード設定時

D13	D02 D11	D0
CH 0	変換データ	
CH 1	//	
CH 2	//	
CH 3	//	
CH 4	//	
CH 5	//	
CH 6		
CH 7		

注) チャンネル番号(無)設定時 D15-D12 は" 0" 。

○ ロング・ワード設定時

D31 D28 D27	D16 D15 D12 D11	D0
CH 0	変換データ	CH 1 変換データ
CH 2	//	CH 3 //
CH 4	//	CH 5 //
CH 6	//	CH 7 //
CH 0		CH 1
CH 2		CH 3
CH 4		

注) チャンネル番号(無)設定時 D31-D28/D15-D12 は" 0" 。

### 3-2-8 マスタ/スレーブ構成

PVME-303 をマスタ/スレーブ構成にてカスケード接続することで多チャンネル同時サンプリングのアプリケーションに対応することができます。

(PVME-303のマスタ/スレーブ構成は VME bus アンキテクチャのアクセス・モードとは関係なく常にスレーブ・モードとしてアクセスされます。)

#### ○ 動作概要

マスタは設定されたA/Dモードに従いサンプリングを開始すると同時にスレーブへの同期信号を出力します。

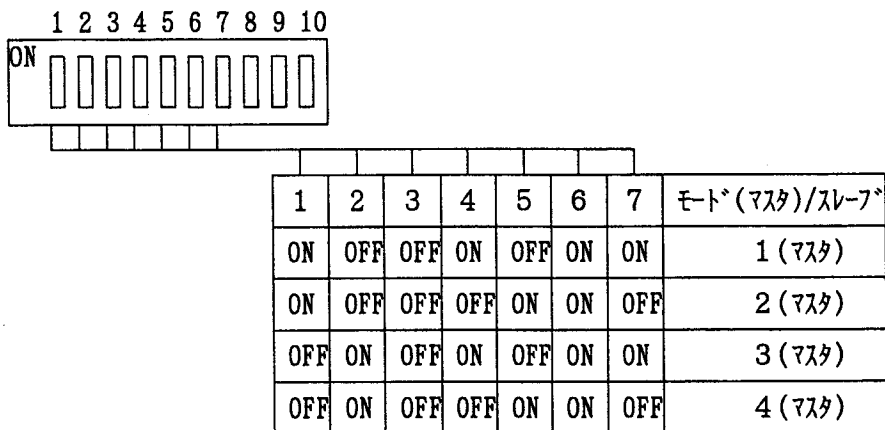
スレーブは、マスタからの同期信号に従いサンプリングを行います。

この時プログラム（ソフト）制御は、マスタ側にて各モードに対応した各レジスタにつき所定の設定を行い、スレーブ側はコントロールレジスタ 6 にてチャンネル設定を行うのみです。

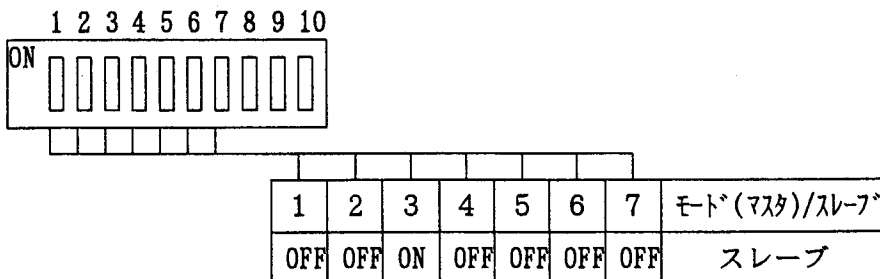
#### ○ マスタ/スレーブ設定

マスタ/スレーブ設定はディップスイッチ（DP1）にて行います。

マスタ

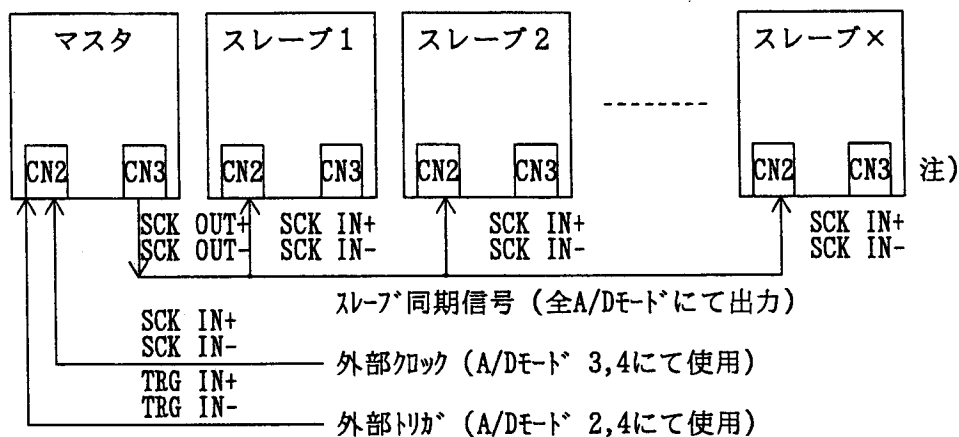


スレーブ



○ コネクタ接続

マスタ/スレーブ構成のコネクタ接続は図4-1の様にします。



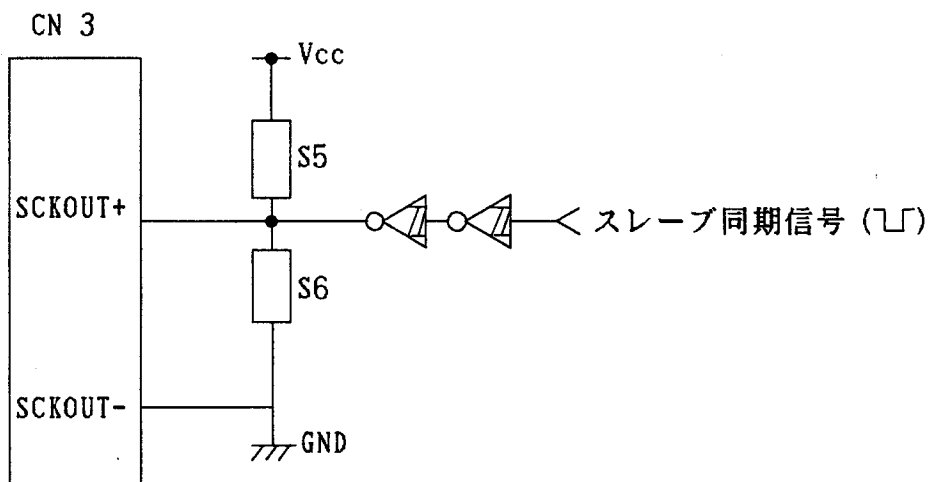
注) CN2 : フロントパネル側 CN2 コネクタ

CN3 : フロントパネル側 CN3 コネクタ

信号電氣的仕様

外部トリガ	コネクタ/+側(ピン)/-側(ピン)	CN2/TRG IN+(3)/TRG IN-(4)	
	入力レベル	TTLレベル 立ち下りエッジ (↓)	
	パルス幅 (min)	50 [ns]	
	ディレイタイム (max)	50 [ns]	
外部クロック	コネクタ/+側(ピン)/-側(ピン)	CN2/SCK IN+(7)/SCK IN-(8)	
	入力レベル	TTLレベル 立ち下りエッジ (↓)	
	パルス幅 (min)	50 [ns]	
	ディレイタイム (max)	50 [ns]	
同期信号	コネクタ/+側(ピン)/-側(ピン)	CN3/SCKOUT+(7)/SCKOUT-(8)	
	出力波形	パルス (□)	
	出力レベル	TTLレベル	
	出力電流	LOH (max)	-15 [mA]
		LOL (max)	48 [mA]
パルス幅 (min)	100 [ns]		

スレーブ同期信号出力回路)



ターミネイトの抵抗値を変更の場合は S1、S2 をお取り換え下さい。又、出荷時は  $S1=220\Omega$ 、 $S2=330\Omega$  が実装されています。

又、ターミネイト抵抗は本ボードをカスケード接続にて御使用の場合はマスタ及び最終スレーブボードのみ実装することを推奨致します。

注) SCK OUT+ の  $I_{OL}=48\text{mA}(\text{max})$  ですので、抵抗値変更の際は、ドライブ電流を十分考慮して下さい。



### 3-2-9 フィルタリング機能

各入力チャンネルに対応するコンデンサを外付けする事で、フィルタを構成する事ができます。フィルタリング用にアンプ (M9~16) のサミング点出力端子 (ピン5, 10) があります。内蔵コンデンサに並列に適合したコンデンサを外付けすると出力アンプの応答周波数を抑えることができます。

このことにより応答周波数を犠牲にしてノイズを低減させることができます。

コンデンサとセトリング/カットオフ周波数の関係は図1, 2又、各入力チャンネルに対応するコンデンサの部品番号は図3を参照して下さい。

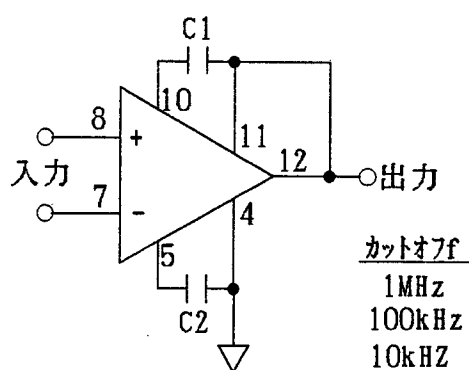
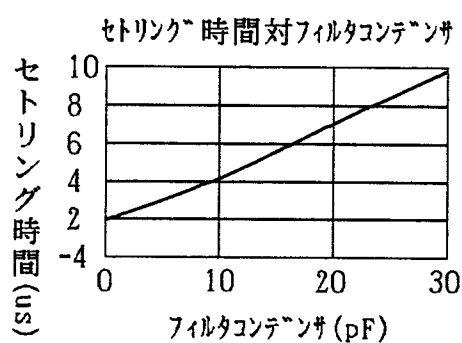


図1. セトリング時間(us) -

フィルタコンデンサ(pF)

図2. カットオフ周

波数 - フィルタコンデンサ(pf)

入力チャンネル	フィルタコンデンサ部品番号
0	C119, C120
1	C121, C122
2	C123, C124
3	C125, C126
4	C127, C128
5	C129, C130
6	C131, C132
7	C133, C134

図3. 入力チャンネル - フィルタコンデンサ部品番号

## 第4章 割り込み

#### 4-1 割り込み

##### ○ 概 要

割り込み動作は、ベクタ方式であり要求解除は、応答時開放 (ROAK) 又は、レジスタ・アクセス時開放 (ROBA) にて行われます。

割り込み条件設定はコントロール レジスタ 0～3 及びベクタ レジスタ 0～3 にて行います。

割り込み要因は、FIFO メモリの状態に対応した EF・HF・FF 及びサンプリング終了を知らせる DONE があります。

##### ○ 要 因

- ① E F 割り込み要因の EF は FIFO メモリにデータがある状態からすべてのデータを読み込み、空の状態になったときアサートされます。
- ② H F 割り込み要因の HF は FIFO メモリにデータが半分以上になったときアサートされます。
- ③ F F 割り込み要因の FF は FIFO メモリにデータがメモリ容量に達したときアサートされます。
- ④ D O N E サンプリングが変換回数に達したときアサートされます。

○ 要因とレジスタ

割り込み要因とレジスタの関係は次のようになります。

要因	レジスタ名	内容
E F	コントロール レジスタ 0	割り込み動作の制御
	ベクタ レジスタ 0	ベクタ番号 (8ビット)
H F	コントロール レジスタ 1	割り込み動作の制御
	ベクタ レジスタ 1	ベクタ番号 (8ビット)
F F	コントロール レジスタ 2	割り込み動作の制御
	ベクタ レジスタ 2	ベクタ番号 (8ビット)
D O N	コントロール レジスタ 3	割り込み動作の制御
	ベクタ レジスタ 3	ベクタ番号 (8ビット)

○ レジスタ設定

コントロール レジスタ 0～3 及びベクタ レジスタ 0～3 のフォーマットは次のようになります。

外部割り込みレベル								レジスタ名
F	FAC	XIN	IRE	IRAC	L2	L1	L0	コントロール レジスタ 0
F	FAC	XIN	IRE	IRAC	L2	L1	L0	コントロール レジスタ 0
F	FAC	XIN	IRE	IRAC	L2	L1	L0	コントロール レジスタ 0
F	FAC	XIN	IRE	IRAC	L2	L1	L0	コントロール レジスタ 0
V7	V6	V5	V4	V3	V2	V1	V0	ベクタ レジスタ 0
V7	V6	V5	V4	V3	V2	V1	V0	ベクタ レジスタ 0
V7	V6	V5	V4	V3	V2	V1	V0	ベクタ レジスタ 0
V7	V6	V5	V4	V3	V2	V1	V0	ベクタ レジスタ 0
7	6	5	4	3	2	1	0	レジスタ ビット

## 4-2 コントロール レジスタ

コントロール レジスタの各ビットについて説明致します。

- 各割り込みレベルは、L2・L1・L0（ビット2・1・0）にて設定します。  
L2・L1・L0にすべて0（ロー・レベル）を設定したときは、割り込み要求は不許可となります。  
又、レベルについては、表に従い設定を行います。

L 2	L 1	L 0	レ ベ ル
0	0	0	割り込み禁止
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

- 割り込み許可はIRE（ビット4）にて設定します。  
IREを1（ハイ・レベル）にすることで割り込み要求の許可が設定されます。  
このことから、割り込み要因がアサートしてもIREが0（ロー・レベル）に設定されていると、割り込み要求はアサートされません。

IRE	許可／不許可
0	不許可
1	許可

- 割り込み自動解除（応答時開放(ROAK)/レジスタ・アクセス時開放(RORA)設定)  
IRAC（ビット3）に1（ハイ・レベル）を設定したとき、割り込みアクノリッチ・サイクル中に、IRE（ビット4）は0（ロー・レベル）が設定される為、割り込み要求は解除されます。  
又、要因にて再度割り込み要求をアサートするには、IREに1（ハイ・レベル）を設定します

IRAC	自 動 解 除
0	OFF
1	ON

レジスタ・アクセス時開放(RORA)  
応答時開放(ROAK)

#### 4. 外部/内部ベクタ

割り込み動作では各ベクタレジスタの設定値が使用されますので X/IN (ビット5)、0 (ロー・レベル) に設定します。

X-IN	外部/内部ベクタ
0	内部ベクタ

#### 5. フラグ

フラグは、割り込み動作に影響されることなく書き込み/読み込みが行えます。フラグは、システム資源管理のステータス及びマルチタスクやマルチプロセッサのセマフォとして使用されます。又、フラグ (F) はビット7に配置され、MP U 68000 系の Test & Set (TAS) 命令により使用されます。

F	アサート/ネゲート
0	ネゲート
1	アサート

#### 6. フラグ自動解除

フラグ (F) と IRAC (ビット6) に1 (ハイ・レベル) が設置されているとき、割り込みアクノリッチ・サイクル中にフラグ (F) は解除されます。この特長は割り込みステータスやバス・メッセージに使用されます。

注) 各割り込み要因について同一レベルを設定した場合は、優先順位の高い方から処理されます。

DONE > FF > HF > EF  
高い ← 優先順位 → 低い

#### 4-3 ベクタ・レジスタ

各要因についてベクタ・レジスタ0~3にて8ビットのベクタ番号を設定します。

V7	V6	V5	V4	V3	V2	V1	V0	ビット
MSB							LSB	ベクタ番号

## 第5章 ポーリング

## 5-1 ポーリング

### 概要

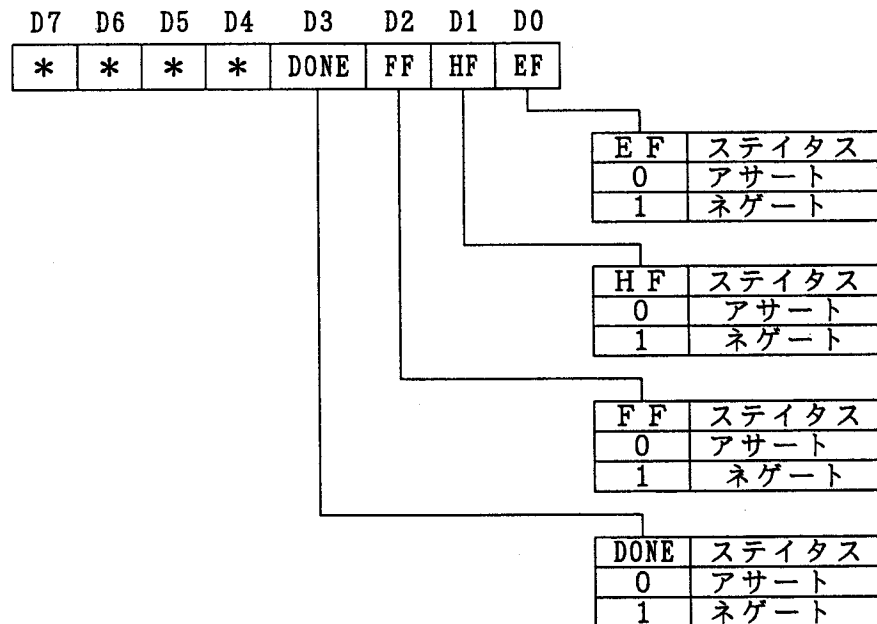
コントロール レジスタ10 を読み出すことで FIFO メモリの状態及びサンプリングの終了をステータスとして読み出すことができます。

### 要因

- ① EF FIFO メモリのデータが空の状態にアサートされます。
  - ② HF FIFO メモリのデータが半分以上の状態にアサートされます。
  - ③ FF FIFO メモリのデータが一杯の状態にアサートされます。
  - ④ DONE サンプリング終了時にアサートされます。
- ただし、DONE 要因は割り込み禁止にて行います。

### レジスタ・フォーマット

コントロール レジスタ10 (ベース・アドレス + 2Bh)



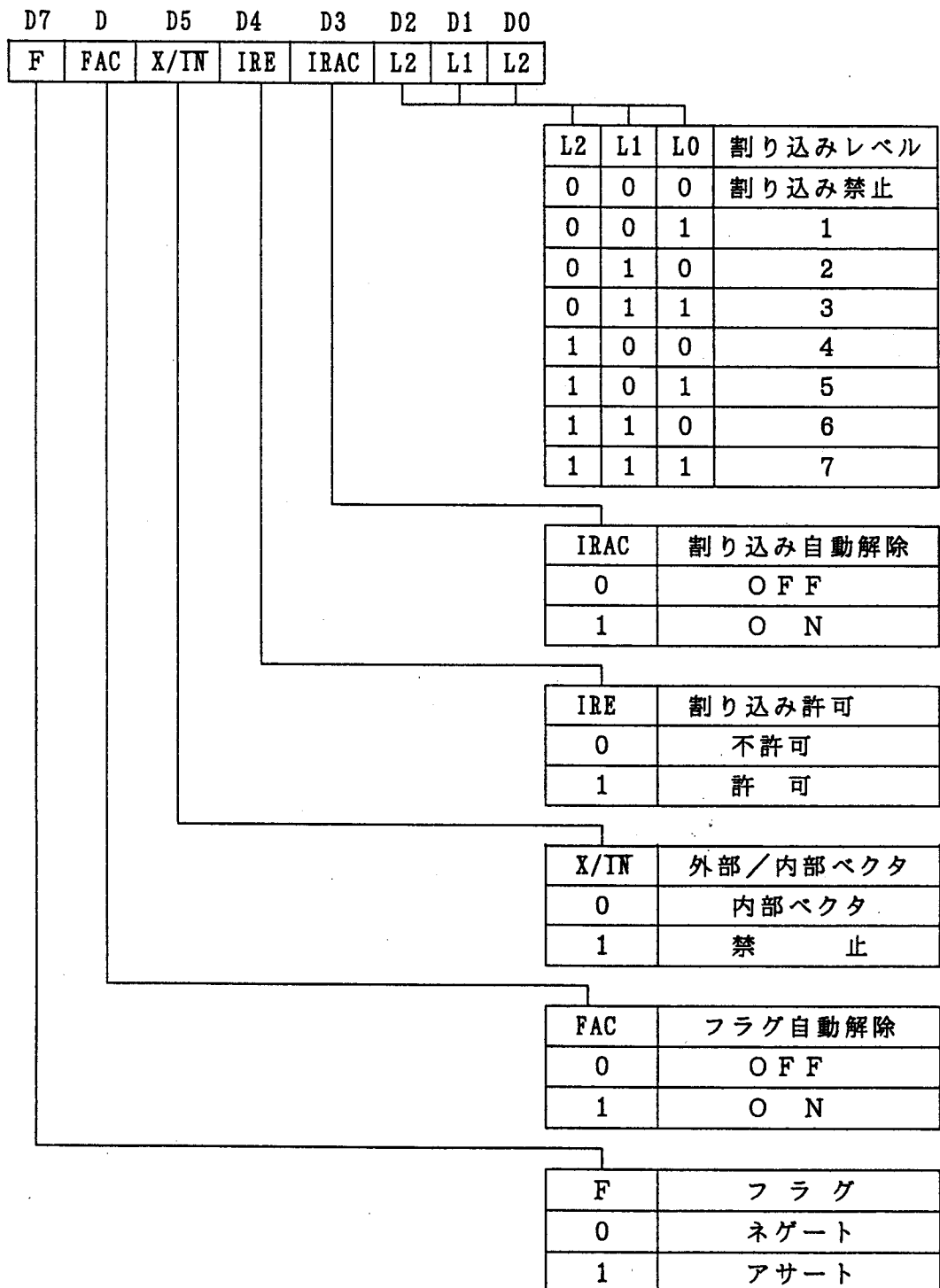


## 第 6 章 レジスタ・フォーマット

6-1 レジスタ・フォーマット

注)

- コントロール レジスタ 0 (ベース・アドレス + 01H) (バイト-リード / ライト)  
 割り込み要因 EF についての割り込み条件設定 レジスタ です。

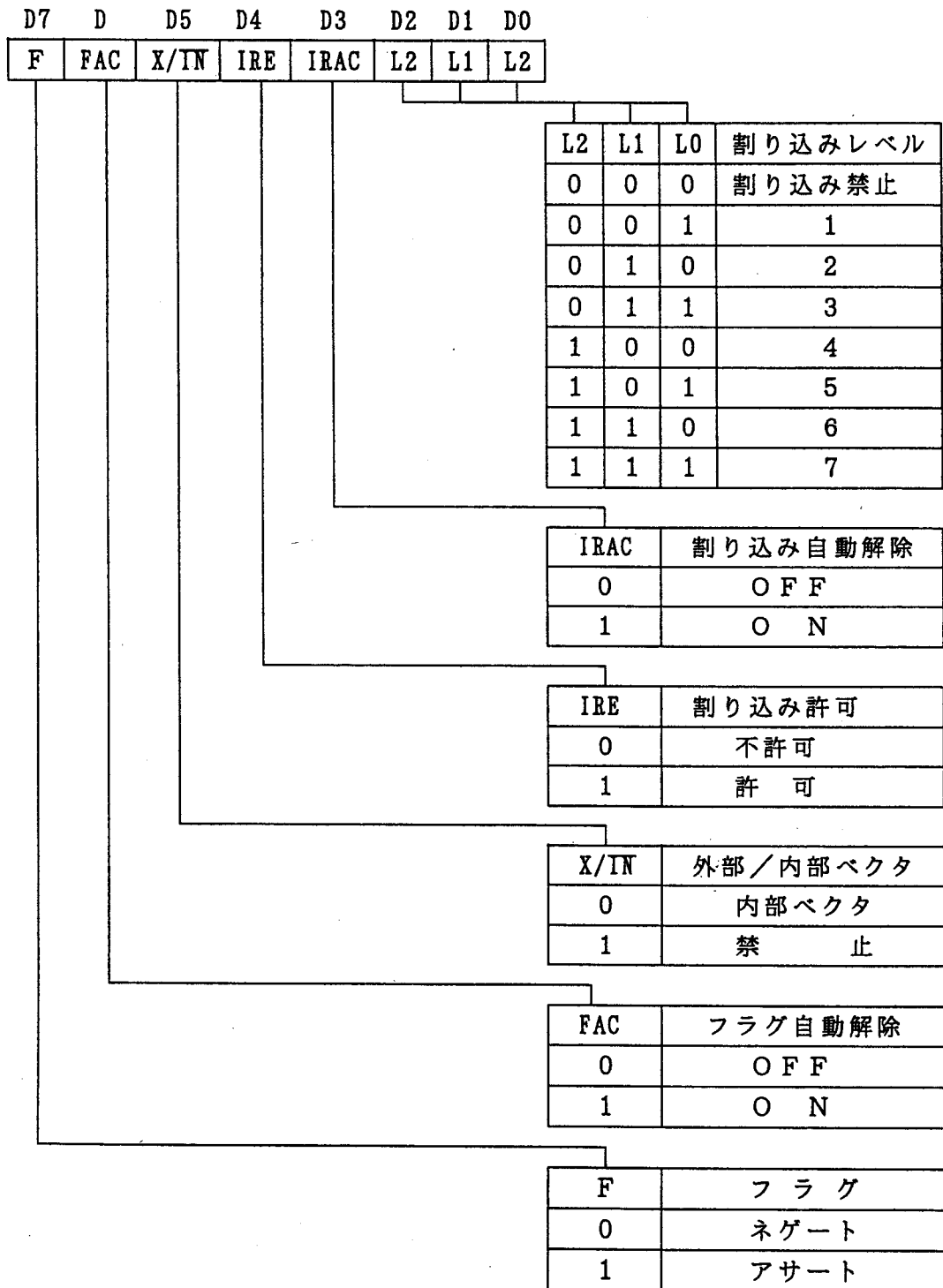


注) (例)バイト-リード/ライト: 右記の表現はそのレジスタはバイト・アクセスの  
 リード/ライトが行えることを示します。

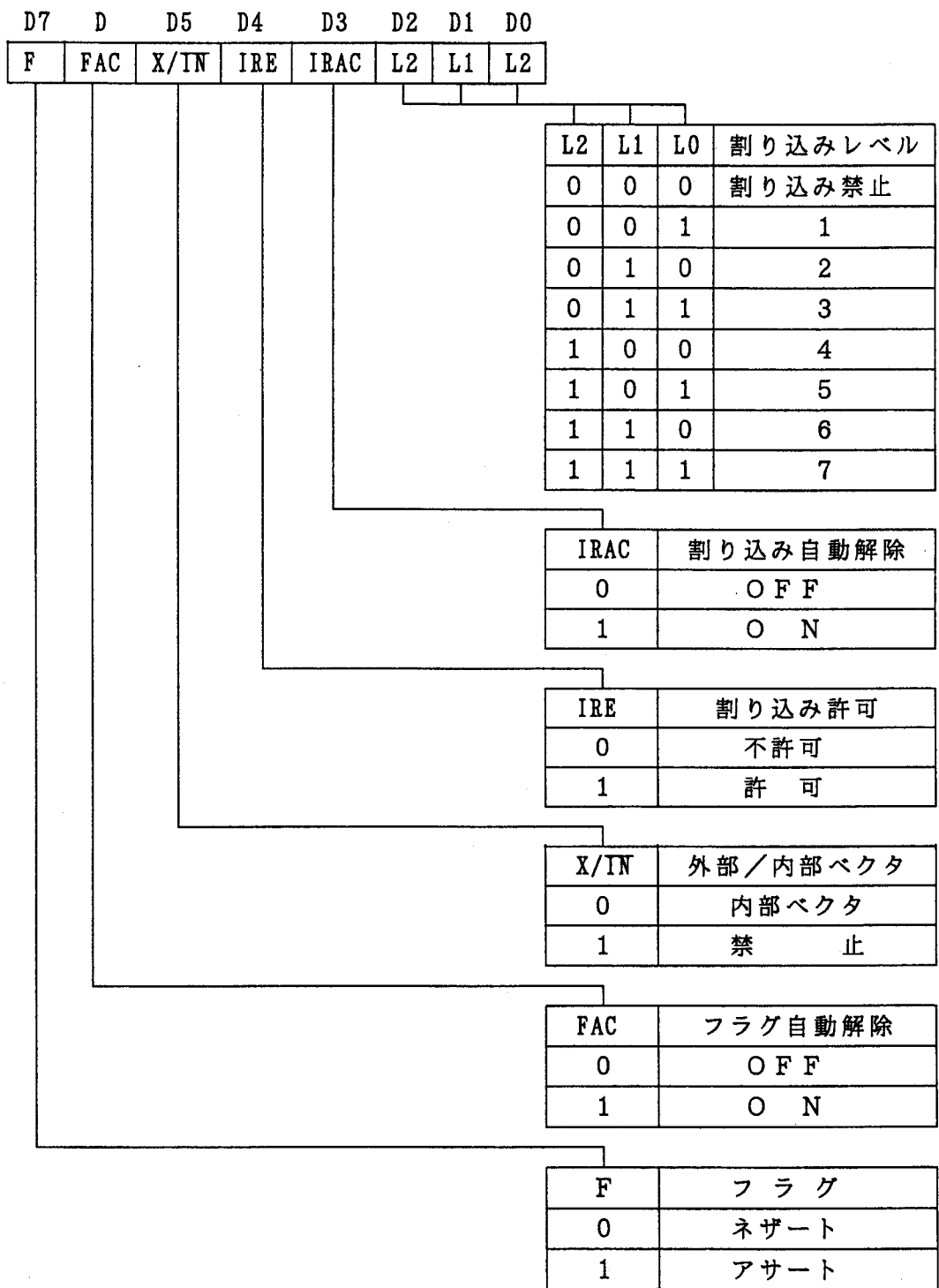
- コントロール レジスタ 1 (ベース・アドレス + 03H) (ハイト・リード / ライト)  
 割り込み要因 HF についての割り込み条件設定 レジスタ です。



- コントロール レジスタ 2 (ベース・アドレス + 05H) (バイト・リード / ライト)  
 割り込み要因 FF についての割り込み条件設定 レジスタ です。

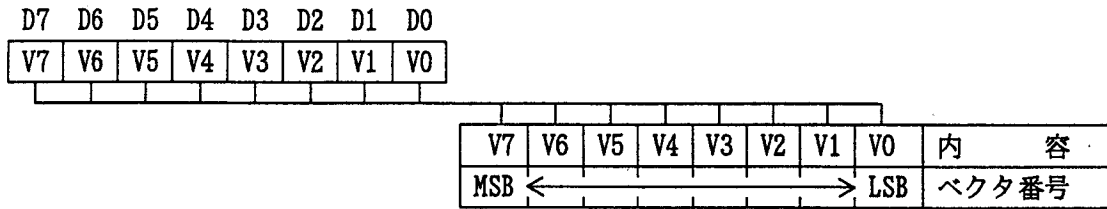


- コントロール レジスタ 3 (ベース・アドレス + 07H) (ライト-リード / ライト)  
 割り込み要因 DONE についての割り込み条件設定 レジスタ です。



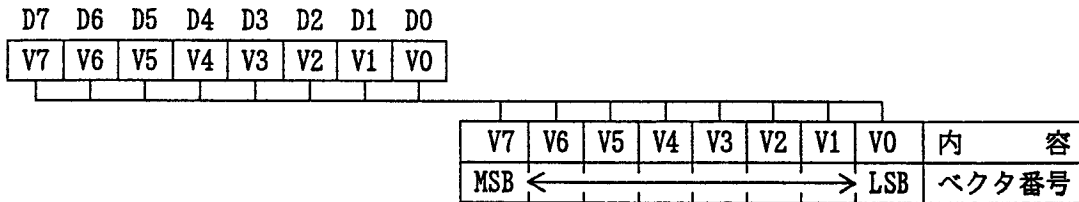
- ベクタ レジスタ 0 (ベース アドレス + 09 h) (バイト-リード/ライト)

割り込み要因 EF に対するベクタ番号の設定レジスタです。



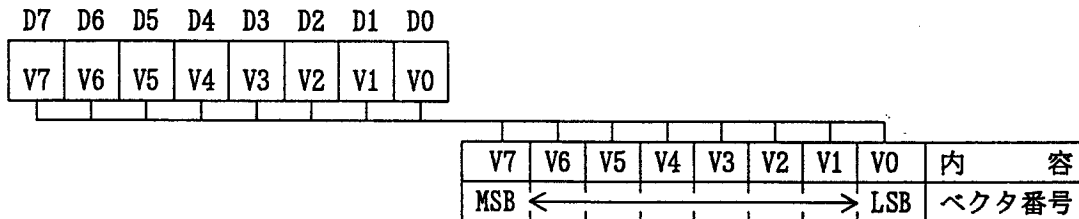
- ベクタ レジスタ 1 (ベース アドレス + 0B h) (バイト-リード/ライト)

割り込み要因 HF に対するベクタ番号の設定レジスタです。



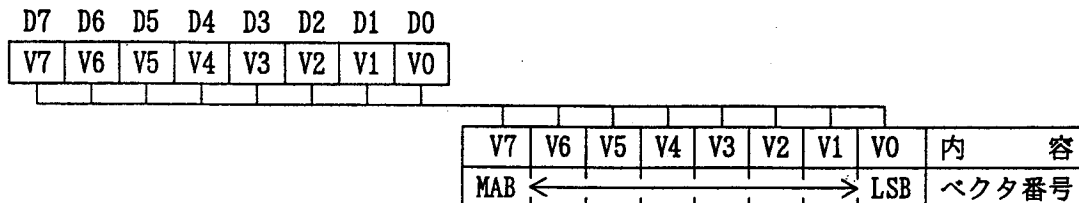
- ベクタ レジスタ 2 (ベース アドレス + 0D h) (バイト-リード/ライト)

割り込み要因 FF に対するベクタ番号の設定レジスタです。

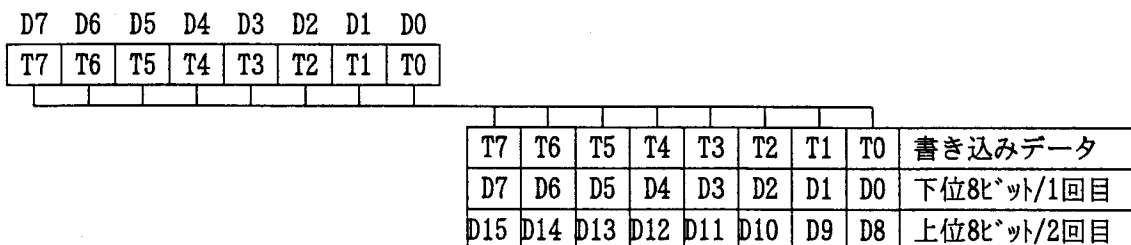


- ベクタ レジスタ 3 (ベース アドレス + 0F h) (バイト-リード/ライト)

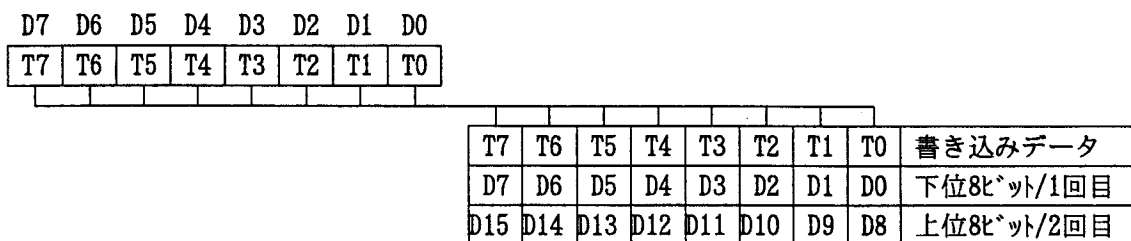
割り込み要因 DONE に対するベクタ番号の設定レジスタです。



- タイマー レジスタ 0 (ベース アドレス + 11h) (バイト-ライト)  
インターバル タイマ用タイマー1に16ビット表現のカウンタ値を下位バイト、上位バイトの順序にて設定するレジスタです。



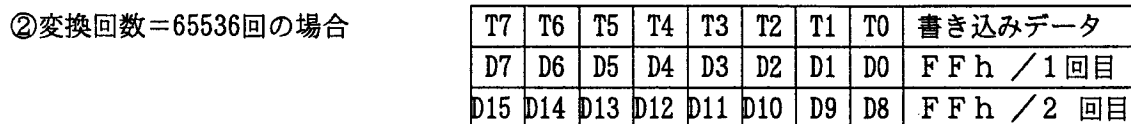
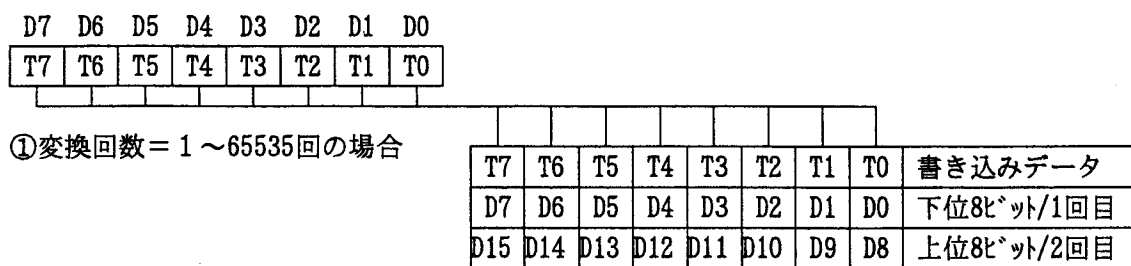
- タイマー レジスタ 1 (ベース アドレス + 13h) (バイト-ライト)  
インターバル タイマ用タイマー2に16ビット表現のカウンタ値を下位バイト、上位バイトの順序にて設定するレジスタです。



- カウンタ レジスタ 0 (ベース アドレス + 15h) (バイト-ライト)

変換回数用カウンタに16ビット表現のカウンタ値を下位バイト、上位バイトの順序にて設定するレジスタです。

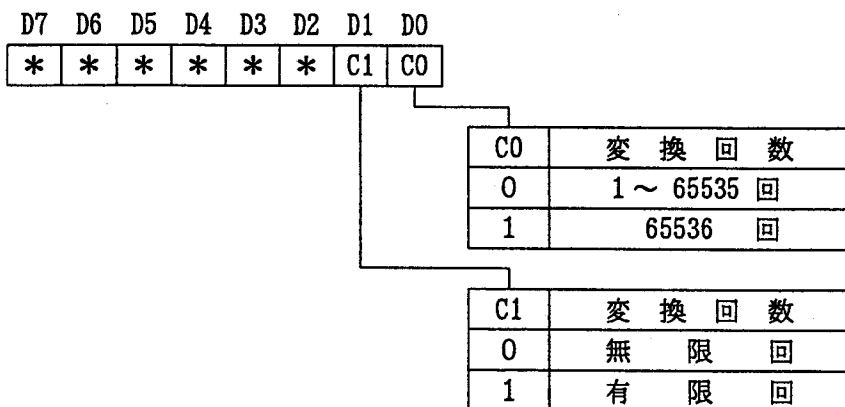
ただし、カウンタ レジスタ 0の設定はコントロール レジスタ 5と共に設定しますので変換回数設定シーケンスに従うようにして下さい。



- ③変換回数 = 無限回の場合     カウンタ レジスタ 0 設定不要。

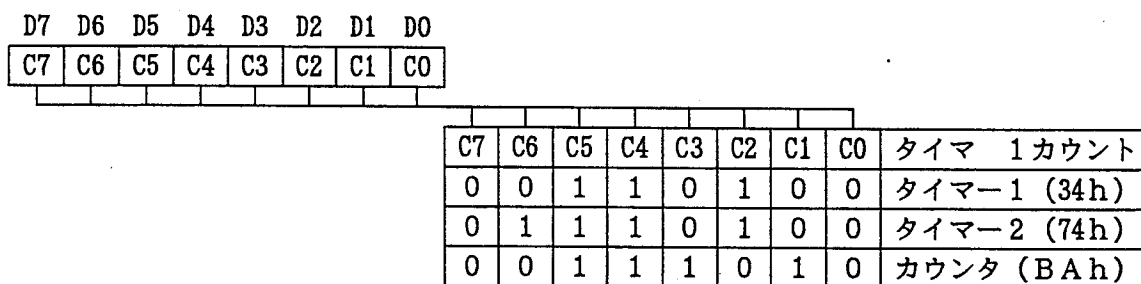
○ コントロール レジスタ 5 (ベース アドレス + 21h) (バイト-ライト)

変換回数用カウンタの設定を行う際に、カウンタ レジスタ 0 と共に設定を行うレジスタです。



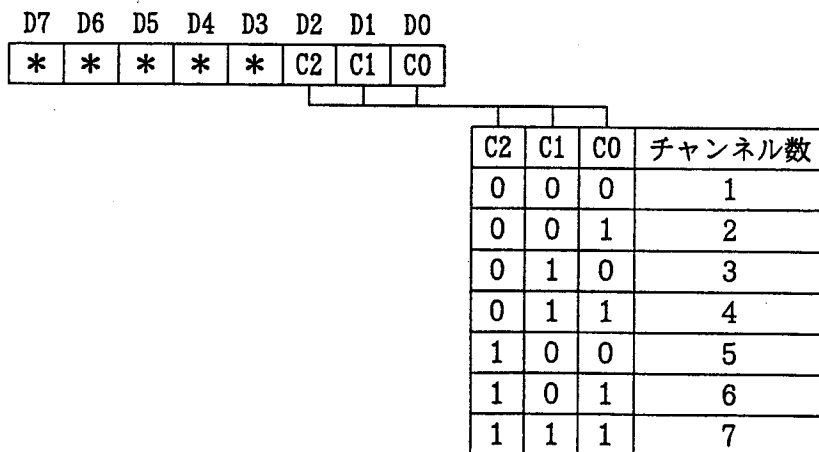
○ コントロール レジスタ 4 (ベース アドレス + 17h) (バイト-ライト)

インターバルタイマ用タイマー1、タイマー2及び変換回数用カウンタの初期設定を行うレジスタです。3つのコードをすべてコントロール レジスタ 4 に設定して下さい。又、カウンタ値変更時に再度設定する必要はありません。



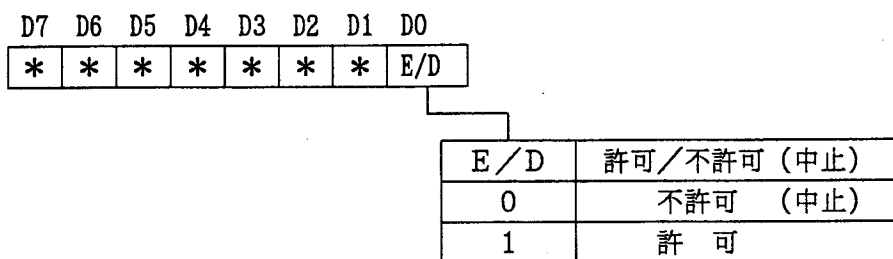
○ コントロール レジスタ 6 (ベース アドレス + 23h) (バイト-ライト)

本ボードにてサンプリングに使用するチャンネル数を設定するレジスタです。

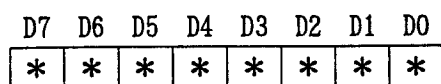




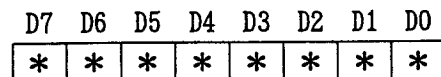
- コントロール レジスタ 7 (ベース アドレス +25h) (バイト-ライト)  
トリガ信号について許可/不許可の設定を行うレジスタです。又、サンプリング動作の中止設定も行うことができます。



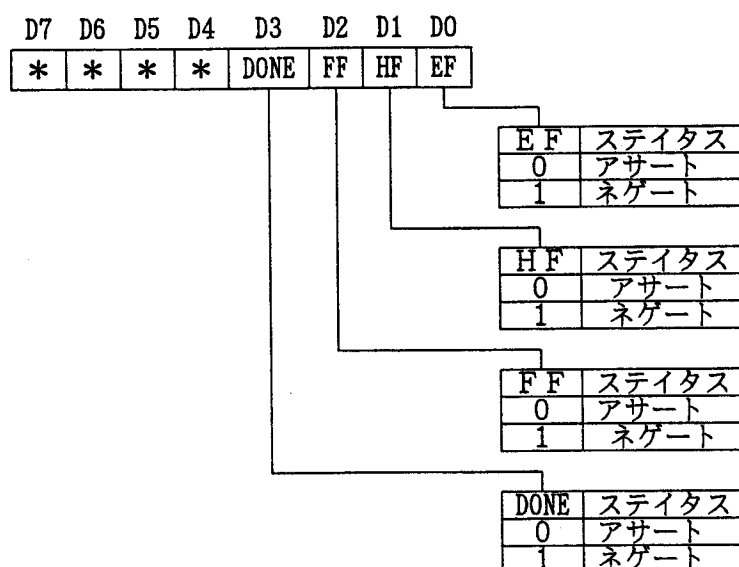
- コントロール レジスタ 8 (ベース アドレス +27h) (バイト-ライト)  
A/D変換モード1・2に使用する内部トリガ発生用レジスタです。バイト・アクセスの書き込み操作にて内部トリガが発生します。



- コントロール レジスタ 9 (ベース アドレス +29h) (バイト-ライト)  
FIFO メモリのリセットを設定するレジスタです。リセット後の FIFO メモリは各フラグ (EF・HF・FF) の状態、及び、内部データが初期化されます。設定は、バイト・アクセスの書き込みにて行われます。



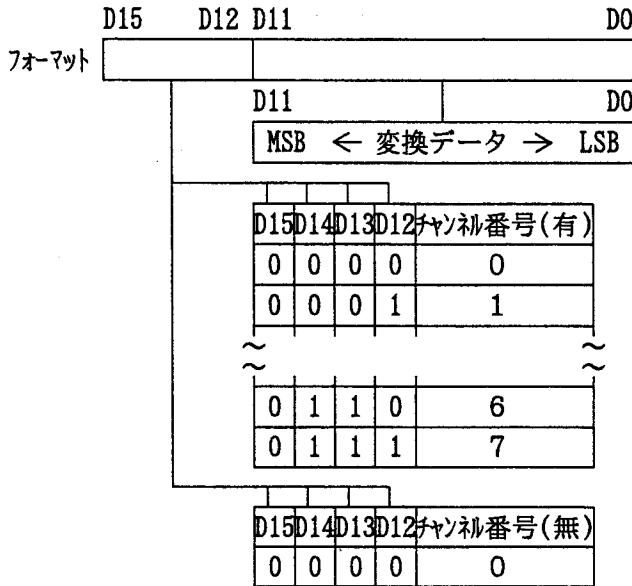
- コントロール レジスタ 10 (ベース アドレス +2Bh) (バイト-ライト)  
ポーリング ステータス参照用のレジスタです。バイト アクセスの読み込みにて参照が行えます。



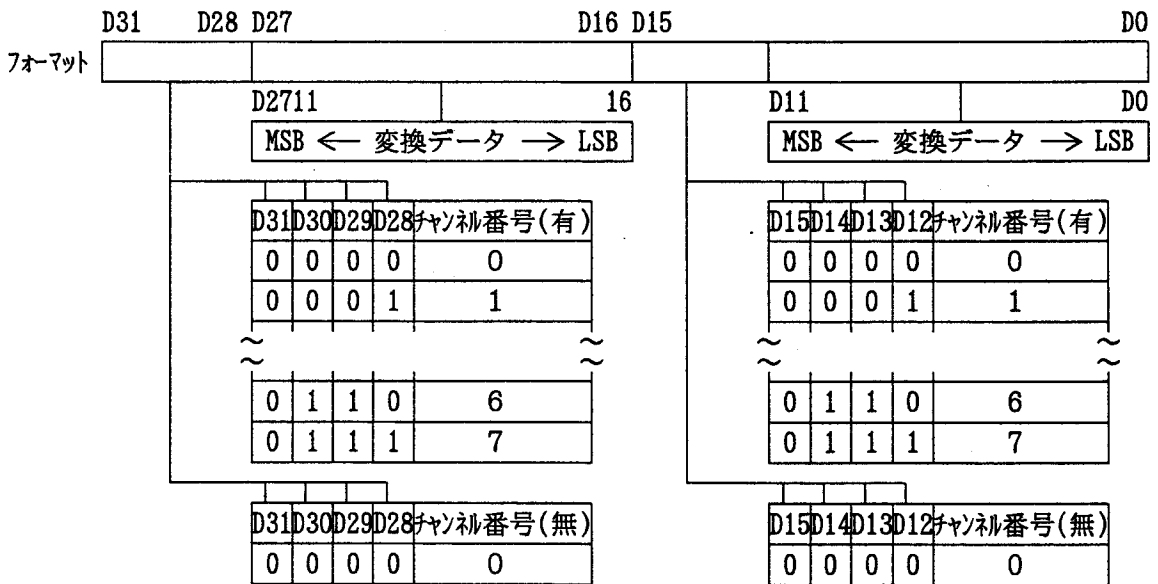
○ データ レジスタ 0 ( ベース アドレス +30h ) (ワード/ロングワード・リード)

FIFO メモリに蓄積されている変換データの参照用レジスタです。変換データにチャンネル番号が付加できる他、FIFO メモリ・リード・モードの設定 (ディップスイッチ DP1 の9ビット) により、ワード又は、ロングワードのいずれかのアクセスにて読み出すことができます。

① ワード設定時 (FIFO リード・モード)



② ロング・ワード設定時 (FIFO リード・モード)



## 第 7 章 ティップスイッチ/ジャンパー設定

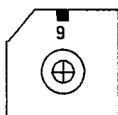
## 7-1 初期設定値

項目	初期値	部品番号
アドレスモード	OR (特権及び非特権) / 標準 (A24)	LD7
ベースアドレス	FC4B0000h	LD1-LD6
インターハルタイム	カスケード接続	JP17
チャンネル・ナンバー	有	DP1-8
FIFOアクセス	ワード	DP1-9
A/Dモード	モード1 (マスタ)	DP1-7
チャンネル 0	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP1, JP9 / SC7
チャンネル 1	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP2, JP10 / SC8
チャンネル 2	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP3, JP11 / SC9
チャンネル 3	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP4, JP12 / SC10
チャンネル 4	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP5, JP13 / SC11
チャンネル 5	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP6, JP14 / SC12
チャンネル 6	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP7, JP15 / SC13
チャンネル 7	ハイゾーラ ± 10.0[V]レンジ / 電圧入力	JP8, JP16 / SC14
SCK IN ターミネイト	SC1 : 220 Ω      SC2 : 330 Ω	SC1, SC2
SCK OUT ターミネイト	SC3 : 220 Ω      SC4 : 330 Ω	SC3, SC4
TRG IN ターミネイト	SC5 : 220 Ω      SC6 : 330 Ω	SC5, SC6

## 7-2 アドレス・モード設定

アドレス・モードはLD7にて設定します。

LD7



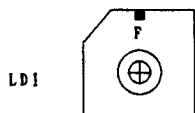
LD7	アドレス・モード
0	非特権 / 拡張 (A32)
1	非特権 / 標準 (A24)
2	非特権 / ショート (A16)
3	*
4	特権 / 拡張 (A32)
5	特権 / 標準 (A24)
6	特権 / ショート (A16)
7	*
8	OR / 拡張 (A32)
9	OR / 標準 (A24)
A	OR / ショート (A16)
B	*
C	*
D	*
E	*
F	*

\* : 禁止

### 7-3 ベース・アドレス 設定

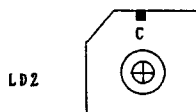
- ① アドレス A 3 1 - A 2 8 設定

LD1 にて Hex コードにより設定します。



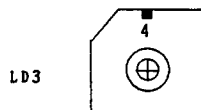
- ② アドレス A 2 7 - A 2 4 設定

LD2 にて Hex コードにより設定します。



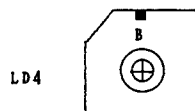
- ③ アドレス A 2 3 - A 2 0 設定

LD3 にて Hex コードにより設定します。



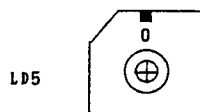
- ④ アドレス A 1 9 - A 1 6 設定

LD4 にて Hex コードにより設定します。



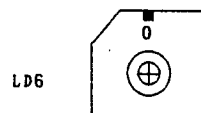
- ⑤ アドレス A 1 5 - A 1 2 設定

LD5 にて Hex コードにより設定します。



- ⑥ アドレス A 1 1 - A 8 設定

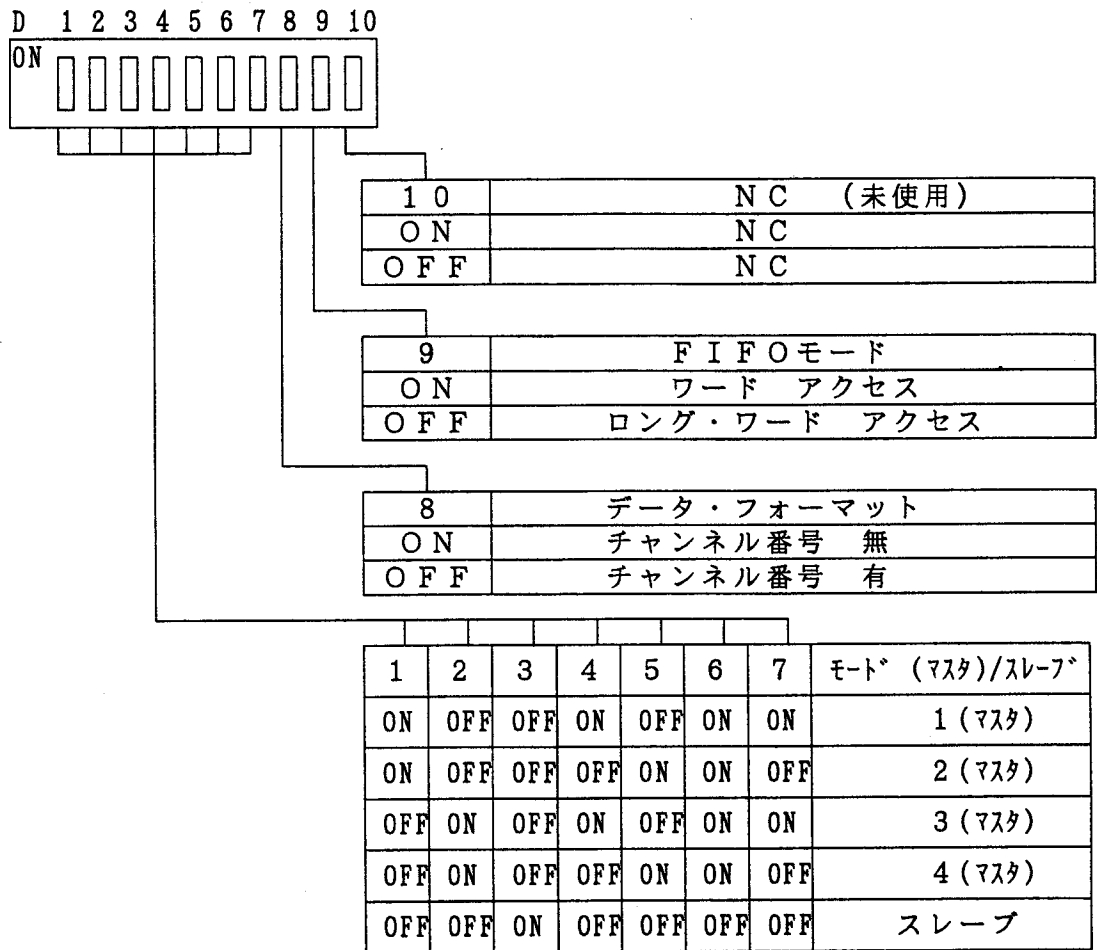
LD6 にて Hex コードにより設定します。



#### 7-4 モード/データ・フォーマット/FIFOモード設定

A/D変換モード、FIFOモード及びデータ・フォーマットの設定はディップ・スイッチ (DP1) にて行います。

DP1

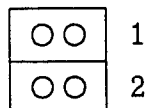


#### 7-5 インターバルタイマ単独/カスケード設定

インターバルタイマ用タイマー1及びタイマー2のカスケード接続設定が行えます。

ジャンパー

(JP25)



1	2	設定
●	○	タイマー1 単独
○	●	カスケード接続

## 7-6 入力レンジ設定

チャンネル0-7のユニポーラ/バイポーラ・レンジ設定はジャンパー (JP1-16) に行います。

各チャンネルとジャンパーの対応及びフォーマットは次の通りです。

チャンネル番号	ユニポーラ/バイポーラ	レンジ
0	JP 9	JP 1
1	JP 10	JP 2
2	JP 11	JP 3
3	JP 12	JP 4
4	JP 13	JP 5
5	JP 14	JP 6
6	JP 15	JP 7
7	JP 16	JP 8

### ユニポーラ/バイポーラ設定フォーマット

#### JP9-JP16共通

3 1

○	○
○	○

3	1	ユニポーラ/バイポーラ
●	○	ユニポーラ
○	●	バイポーラ

### レンジ設定フォーマット

7 5 3 1

○	○	○	○
○	○	○	○

(ユニポーラ時)

7	5	3	1	レンジ [V]
○	○	○	●	0 - 10.0
○	●	○	○	0 - 5.0
●	○	○	○	0 - 2.5

(バイポーラ時)

7	5	3	1	レンジ [V]
○	○	●	○	±10.0
○	○	○	●	± 5.0
○	●	○	○	± 2.5



## 7-7 電圧／電流入力設定

チャンネル0～7について、電圧又は、電流入力の設定を SC7～SC14 にて行います。

又、チャンネルと部品番号の対応及び、設定方法は図の通りです。

チャンネル番号	部品番号	電圧入力設定	電流入力設定
0	SC 7	オープン	I/V 変換抵抗実装
1	SC 8	//	//
2	SC 9	//	//
3	SC 10	//	//
4	SC 11	//	//
5	SC 12	//	//
6	SC 13	//	//
7	SC 14	//	//

注) I/V変換抵抗は、2.54 ピッチのものを選び下さい。

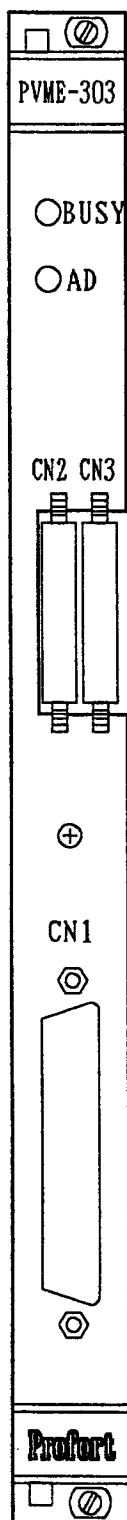
## 7-8 SCK IN / SCK OUT / TRG IN ライン ターミネイト抵抗値設定

SCK IN / SCK OUT / TRG IN ライン ターミネイト抵抗値は対応する部品番号に、任意の抵抗値を実装することで設定できます。ただし、ターミネイト抵抗はあらかじめ初期値により実装されています。

信号ライン名	部品番号 (7°ルアップ側)	初期値	部品番号 (7°ルダウン側)	初期値
SCK IN	SC 1	220 (Ω)	SC 2	330 (Ω)
SCK OUT	SC 3	220 (Ω)	SC 4	330 (Ω)
TRG IN	SC 5	220 (Ω)	SC 6	330 (Ω)

## 第 8 章 ピン・アサイン

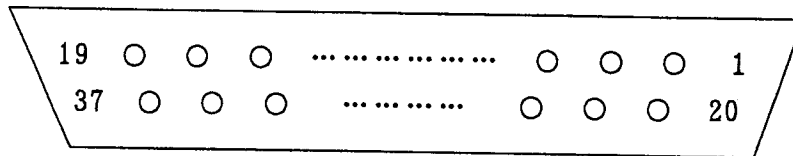
8-1 フロントパネル外観



- BUSY LED . . . . . ボードがスレーブ  
アクセス時点灯
- AD LED . . . . . A/D変換実行中点灯
- CN 2 . . . . . 外部コントロール信号入力コネクタ
- CN 3 . . . . . A/D変換 マスターボード時に  
コントロール信号をA/D変換  
スレーブボードに出力するコネクタ
- CN 1 . . . . . アナグロ信号入力コネクタ

8-2 アナログ入力コネクタ (CN 1)

- CN 1 (DSUB-37ピン) : 本体側 17LE-13370-27 (D4AB) (DDK製)  
: ケーブル側 17JE-23370-02 (D8A)



DSUBコネクタ (正面図)

○ ピン アサイン表

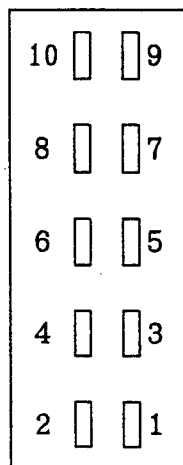
ピン番号	信号ニーマニック	ピン番号	信号ニーマニック
1	ch7 IN+	20	AGND
2	ch7 IN-	21	AGND
3	ch6 IN+	22	AGND
4	ch6 IN-	23	AGND
5	ch5 IN+	24	AGND
6	ch5 IN-	25	AGND
7	ch4 IN+	26	AGND
8	ch4 IN-	27	AGND
9	GND	28	AGND
10	ch3 IN+	29	AGND
11	ch3 IN-	30	AGND
12	ch2 IN+	31	AGND
13	ch2 IN-	32	AGND
14	ch1 IN+	33	AGND
15	ch1 IN-	34	AGND
16	ch0 IN+	35	AGND
17	ch0 IN-	36	AGND
18	AGND	37	AGND
19	AGND		

- 注) ch IN + : アナログ入力 (+側)
- ch IN - : アナログ入力 (-側)
- AGND : コモンランド (アナログ)
- ch X : チャンネル・ナンバ表記

8-3 同期信号コネクタ (CN2・CN3)

○ 同期信号入力コネクタ (CN2)

CN2 (リボンコネクタ 10ピン) : 本体側 FCN-705F010-AU/M (富士通)  
 : ケーブル側 FCN-707B010-AU/O



リボンコネクタ (正面図)

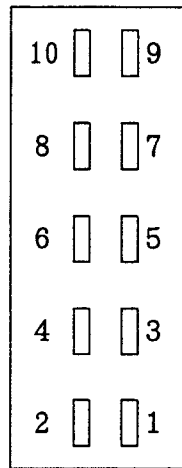
ピン番号	信号ニーモニック	ピン番号	信号ニーモニック
1	DGND	2	DGND
3	TRG IN+	4	TRG IN-
5	DGND	6	DGND
7	SCK IN+	8	SCK IN-
9	DGND	10	DGND

ピン アサイン表

- 注) TRG IN + : 外部トリガ入力 (+側)  
 TRG IN - : 外部トリガ入力 (-側)  
 SCK IN + : 外部クロック入力 (+側)  
 SCK IN - : 外部クロック入力 (-側)  
 DGND : コモン グランド (デジタル)

○ 同期信号コネクタ (CN3)

CN3 (リボンコネクタ 10ピン) : 本体側 FCN-709F010-AU/MS (富士通)  
 : ケーブル側 FCN-707B010-AU/0



リボンコネクタ (正面図)

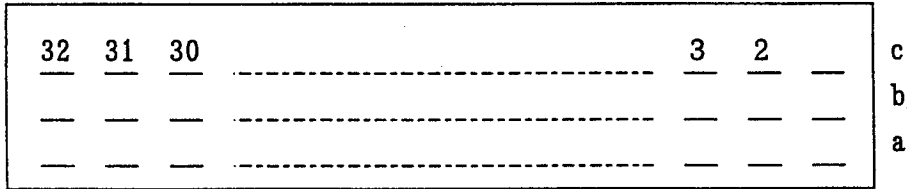
ピン番号	信号ニーマニック	ピン番号	信号ニーマニック
1	DGND	2	DGND
3	DGND	4	DGND
5	DGND	6	DGND
7	SCK OUT +	8	SCK OUT -
9	DGND	10	DGND

ピン アサイン表

注) SCK OUT + : スレーブ同期信号出力 (+側)  
 SCK OUT - : スレーブ同期信号出力 (-側)  
 DGND : コモン グランド (デジタル)

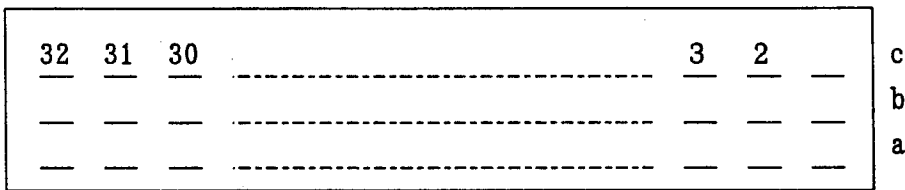
8-4 VME bus コネクタ (P1・P2)

- P1コネクタ (DIN コネクタ 96ピン) : 100-096-053 (PANDUIT製)



D I Nコネクタ (正面図)

- P2コネクタ (DIN コネクタ 96ピン) : 100-096-053 (PANDUIT製)



D I Nコネクタ (正面図)

○ P1コネクタ ピンアサイン表

ピン番号	列 a の信号ニーモニック	列 b の信号ニーモニック	列 c の信号ニーモニック
1	D00	BBSY*	D08
2	D01	BCLR*	D09
3	D02	ACFAIL*	D10
4	D03	BG0IN*	D11
5	D04	BG0OUT*	D12
6	D05	BG1IN*	D13
7	D06	BG1OUT*	D14
8	D07	BG2IN*	D15
9	GND	BG2OUT*	GND
10	SYSCLK	BG3IN*	SYSFAIL*
11	GND	BG3OUT*	BERR*
12	DS1*	BR0*	SYSRESET*
13	DS0*	BR1*	LWORD*
14	WRITE*	BR2*	AM5
15	GND	BR3*	A23
16	DTACK*	AM0	A22
17	GND	AM1	A21
18	AS*	AM2	A20
19	GND	AM3	A19
20	IACK*	GND	A18
21	IACKIN*	SERCLK(1)	A17
22	IACKOUT*	SERDAT*(1)	A16
23	AM4	GND	A15
24	A07	IRQ7*	A14
25	A06	IRQ6*	A13
26	A05	IRQ5*	A12
27	A04	IRQ4*	A11
28	A03	IRQ3*	A10
29	A02	IRQ2*	A09
30	A01	IRQ1*	A08
31	-12V	+5V STDBY	+12V
32	+5V	+5V	+5V



○ P2コネクタ ピンアサイン表

ピン番号	列 a の信号ニーモニック	列 b の信号ニーモニック	列 c の信号ニーモニック
1	使用者定義用	+5V	使用者定義用
2	使用者定義用	GND	使用者定義用
3	使用者定義用	保留	使用者定義用
4	使用者定義用	A24	使用者定義用
5	使用者定義用	A25	使用者定義用
6	使用者定義用	A26	使用者定義用
7	使用者定義用	A27	使用者定義用
8	使用者定義用	A28	使用者定義用
9	使用者定義用	A29	使用者定義用
10	使用者定義用	A30	使用者定義用
11	使用者定義用	A31	使用者定義用
12	使用者定義用	GND	使用者定義用
13	使用者定義用	+5V	使用者定義用
14	使用者定義用	D16	使用者定義用
15	使用者定義用	D17	使用者定義用
16	使用者定義用	D18	使用者定義用
17	使用者定義用	D19	使用者定義用
18	使用者定義用	D20	使用者定義用
19	使用者定義用	D21	使用者定義用
20	使用者定義用	D22	使用者定義用
21	使用者定義用	D23	使用者定義用
22	使用者定義用	GND	使用者定義用
23	使用者定義用	D24	使用者定義用
24	使用者定義用	D25	使用者定義用
25	使用者定義用	D26	使用者定義用
26	使用者定義用	D27	使用者定義用
27	使用者定義用	D28	使用者定義用
28	使用者定義用	D29	使用者定義用
29	使用者定義用	D30	使用者定義用
30	使用者定義用	D31	使用者定義用
31	使用者定義用	GND	使用者定義用
32	使用者定義用	+5V	使用者定義用

## 第 9 章 アナログ調整

## 9-1 調整方法

PVME-303はアナログ入力について6レンジが用意されておりますが、出荷時には±10[V]レンジにて正確に調整されています。

したがって、他のレンジにて御使用の際は所定のジャンパーにて希望のレンジを設定後、再度調整が必要となります。

調整については次の手順に従い行います。

### 1) ボードの各設定事項

項目	設定値	部品番号
アドレス モード	OR (特権及び非特権)	LD 7
ベース アドレス	FC4B0000h	LD 1-LD 6
インターバル タイマ	カスケード接続	JP 17
チャンネル・ナンバー	有	DP 1-8
FIFO アクセス	ワード	DP 1-9
A/D モード	モード1 (マスタ)	DP 1-7

\* 出荷時設定

### 2) 入力レンジ

チャンネル0-7のユニポーラ/バイポーラ・レンジ設定はジャンパー (JP1-16)にて行います。

各チャンネルとジャンパーの対応及びフォーマットは次の通りです。

チャンネル番号	ユニポーラ/バイポーラ	レンジ
0	JP 9	JP 1
1	JP 10	JP 2
2	JP 11	JP 3
3	JP 12	JP 4
4	JP 13	JP 5
5	JP 14	JP 6
6	JP 15	JP 7
7	JP 16	JP 8

ユニポーラ／バイポーラ設定フォーマット

JP9-JP16共通

3 1

○	○
○	○

3	1	ユニポーラ／バイポーラ
●	○	ユニポーラ
○	●	バイポーラ

レンジ設定フォーマット

7 5 3 1 (ユニポーラ時)

○	○	○	○
○	○	○	○

7	5	3	1	レンジ [V]
○	○	○	●	0 - 10.0
○	●	○	○	0 - 5.0
●	○	○	○	0 - 2.5

(バイポーラ時)

7	5	3	1	レンジ [V]
○	○	●	○	± 10.0
○	○	○	●	± 5.0
○	●	○	○	± 2.5

3) 調整プログラム (TEST 303 : C)

付随した調整プログラム (C言語記述) をシステムに入力後、コンパイルにて実行して下さい。

又、このときベース・アドレスを変換する場合はファイル (ADRS 303 : H) のベース・アドレス部を希望の値に変更願います。

4) 調整プログラム実行

調整プログラム実行後、モニタ上にチャンネル・ナンバ/変換データが次の様に表示されます。

各チャンネルの調整はこのモニタに表示されたデータを確認しながら所定のトリマを調整することで行います。

注) モニタはVT52モードに設定とします。

```

*****
* OFFSET & GAIN ADJUST *
*****

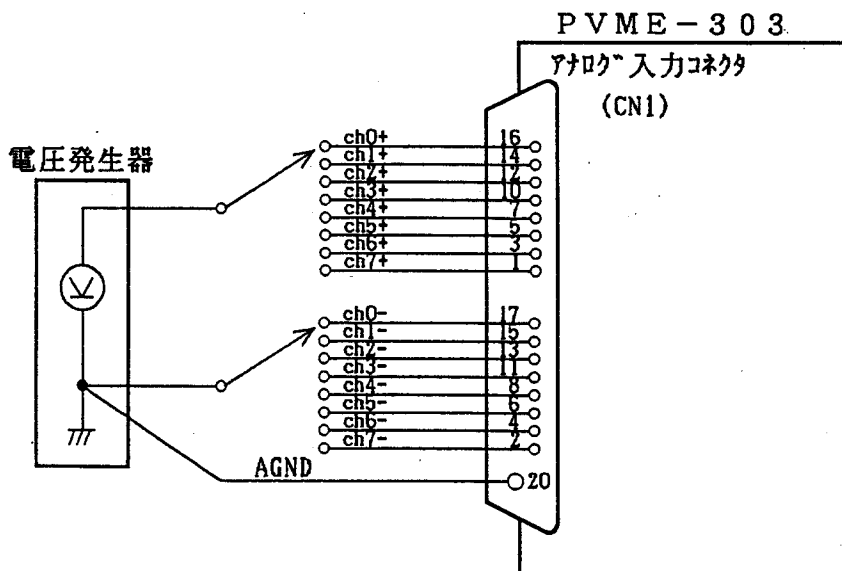
(CHANNEL) (A/D DATA)
ch:0      data:800
ch:1      data:801
ch:2      data:801
ch:3      data:800
ch:4      data:800
ch:5      data:800
ch:6      data:801
ch:7      data:801 変換データ (HEX表示)

*now CHO-7 ANALOG ADJUSTING*
  
```

モニタ表示例

5) 調整電圧

調整電圧はアナログ入力コネクタ (CN1) の各チャンネルに対応したピンに接続して下さい。



接続図

## 6) 調整

### ① オフセット調整

各レンジごとに対応した1/2LSB電圧を調整するチャンネルに入力後、モニタ上に表示されるA/D変換値が000~001にて半々に表示するようにチャンネルに対応したトリマを調整します。

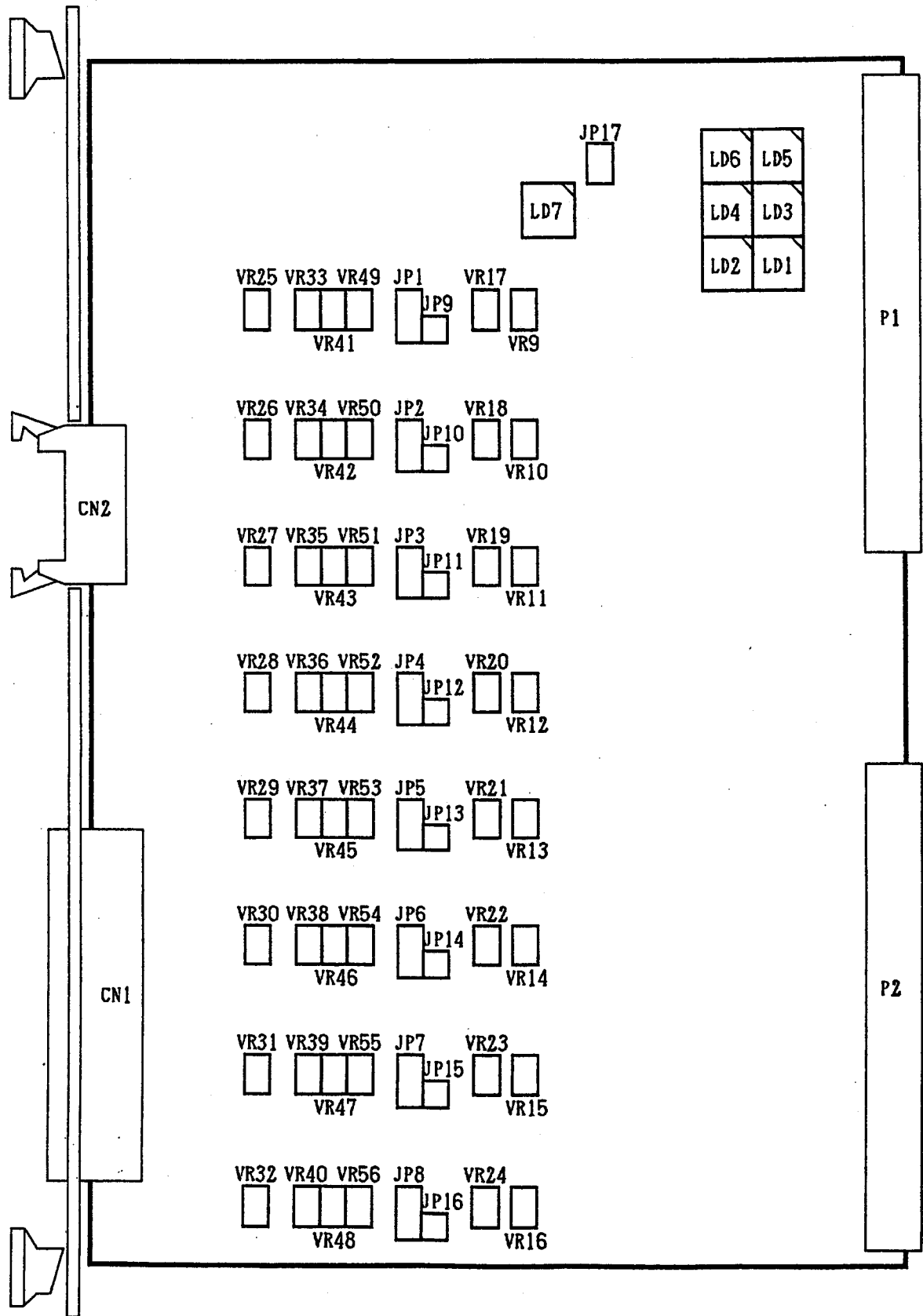
レンジ	1/2LSB電圧[mV]	トリマ (部品番号)							
		ch0	ch1	ch2	ch3	ch4	ch5	ch6	ch7
0-2.5[V]	0.30517	VR17	VR18	VR19	VR20	VR21	VR22	VR23	VR24
0-5.0[V]	0.61035								
0-10.0[V]	1.22070								
±2.5[V]	0.61035	VR9	VR10	VR11	VR12	VR13	VR14	VR15	VR16
±5.0[V]	1.22070								
±10.0[V]	2.44140								

### ② ゲイン調整

各レンジごとに対応するFSR (フルスケール) -1.5LSB電圧を調整するチャンネルに入力後、モニタ上に表示されるA/D変換値がFFE~FFFにて半々に表示するようにチャンネルに対応したトリマを調整します。

レンジ	FSR-1.5LSB 電圧[mV]	トリマ (部品番号)							
		ch0	ch1	ch2	ch3	ch4	ch5	ch6	ch7
0-2.5[V]	2.49908	VR25	VR26	VR27	VR28	VR29	VR30	VR31	VR32
0-5.0[V]	4.99816	VR33	VR34	VR35	VR36	VR37	VR38	VR39	VR40
0-10.0[V]	9.99633	VR41	VR42	VR43	VR44	VR45	VR46	VR47	VR48
±2.5[V]	2.49816	VR33	VR34	VR35	VR36	VR37	VR38	VR39	VR40
±5.0[V]	4.99633	VR41	VR42	VR43	VR44	VR45	VR46	VR47	VR48
±10.0[V]	9.99267	VR49	VR50	VR51	VR52	VR53	VR54	VR55	VR56

9-2 部品位置



9-3 調整プログラム (TEST303:C / ADRS303:H)

```

/*****
/*
/*          PVME-303 OFFSET/GAIN ADJUSTING PROGRAM  */
/*          PROGRAM NAME : TEST303:C              */
/*          Written By Internix.inc              */
/*****
/*-----*/
/* INCLUDE          */
/*-----*/
#include          <MAST303:H>      /* base fc4b0000 */
#include          <stdio.h>
/*-----*/
/* main            */
/*-----*/
main()
{
    init();                /* master initializ */
    clrclr();             /* CRT clear */
    tx0();                /* 'offset/gain adjust' */
    tx1();                /* 'a/d data' */
    while(1)
    {
        adjust();         /* offset/gain adjust */
    }
}
/*-----*/
/* initializ master */
/*-----*/
init()
{
    unsigned char *tr0,*tr1,*cnr,*cr4,*cr5,*cr6;
    tr0 = (unsigned char *) (TR0);
    tr1 = (unsigned char *) (TR1);
    cnr = (unsigned char *) (CNR);
    cr4 = (unsigned char *) (CR4);
    cr5 = (unsigned char *) (CR5);
    cr6 = (unsigned char *) (CR6);
    *cr4 = 0x34;          /* intervaltimer1 */
    *cr4 = 0x74;          /* intervaltimer2 */
    *cr4 = 0xba;          /* loopcounter */
    *tr0 = 0x08;          /* intervaltimer:1 us */
    *tr0 = 0x00;
    *tr1 = 0x0a;          /* intervaltimer:10 us */
    *tr1 = 0x00;
    *cnr = 0x01;          /* loopcounter:1 */
    *cnr = 0x00;
    *cr5 = 0x02;          /* yuugen */
    *cr6 = 0x07;          /* channel:8 */
}
/*-----*/
/* scanning start */
/*-----*/
getad()
{
    unsigned char *cr7,*cr8;
    cr7 = (unsigned char *) (CR7);
    cr8 = (unsigned char *) (CR8);
    ffrs();                /* fifo reset */
    *cr7 = 0x01;          /* triger enable */
    *cr8 = 0xff;          /* triger on */
}
/*-----*/
/* fifo reset */
/*-----*/
ffrs()
{
    unsigned char *cr9;
    cr9 = (unsigned char *) (CR9);
    *cr9 = 0xff;          /* fifo reset */
}
/*-----*/
/* offset/gain adjust */
/*-----*/
adjust()
{
    getad();                /* scanning start */
    done();                /* scan done ? */
    addt();                /* scan data display */
}
/*-----*/
/* scan done check */

```



```

/*-----*/
done()
{
  unsigned char *cr10;
  unsigned char s;
  unsigned char cmd0,cmd1,cmd2,cmd3;
  cr10 = (unsigned char *) (CR10);

  cmd0 = 0x1b;          /* ESC */
  cmd1 = 0x59;          /* Y */
  cmd2 = (15 + 0x20);   /* 15 line */
  cmd3 = (1 + 0x20);   /* 1 calam */

  for(s = 0x08;s == 0x08;)
  {
    s = (*cr10 & 0x08);
    printf("%c%c%c%c",cmd0,cmd1,cmd2,cmd3);
    printf("\n*now CHO-7 ANALOG ADJUSTING*");
  }
}
/*-----*/
/* scan data ch/data */
/*-----*/
addt()
{
  unsigned short *dr0;
  unsigned char i;
  unsigned char cmd0,cmd1,cmd2,cmd3,cmd4;
  unsigned short data,ch,a;

  cmd0 = 0x1b;          /* ESC */
  cmd1 = 0x59;          /* Y */
  cmd2 = (7 + 0x20);   /* 7 line */
  cmd3 = (4 + 0x20);   /* 4 calum */
  cmd4 = (15 + 0x20);  /* 15 calum */

  dr0 = (unsigned short *) (DR0);
  for(i = 0;i < 8;i++)
  {
    data = *dr0;
    ch = ((data & 0xf000)>>12);
    data = (data & 0x0fff);
    printf("%c%c%c%c",cmd0,cmd1,cmd2,cmd3);
    printf("%d",ch);
    printf("%c%c%c%c",cmd0,cmd1,cmd2,cmd4);
    printf("%3x",data);
    cmd2++;
  }
}
/*-----*/
/* crt clear */
/*-----*/
crtclr()
{
  unsigned char cmd0,cmd1,cmd2,cmd3;

  cmd0 = 0x1b;          /* ESC */
  cmd1 = 0x59;          /* Y */
  cmd2 = (0 + 0x20);   /* 0 line */
  cmd3 = (0 + 0x20);   /* 0 calam */

  printf("%c%c%c%c",cmd0,cmd1,cmd2,cmd3); /* 1 line/ 1 calam */

  cmd0 = 0x1b;          /* ESC */
  cmd1 = 0x4a;          /* J */

  printf("%c%c",cmd0,cmd1); /* CRT crear */
}
/*-----*/
/* messege 0,1 */
/*-----*/
tx0()
{
  printf("\n*****");
  printf("\n* OFFSET & GAIN ADJUST *");
  printf("\n*****");
}
tx1()
{
  printf("\n");
}

```

```
printf("\n(CHANNEL) (A/D DATA)");
printf("\n");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
printf("\nch:      data:");
}
/*****/
/* EOF */
/*****/
```

```

/*****/
/*
/*          PVME-303 ADDRESSING          */
/*          PROGRAM NAME : ADRS303:H     */
/*          Written By Internix.inc      */
/*****/

#define      BASE      0xfc4b0000      /* base adrs */
#define      CR0       BASE+0x01      /* irq ef */
#define      CR1       BASE+0x03      /* irq hf */
#define      CR2       BASE+0x05      /* irq ff */
#define      CR3       BASE+0x07      /* irq done */
#define      VR0       BASE+0x09      /* vct ef */
#define      VR1       BASE+0x0b      /* vct hf */
#define      VR2       BASE+0x0d      /* vct ff */
#define      VR3       BASE+0x0f      /* vct done */
#define      TR0       BASE+0x11      /* interval 1 */
#define      TR1       BASE+0x13      /* interval 2 */
#define      CNR       BASE+0x15      /* convert number */
#define      CR4       BASE+0x17      /* timer/counter set */
#define      CR5       BASE+0x21      /* convert number */
#define      CR6       BASE+0x23      /* ch number */
#define      CR7       BASE+0x25      /* triger e/d & a/d stop */
#define      CR8       BASE+0x27      /* internal triger */
#define      CR9       BASE+0x29      /* fifo reset */
#define      CR10      BASE+0x2b      /* poling status */
#define      DRO       BASE+0x30      /* fifo data */

/*****/
/* eof */
/*****/

```

## 第 10 章 サンプル・ソフト

## 10-1 ソフト概要

設定状態の条件にてチャンネル0～7を10 $\mu$ s間隔で同時測定を行い各チャンネルのデータをチャンネル番号とA/D変換値に分けてCRT上に表示。

## 10-2 設定状態

### 1)ハード設定

項目	初期値	部品番号
アドレスモード	0R (特権及び非特権) / 標準 (A24)	LD7
ベースアドレス	FC4B0000h	LD1-LD6
インターバルタイム	バス接続(インターバルタイム1のみ使用)	JP17
チャンネルナンバー	有	DP1-8
FIFOアクセス	ワード	DP1-9
A/Dモード	モード1 (マスタ)	DP1-7
チャンネル 0	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP1, JP9 / SC7
チャンネル 1	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP2, JP10 / SC8
チャンネル 2	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP3, JP11 / SC9
チャンネル 3	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP4, JP12 / SC10
チャンネル 4	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP5, JP13 / SC11
チャンネル 5	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP6, JP14 / SC12
チャンネル 6	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP7, JP15 / SC13
チャンネル 7	ハイゾラ ± 10.0[V]レンジ / 電圧入力	JP8, JP16 / SC14
SCK IN ターミネット	SC1 : 220 $\Omega$ SC2 : 330 $\Omega$	SC1, SC2
SCK OUT ターミネット	SC3 : 220 $\Omega$ SC4 : 330 $\Omega$	SC3, SC4
TRG IN ターミネット	SC5 : 220 $\Omega$ SC6 : 330 $\Omega$	SC5, SC6

### 2)ソフト設定

測定チャンネル数      チャンネル0～7 (8チャンネル)  
 インターバル時間      10 $\mu$ s  
 測定回数              2回 (16データ)

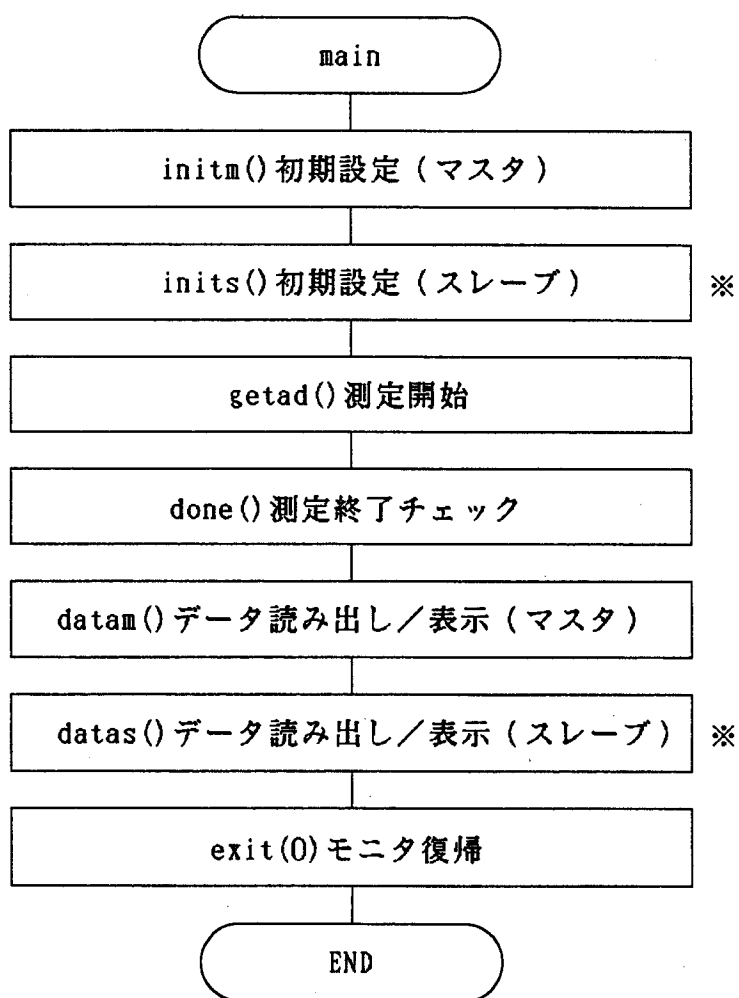
注) 以下に表記するサンプルプログラムは各モード共通に動作することのできるものです。

※印の関数の取扱いを各モードに対応するように実行して下さい。

### 10-3 プログラム構成

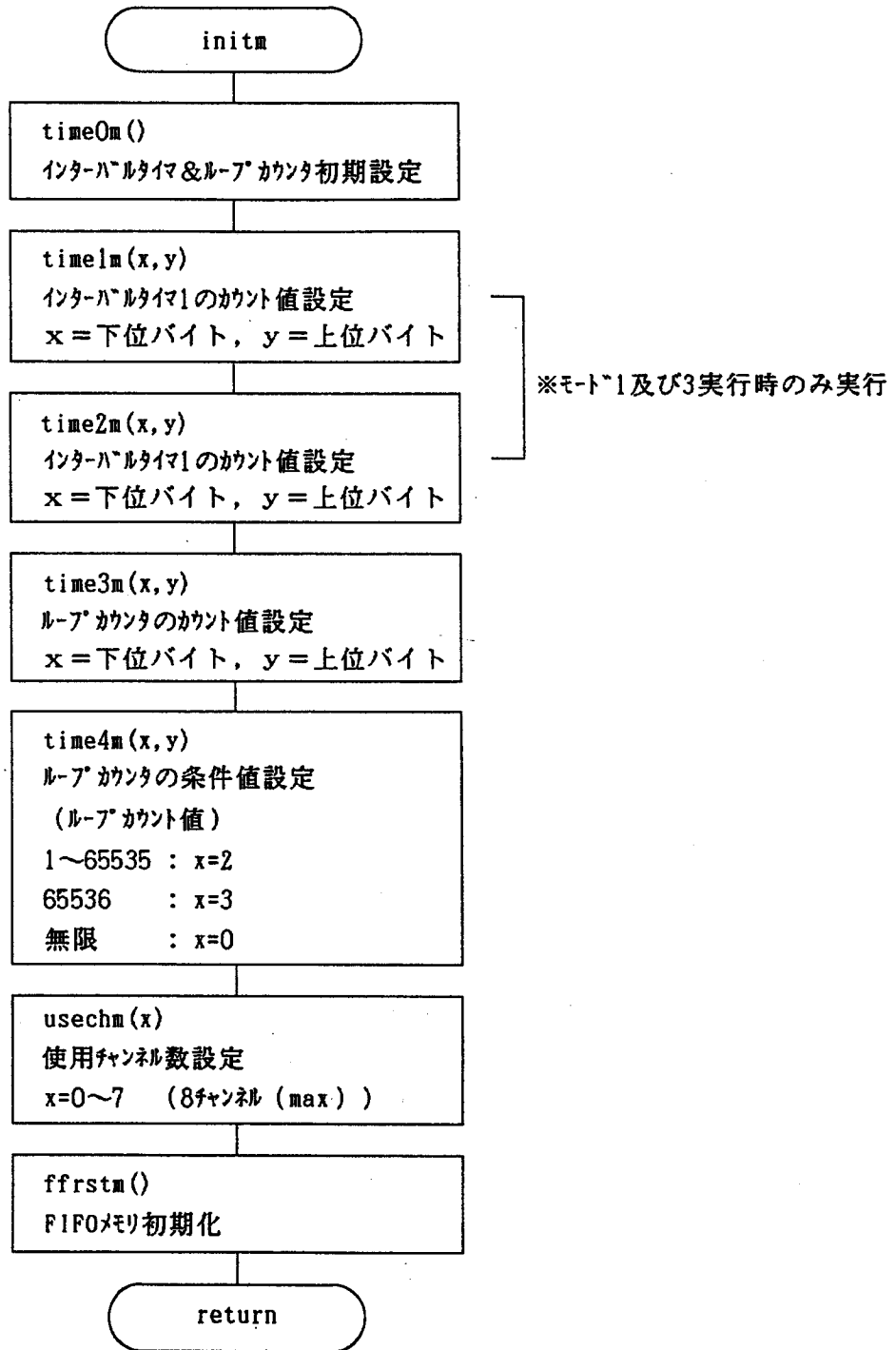
#### 1)関数

##### ①main()

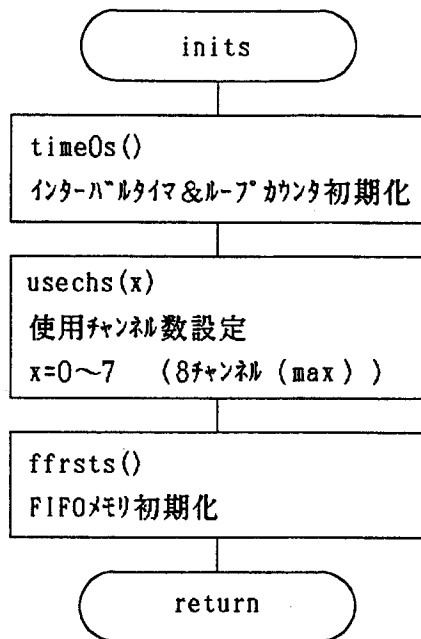


注) ※ : マスタ/スレーブ構成時のみ実行

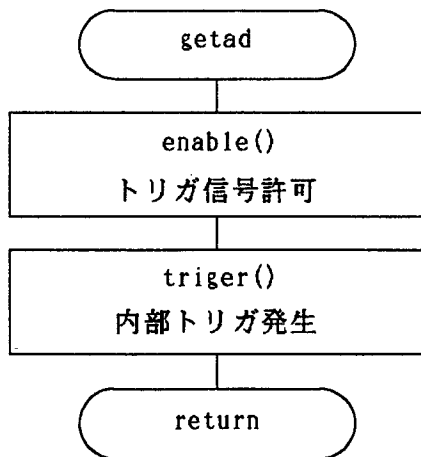
② `initm()`



③ `inits()` ※マスタ/スレーブ構成時のみ実行



④ `getad()`



※モード1及び2設定時のみ実行

注) モード1設定時

内部トリガ発生後、インターバルタイムに同期したサンプリングとなる。

モード2設定時

内部トリガ発生後、外部クロック信号に同期したサンプリングとなる。

モード3設定時

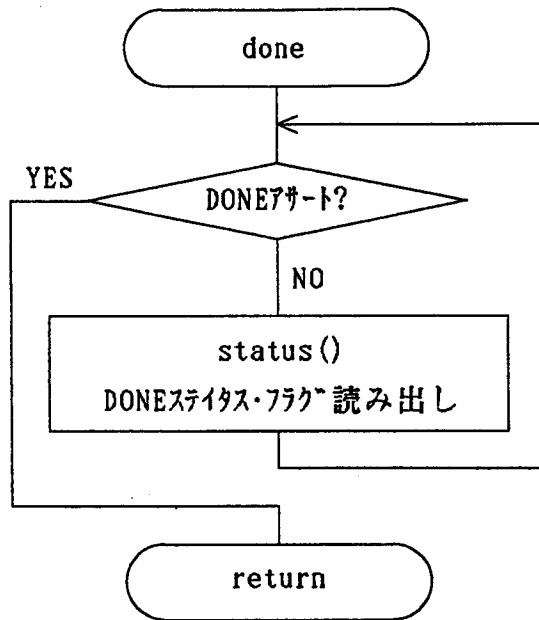
外部トリガ入力後、インターバルタイムに同期したサンプリングとなる。

モード4設定時

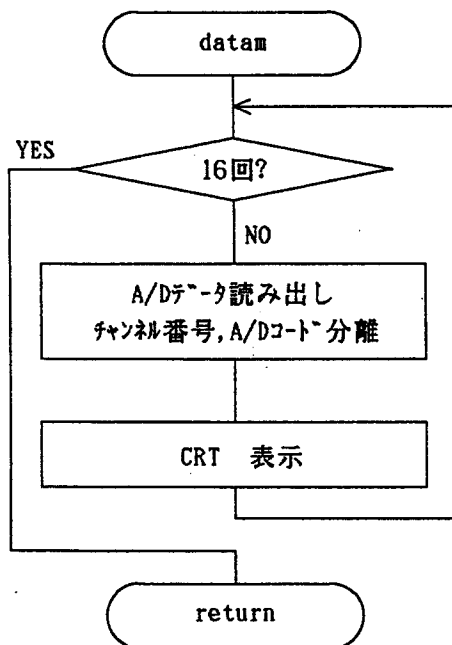
外部トリガ入力後、外部クロック信号に同期したサンプリングとなる。



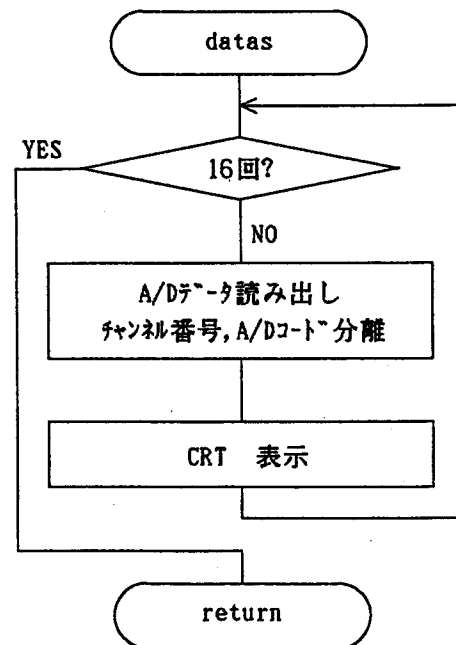
⑤ done ()



⑥ datam ()



⑦ datas () ※マスタ/スレーブ  
構成時のみ実行



```

/*****/
/*                                     */
/*   PVME-303 SAMPLE PROGRAM           */
/*                                     */
/*   measur channel: ch0-7             */
/*   intervaltime   : 10 us           */
/*   loop           : 2 (16 a/d data)  */
/*                                     */
/*   Written By Internix.inc '93/5/11 */
/*****/

/*-----*/
/* Include */
/*-----*/

#include <MAST303:H> /* PVME-303 mastre address */
#include <SLAV303:H> /* (master/slave) PVME-303 slave address */
#include <stdio.h>   /* standerd i/o header file */

/*-----*/
/* Main */
/*-----*/

main()
{
    initm(); /* master initializ */
    /* inits(); /* (master/slave) slave initializ */
    getad(); /* measuer start */
    done(); /* measuer end check */
    datam(); /* master measuer data */
    /* datas(); /* (master/slave) slave measuer data */
    exit(0); /* return to monitor */
}

/*-----*/
/* Initializ master */
/*-----*/

initm()
{
    unsigned char x,y;

    time0m(); /* intervaltimer1,2 & loopcounter initializ */
    x = 0x50; /* (model,3)intervaltimer1:low_byte/word */
    y = 0x00; /* (model,3)intervaltimer1:hi_byte/word */
    time1m(x,y); /* (model,3)intervaltimer1 set */
    x = 0x00; /* (model,3)intervaltimer2:low_byte/word */
    y = 0x00; /* (model,3)intervaltimer2:hi_byte/word */
    time2m(x,y); /* (model,3)intervaltimer2 set */
    x = 0x02; /* loopcounter:low byte/word */
    y = 0x00; /* loopcounter:hi_byte/word */
    time3m(x,y); /* loopcounter set */
    x = 0x02; /* loopcount:1_65535=2/65536=3/mugen=0 */
    time4m(x); /* loopcount set */
    x = 0x07; /* use channel number */
    usechm(x); /* use channel set */
    ffrstm(); /* fifo reset */
}

/*-----*/
/* (master/slave) Initializ slave */
/*-----*/

inits()
{
    unsigned char x,y;

    time0s(); /* intervaltimer1,2 & loopcounter initializ */
    x = 0x07; /* use channel number */
    usechs(x); /* use channel set */
    ffrsts(); /* fifo reset */
}

/*-----*/
/* Measuer start */
/*-----*/

getad()
{
    enable(); /* triger enable */
    triger(); /* (model,2) triger on */
}

```

```

/*-----*/
/* Measuer end */
/*-----*/

done()
{
  unsigned char x;

  for(x = 0x08;x == 0x08;) /* measuer end */
  {
    x = (status() & 0x08); /* read done status */
  }
}

/*-----*/
/* Master data */
/*-----*/

datam()
{
  unsigned short x,y,i;
  unsigned short *dr0;
  dr0 = (unsigned short *) (DR0);
  printf("%n *MASTER MEASUER DATA *");
  for(i = 0;i < 16;i++)
  {
    x = *dr0; /* read a/d data */
    y = (x & 0xf000)>>12; /* channel number */
    x = (x & 0x0fff); /* a/d code */
    printf("%n*CH = %2d:DATA = %3x",y,x);
  }
}

/*-----*/
/* (mstaer/slave) Slave data */
/*-----*/

datas()
{
  unsigned short x,y,i;
  unsigned short *dr0s;
  dr0s = (unsigned short *) (DR0S);
  printf("%n *SLAVE MEASUER DATA *");
  for(i = 0;i < 16;i++)
  {
    x = *dr0s; /* read a/d data */
    y = (x & 0xf000)>>12; /* channel number */
    x = (x & 0x0fff); /* a/d code */
    printf("%n*CH = %2d:DATA = %3x",y,x);
  }
}

/*=====*/
/* Call by initm() */
/*=====*/

time0m()
{
  unsigned char *cr4;
  cr4 = (unsigned char *) (CR4);

  *cr4 = 0x34; /* intervaltmer1 set */
  *cr4 = 0x74; /* intervaltmer2 set */
  *cr4 = 0xba; /* loopcounter set */
}

time1m(x,y)
  unsigned char x,y;
{
  unsigned char *tr0;
  tr0 = (unsigned char *) (TR0);
  *tr0 = x; /* intervaltmer1 low byte/word */
  *tr0 = y; /* intervaltmer1 hi byte/word */
}

time2m(x,y)
  unsigned char x,y;
{
  unsigned char *tr1;
  tr1 = (unsigned char *) (TR1);
  *tr1 = x; /* intervaltmer2 low byte/word */
  *tr1 = y; /* intervaltmer2 hi byte/word */
}

time3m(x,y)
  unsigned char x,y;

```

```

{
  unsigned char *cnr;
  cnr = (unsigned char *) (CNR);
  *cnr = x; /* loopcounter low byte/word */
  *cnr = y; /* loopcounter hi byte/word */
}
time4m(x)
  unsigned char x;
  {
    unsigned char *cr5;
    cr5 = (unsigned char *) (CR5);
    *cr5 = x; /* loopcount */
  }
usechm(x)
  unsigned char x;
  {
    unsigned char *cr6;
    cr6 = (unsigned char *) (CR6);
    *cr6 = x; /* use channel */
  }
ffirstm()
  {
    unsigned char *cr9;
    cr9 = (unsigned char *) (CR9);
    *cr9 = 0x00; /* fifo reset */
  }

/*=====*/
/* (msater/slave) Call by inits() */
/*=====*/
time0s()
  {
    unsigned char *cr4s;
    cr4s = (unsigned char *) (CR4S);
    *cr4s = 0x34; /* intervaltmer1 set */
    *cr4s = 0x74; /* intervaltmer2 set */
    *cr4s = 0xba; /* loopcounter set */
  }
usechs(x)
  unsigned char x;
  {
    unsigned char *cr6s;
    cr6s = (unsigned char *) (CR6);
    *cr6s = x; /* use channel number */
  }
ffirsts()
  {
    unsigned char *cr9s;
    cr9s = (unsigned char *) (CR9S);
    *cr9s = 0x00; /* fifo reset */
  }

/*=====*/
/* Call by getad() */
/*=====*/
enable()
  {
    unsigned char *cr7;
    cr7 = (unsigned char *) (CR7);
    *cr7 = 0x01; /* triger enable */
  }
triger()
  {
    unsigned char *cr8;
    cr8 = (unsigned char *) (CR8);
    *cr8 = 0xff; /* (model,2) triger on */
  }

/*=====*/
/* Call by done() */
/*=====*/
status()
  {
    unsigned char *cr10;
    unsigned char x;
    cr10 = (unsigned char *) (CR10);
    x = (*cr10 & 0x0f);
    return(x);
  }
/*=====*/
/* e.o.f */
/*=====*/

```

```

/*****/
/*
/*          PVME-303 SAMPLE PROGRAM (MASTER ADDRESS)          */
/*
/*
/*****/

#define          BASE          0xfc4b0000          /* base adrs */
#define          CR0           BASE+0x01          /* irq ef */
#define          CR1           BASE+0x03          /* irq hf */
#define          CR2           BASE+0x05          /* irq ff */
#define          CR3           BASE+0x07          /* irq done */
#define          VR0           BASE+0x09          /* vct ef */
#define          VR1           BASE+0x0b          /* vct hf */
#define          VR2           BASE+0x0d          /* vct ff */
#define          VR3           BASE+0x0f          /* vct done */
#define          TR0           BASE+0x11          /* interval 1 */
#define          TR1           BASE+0x13          /* interval 2 */
#define          CNR           BASE+0x15          /* convert number */
#define          CR4           BASE+0x17          /* timer/counter set */
#define          CR5           BASE+0x21          /* convert number */
#define          CR6           BASE+0x23          /* ch number */
#define          CR7           BASE+0x25          /* trigger e/d & a/d stop */
#define          CR8           BASE+0x27          /* internal trigger */
#define          CR9           BASE+0x29          /* fifo reset */
#define          CR10          BASE+0x2b          /* poling status */
#define          DR0           BASE+0x30          /* fifo data */

/*****/
/* eof */
/*****/

```

```

/*****/
/*
/*          PVME-303 SAMPLE PROGRAM (SLAVE ADDRESS)          */
/*
/*
/*****/

#define          BASES         0xfc4b0000          /* base adrs */
#define          CR0S          BASES+0x10001       /* irq ef */
#define          CR1S          BASES+0x10003       /* irq hf */
#define          CR2S          BASES+0x10005       /* irq ff */
#define          CR3S          BASES+0x10007       /* irq done */
#define          VR0S          BASES+0x10009       /* vct ef */
#define          VR1S          BASES+0x1000b       /* vct hf */
#define          VR2S          BASES+0x1000d       /* vct ff */
#define          VR3S          BASES+0x1000f       /* vct done */
#define          TR0S          BASES+0x10011       /* interval 1 */
#define          TR1S          BASES+0x10013       /* interval 2 */
#define          CNRS          BASES+0x10015       /* convert number */
#define          CR4S          BASES+0x10017       /* timer/counter set */
#define          CR5S          BASES+0x10021       /* convert number */
#define          CR6S          BASES+0x10023       /* ch number */
#define          CR7S          BASES+0x10025       /* trigger e/d & a/d stop */
#define          CR8S          BASES+0x10027       /* internal trigger */
#define          CR9S          BASES+0x10029       /* fifo reset */
#define          CR10S         BASES+0x1002b       /* poling status */
#define          DROS          BASES+0x10030       /* fifo data */

/*****/
/* eof */
/*****/

```

## 第 1 1 章 そ の 他

#### 10-1 保証について

添付の保証返信カードに御記入の上、御返送下さい。

本ボードの無償保証期間は、お買い求めの日から1年です。ただし、環境条件、操作方法、その他正常な使用方法で使用されるものに限りです。

下記に該当するものについては、無償保証期間であっても無償修理対象とはなりません。

- a. お客様の誤った使用方法、あるいは不注意によって生じた故障、損傷。
- b. 不当な改造、修理に帰すと認められる故障、損傷。
- c. 自然災害によって生じた故障、損傷。
- d. 保証書の御提示がない場合。

#### 10-2 オーダーリング・インフォメーション

PVME-303

1993年 6月 M303-01 (第一版)

- ・本マニュアルの内容については、予告なく変更修正する場合があります。
- ・本マニュアルに記載されている以外のご使用によって損害が発生した場合、当社では責任を負いかねますのでお取扱いには十分ご注意下さい。

インターニックス株式会社  
八王子開発センター

〒192 東京都八王子市片倉町59-10

TEL. 0426 (48) 5200

FAX. 0426 (48) 5201