

## 第4章．汎用Cハンドラ

本機（PMC - 354AT）を簡単に運転・操作することのできる汎用ハンドラ（LIB）です。MS - C、またはTURBO - C、BORLAND - C等でリンクできます。本機の基本的な機能が関数化されており、ユーザは自身の記述するメインルーチンの中から自由に呼び出して使用することができます。【最大32チャンネル＝8ボードの運転も可能】

### 4-1. 適用システム

パソコン本体	：	IBM PC / AT互換機（要ISAバス）
本体メモリ量	：	標準メモリ640KB以上。
OS・言語	：	MS - DOS（3.1以上）、および MS - C（7.0）、Visual - C++、 TURBO - C++（4.0）、または BORLAND - C++（3.1以上）
供給メディア	：	本製品（PMC - 354AT）添付の3.5インチFD
チャンネル数	：	最大32（ボード数＝8枚）

【注】動作の制約等： 複数ボード使用時でも割り込みは1系統のみ。すなわち、割り込み処理ルーチン内で要求元チャンネルを探す方法を採用している。

動作開始、および終了検出は各チャンネルごとに行う。すなわち、チャンネル間の動作に同期性は無い。

指定送り速度は特に無理な条件でない限り、1%以内の誤差で実行される。

図4 - 1 . プログラム構造



## 4-2. 使用方法

**ハードウェアの準備** : 本ハンドラは最大8枚のPMC - 354ATを運転することができます。各ボードの設定、接続は以下のとおりです。

1枚だけ使用するときは、ボード上のスイッチによるI/Oベースアドレス値(1 - 5項)を本ハンドラ初期設定関数に渡す値と一致させます。

複数枚を使用するときは、1 枚目ボード上のスイッチによる I / O ベースアドレス値を本ハンドラ初期設定関数に渡す値と一致させます。割り込みを使用するときは全ボードを同一レベルに設定します。2 枚目以降の各ボードの I / O ベースアドレスは 1 0 0 0 h ずつ加算した値に設定します。ソフト上の制御チャンネル番号は、

1 枚目：0～3チャンネル（X，Y，Z，U）、  
2 枚目：4～7チャンネル（" " " "）、  
8 枚目：28～31チャンネル（" " " "）、  
となります。

表4 - 2 A . 各ボード上のI / Oアドレス・スイッチ設定例

	ディップスイッチDIPSW1						ディップスイッチDIPSW2						I/Oベースアドレス
ボード	1	2	3	4	5	6	1	2	3	4	5	6	(hex)
1 枚目	ON	ON	ON	ON	ON	ON	ON	OFF	ON	ON	OFF	ON	0120 初期設定
2 枚目	ON	ON	ON	OFF	ON	ON	ON	OFF	ON	ON	OFF	ON	1120
3 枚目	ON	ON	OFF	ON	ON	ON	ON	OFF	ON	ON	OFF	ON	2120
4 枚目	ON	ON	OFF	OFF	ON	ON	ON	OFF	ON	ON	OFF	ON	3120
5 枚目	ON	OFF	ON	ON	ON	ON	ON	OFF	ON	ON	OFF	ON	4120
6 枚目	ON	OFF	ON	OFF	ON	ON	ON	OFF	ON	ON	OFF	ON	5120
7 枚目	ON	OFF	OFF	ON	ON	ON	ON	OFF	ON	ON	OFF	ON	6120
8 枚目	ON	OFF	OFF	OFF	ON	ON	ON	OFF	ON	ON	OFF	ON	7120

【注】ディップスイッチDIP SW2の6番を全ボード共、ONに設定すること。（フルデコード・モード）

## プログラムの作成

御自身の作成したメインプログラムをコンパイル、本ハンドラ（LIB）とリンクして使用します。テストには付属のサンプルを利用してください。なお、ライブラリ等は3 - 1項にしたがってインストールしておきます。

図 4 - 2 A . ディレクトリ

```
¥
|
|
MSCIENCE
|
HND354C - - INCLUDE - - H354.H   : ハンドラ共通ヘッダファイル
|
| - - LIB - - - 354TS.LIB   : TURBO - C , BORLAND - C 用スモールモデル
|   | - - 354TL.LIB   : TURBO - C , BORLAND - C 用ラージモデル
|   |
|   | - - 354MS.LIB   : MS - C 用スモールモデル
|   | - - 354ML.LIB   : MS - C 用ラージモデル
|
| - - SMP354CH - - - SAMPLE .C   : サンプル・ソース
```

表 4 - 2 B . 関数一覧

関数名	機能・動作	主なパラメータ等
pmc__open__pmcsys	ボード、本ハンドラの初期化	I / Oアドレス、割り込みレベル
pmc__set__hwmode	ハードウェア動作モードの設定	出力方式、原点方式、アラーム
pmc__pout__const	定速送り動作 / 開始	パルス数、速度、方向
pmc__pout__hspeed	高速送り動作 / 開始	パルス数、速度、方向
pmc__stop__prompt	即停止	
pmc__stop__sdown	減速停止	
pmc__orig__const	定速原点復帰 / 開始	速度、方向
pmc__orig__hspeed	高速原点復帰 / 開始	速度、方向
pmc__set__cpc	現在位置カウンタの設定	カウンタ設定値、計数単位
pmc__status__cur	現在位置・速度ステータスを得る	現在位置、速度
pmc__status__bs	基本ステータスを得る	動作状態、ほか
pmc__status__ex	拡張ステータスを得る	割り込み・停止原因、アラーム
pmc__gps__inp	汎用入力ビットの値を得る	各軸 2 B I T ( 現在値読み込み )
pmc__gps__out	汎用出力ビットの値を更新する	各軸 2 B I T ( ラッチ出力 )
pmc__onintr__func	割り込みで実行するユーザ関数の指定	割り込み要因
pmc__close__pmcsys	本ハンドラの終了	
pmc__get__libver	本ハンドラのバージョン情報取得	

### 応用プログラムの記述

ユーザプログラムは以下の手順で記述します。 実際の作成時はサンプル・ソース ( 図 4 - 2 A 参照 ) に習ってください。

- ( 1 ) 初期化を行う。 【pmc\_\_open\_\_pmcsys ( )】

ここではボード上で設定した I / O ベースアドレスを本ハンドラが認識すると同時に使用する場合の割り込みレベル設定、ボードリセット、その他、ハンドラ内の参照テーブルやデータバッファを初期化する。

- ( 2 ) ハードウェア条件を設定する。 【pmc\_\_set\_\_hwmode ( )】

パルス出力方式 ( パルスモータ・ドライブユニットに合わせる )  
 パルス出力極性 ( " " " " " )  
 原点方式 ( 2 - 7 項 , 参照 )  
 アラーム制御 ( 2 - 6 項 / 2 - 16 項 , 参照 )

- ( 3 ) 動作開始。 ( 定速 / 高速、通常送り / 原点復帰 )

- ( 4 ) 動作状態 ( ステータス ) 取得。 【pmc\_\_status\_\_ ( )】

- ( 5 ) サンプリング停止。 【pmc\_\_stop\_\_ ( )】

- ( 6 ) 本ハンドラを終了する。 【pmc\_\_close\_\_pmcsys ( )】

### 4-3. 関数セット

初期化、動作条件設定、スタート、ストップ、ステータス取得、汎用割り込み処理、等々、各関数はパルスモータの動作を実現する基本機能単位となっています。また各関数は自身の性格から適切な実行手順があります。（前4 - 2項、参照）

#### 【1】PMC - 354ATボード、および本ハンドラの初期化

<pre>int pmc_open_pmcsys(int num_board, WORD address,                     int intr_no)</pre>	
address	PMC - 354ATボード1枚のみ使用時、または複数枚使用時の1枚目ボードに設定されたI/Oベースアドレス（表4 - 2参照）。標準 = 0x100
num_board	PMC - 354ATボードの使用枚数。/ 初期値 = 1
intr_no	使用する割り込みレベル。/ 5, 10, 11, 12, 15 から選択。 範囲外の値は不使用を意味する。（割り込みは全ボードを同一に設定する。）
戻り値	正常終了時： 0 エラー時： エラーコード（負の値 / 表4 - 3参照）
機能・動作	PMC - 354ATボード上で設定したI/Oベースアドレスを本ハンドラが認識すると同時に、使用する場合の割り込みレベル設定、ボードリセット、その他、ハンドラ内の参照テーブルやデータバッファを初期化する。

#### 【2】各軸ハードウェア関連の基本設定

<pre>int pmc_set_hwmode(int axis, int pout_mode, pout_pol,                    int orig_recog, int orig_offset,                    int alm_enb)</pre>	
axis	指定軸の番号。（1枚目ボードのX, Y, Z, Uを各0, 1, 2, 3とし、以下31まで）
pout_mode	パルス出力方式。/ 0：共通、 1：個別 （1 - 2項、1 - 3項、参照）
pout_pol	パルス出力極性。/ 0：負論理、 1：正論理 （2 - 5項、参照）
orig_recog	原点制御方式。 / 0：OLS、 1：Z相DLS、 2：Z相OL（2 - 7項、参照）
orig_offset	Z相方式のとき、指標点（DLS、またはOLS）から原点までのZ相入力数。/ 1 ~ 16
alm_enb	アラーム制御。/ 0：不使用、 1：B接点動作 2：A接点動作 （1 - 3項、参照）
戻り値	正常終了時： 0 エラー時： エラーコード（負の値 / 表4 - 3参照）
機能・動作	ハードウェアの制御方式、接続、信号極性、等のシステム条件を本ハンドラに認識させる。本設定は制御軸ごとに行う。

## 【3】定速送り開始

<pre>int pmc_pout_const(int axis, long *base_speed,                   long pout_num, int pout_dir,                   int pout_while, int intr_onstop)</pre>	
axis	制御軸の番号。 / 0 ~ 31
base_speed	パルス出力速度。(pps)
pout_num	パルス出力数。(指定数)
pout_dir	パルス出力方向。 / 0 : CW(+), 1 : CCW(-)
pout_while	パルス出力制御。 / 0 : 無限連続(停止操作、またはリミットまで)、1 : 指定数だけ
intr_onstop	停止時の割り込み制御。 / 0 : 禁止、 1 : 許可
戻り値	正常終了時 : 0 エラー時 : エラーコード(負の値 / 表4 - 3 参照)
機能・動作	指定制御軸を指定速度(定速)で送り開始する。

## 【4】高速送り開始

<pre>int pmc_pout_hspeed(int axis, long *base_speed,                    long *h_speed, long pout_num,                    int pout_dir, int pout_while,                    long pout_acc, int pout_dls,                    int intr_onstop)</pre>	
axis	制御軸の番号。 / 0 ~ 31
base_speed	パルス出力ベース速度。(pps)
h_speed	パルス出力高速速度。(pps)
pout_num	パルス出力数。(指定数)
pout_dir	パルス出力方向。 / 0 : CW(+), 1 : CCW(-)
pout_while	パルス出力制御。 / 0 : 無限連続(停止操作、またはリミットまで)、1 : 指定数だけ
pout_acc	加速(=減速)点数。 /
pout_dls	DLS入力中の制御。 / 0 : 無視、 1 : 減速
intr_onstop	停止時の割り込み制御。 / 0 : 禁止、 1 : 許可
戻り値	正常終了時 : 0 エラー時 : エラーコード(負の値 / 表4 - 3 参照)
機能・動作	<p>指定制御軸を指定速度(高速=加減速)で送り開始する。 無限連続を指定したときは高速に達した後、連続出力を(停止操作、またはリミットまで)維持する。</p> <p>なお、高速速度はベース速度の8191倍を超えることができない。 (各速度レジスタは13BIT、倍率レジスタは共通のため。)</p>

## 【5】即停止

<code>int pmc_stop_prompt(int axis)</code>	
<code>axis</code>	制御軸の番号。 / 0 ~ 31
戻り値	正常終了時 : 0 エラー時 : エラーコード (負の値 / 表4 - 3 参照)
機能・動作	パルス出力を即停止する。 当軸・直前のパルス出力開始関数で停止時割り込み制御を許可していた場合は割り込み要求が発生する。

## 【6】減速停止

<code>int pmc_stop_sdown(int axis)</code>	
<code>axis</code>	制御軸の番号。 / 0 ~ 31
戻り値	正常終了時 : 0 エラー時 : エラーコード (負の値 / 表4 - 3 参照)
機能・動作	高速動作中の停止操作に使用する。 【4】高速動作開始で指定したベース速度まで減速して停止する。(本関数実行から減速点数だけ追加出力して停止する。) 当軸・直前のパルス出力開始関数で停止時割り込み制御を許可していた場合は、停止時に割り込み要求が発生する。

## 【7】定速原点復帰

<code>int pmc_hp_const(int axis, long *base_speed, int pout_dir, int intr_onstop)</code>	
<code>axis</code>	制御軸の番号。 / 0 ~ 31
<code>base_speed</code>	パルス出力速度。(pps)
<code>pout_dir</code>	パルス出力方向。 / 0 : CW (+)、 1 : CCW (-)
<code>intr_onstop</code>	停止時の割り込み制御。 / 0 : 禁止、 1 : 許可
戻り値	正常終了時 : 0 エラー時 : エラーコード (負の値 / 表4 - 3 参照)
機能・動作	指定制御軸を指定速度 (定速) で原点復帰動作を開始する。