

制 定 日	2006, 9, 8
改 定 日	2011, 5, 16

High-Speed Motor Control Board for PCI

**SMC-4DL-PCI**

**SMC-8DL-PCI**

High-Speed Motor Control Board for PCI Express

**SMC-4DL-PE**

**SMC-8DL-PE**

SMC-4P(PCI)、SMC-2P(PCI)からの移行ガイド

## 目次

1	概要	3
2	入出力部仕様の相違点と結線方法	3
2.1	パルス出力部	3
2.2	エンコーダ入力部	4
2.3	汎用出力部	6
2.4	リミット入力部・汎用入力部	7
3	コネクタピン配置	8
4	CCB-SMC1 と CCB-SMC2 の相違点	9
5	ドライバ概要	10
6	API-SMC(98/PC)と API-SMC(WDM)の相違点	10
6.1	デバイスマネージャでの扱いの違い	10
6.2	デバイス名称の定義について	11
6.3	API 関数について	12
6.4	初期設定関数について	13
7	新旧関数比較	14
7.1	初期化関数	14
7.2	基本動作設定関数	15
7.3	動作ステータス取得関数	16
7.4	モーション動作関数	17
7.5	制御信号設定関数	17
7.6	初期設定関数	18
7.7	拡張動作設定関数	19
7.8	パラメータ演算関数	19
7.9	イベント関数	19

# 1 概要

「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」の入出力部仕様は「SMC-2P(PCI)」「SMC-4P(PCI)」と異なります。既存システムからのスムーズな移行にご活用いただけるよう、以下に入出力仕様の相違点、および、結線に関する注意事項をまとめました。

## 2 入出力部仕様の相違点と結線方法

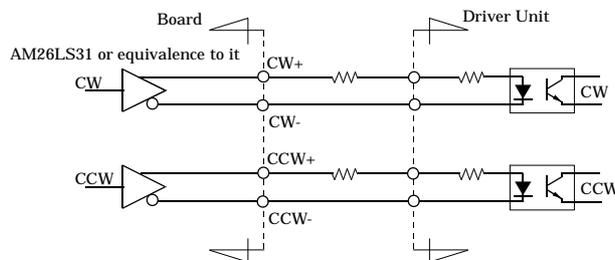
### 2.1 パルス出力部

相違点を以下の表でまとめます。

項目	SMC-2P(PCI), SMC-4P(PCI)仕様	SMC-4DL-PCI, SMC-8DLPCI, SMC-4DL-PE, SMC-8DL-PE 仕様
出力形式	オープンコレクタ出力	差動ラインドライバ出力
速度範囲	0.1 – 1Mpps	0.3 – 9.8Mpps
出力形式	2パルス方式, 共通パルス方式	2パルス方式, 共通パルス方式, 90度位相差パルス方式

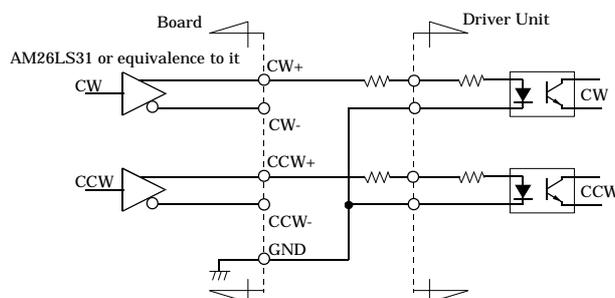
「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」におけるパルス出力部の回路とフォトカプラ入力との接続方法を以下に示します。「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」のパルス出力部は H レベル出力時で 2.5V(min.)、L レベル出力時で 0.5V(max.)の電圧が出力されます。フォトカプラ入力のドライバユニットと接続する際はドライバユニットのパルス入力部が上記電圧で動作するか仕様確認を行った上で接続ください。また、ノイズによる誤動作を防ぐため、他の信号線またはノイズ源から可能な限り離して配線してください。

#### フォトカプラ入力との接続(ドライバユニットが差動出力との接続を保証しているとき)



#### パルス出力回路(フォトカプラ入力との接続)

##### フォトカプラ入力との接続



#### パルス出力回路(フォトカプラ入力との接続)

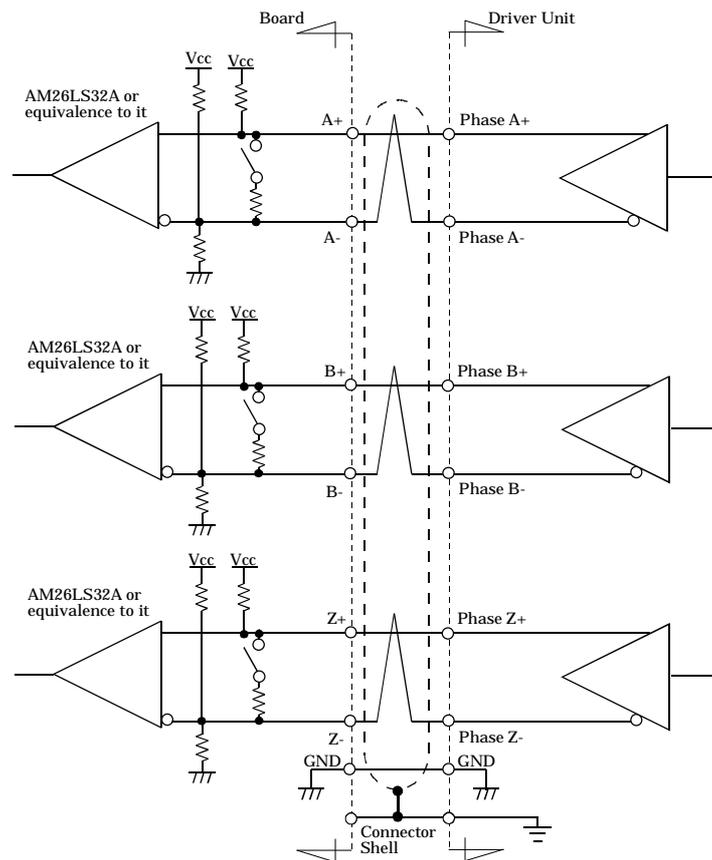
## 2.2 エンコーダ入力部

相違点を以下の表でまとめます。

項目	仕様	
	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DLPCI, SMC-4DL-PE, SMC-8DL-PE
入力形式	高速フォトカプラ絶縁入力	差動入力
応答周波数	1MHz Max.	5MHz Max. (差動出力), 3MHz Max. (TTL レベル出力), 1MHz Max. (オープンコレクタ出力)

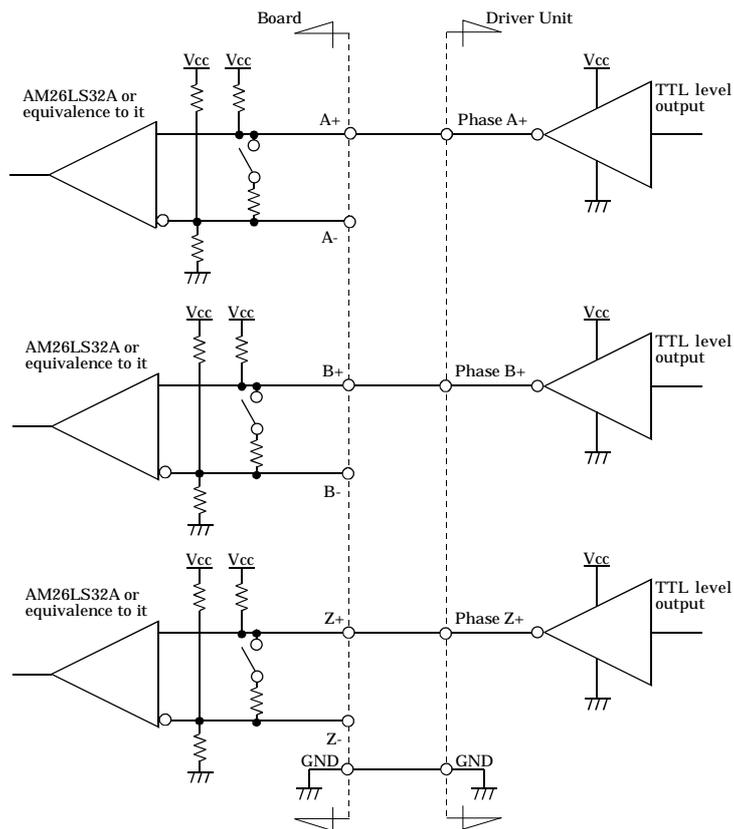
「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」におけるエンコーダ入力部の回路と差動出力、オープンコレクタ出力、TTL レベル出力との接続方法を以下に示します。オープンコレクタ出力、TTL レベル出力接続時は、終端抵抗を挿入しないでください。誤動作、発熱、故障の原因になります。また、使用するケーブルはラインドライバ出力時 10m 以内、オープンコレクタ出力時 3m 以内、TTL レベル出力時 1.5m 以内で使用してください。ノイズによる誤動作を防ぐため、他の信号線またはノイズ源から可能な限り離して配線してください。

### 差動出力との接続



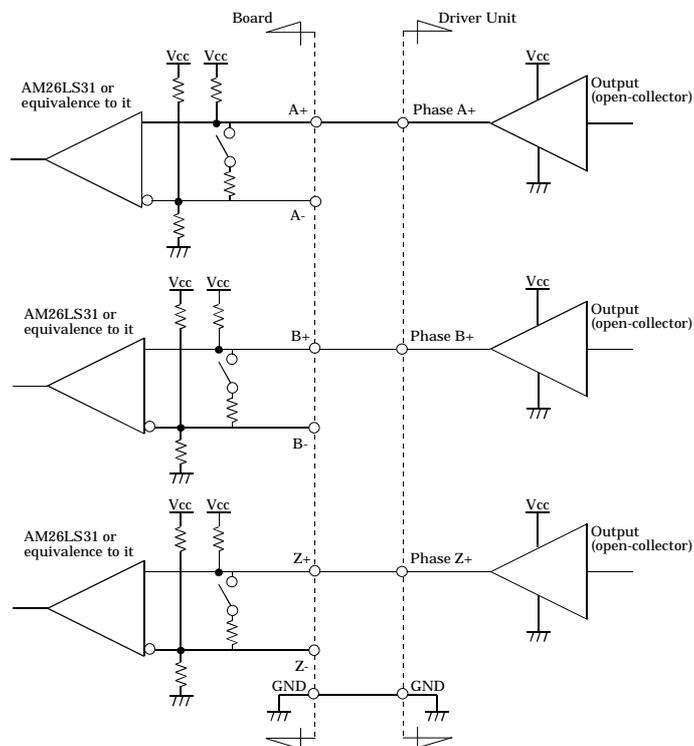
エンコーダ入力回路

## TTL レベル出力との接続



エンコーダ入力回路

## オープンコレクタ出力との接続



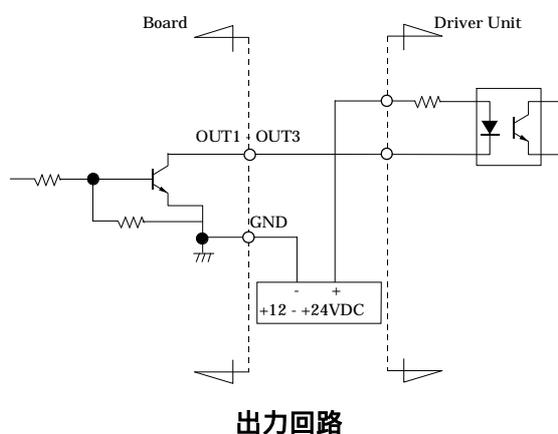
エンコーダ入力回路

## 2.3 汎用出力部

相違点を以下の表でまとめます。

項目	SMC-2P(PCI), SMC-4P(PCI)仕様	SMC-4DL-PCI, SMC-8DLPCI, SMC-4DL-PE, SMC-8DL-PE 仕様
応答時間	1 $\mu$ sec(Max.)	10 $\mu$ sec (入力側負荷 510 $\Omega$ 、+24VDC 使用時)
出力定格電流	1 点あたり最大 100mA	1 点あたり最大 100mA、 1 軸あたり最大 300mA

「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」における汎用出力部の回路とフォトカプラ入力との接続方法を以下に示します。汎用出力部はオープンコレクタ出力になっていますので駆動するためには電源供給源のグランド線の接続が必要です。

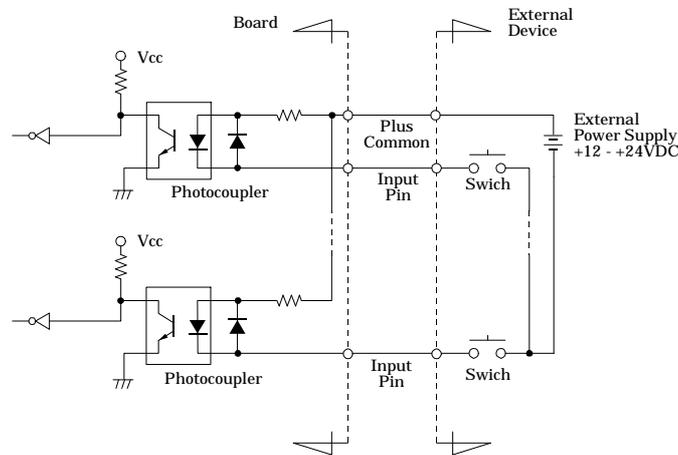


## 2.4 リミット入力部・汎用入力部

相違点を以下の表でまとめます。

項目	SMC-2P(PCI), SMC-4P(PCI)仕様	SMC-4DL-PCI, SMC-8DLPCI, SMC-4DL-PE, SMC-8DL-PE 仕様
入力形式	高速フォトカプラ絶縁入力, フォトカプラ絶縁入力	フォトカプラ入力
応答時間	1 $\mu$ sec Max. , 1msec Max.	200 $\mu$ sec Max.
入力抵抗	1.8k , 3k	4.7k
入力 ON 電流	5.0mA 以上, 3.4mA 以上	2.0mA 以上
入力 OFF 電流	0.5mA 以下, 0.16mA 以下	0.16mA 以下

「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」におけるリミット入力部、汎用入力部の回路とスイッチ回路との接続方法を以下に示します。汎用入力部はフォトカプラによる電流駆動入力になっていますので駆動するためには外部電源(+12V - +24V)が必要です。



リミット入力/汎用入力/制御入力回路

### 3 コネクタピン配置

参考として、「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」の 0 軸目のコネクタピン配置とそれに対応する「SMC-2P(PCI)」「SMC-4P(PCI)」0 軸目のコネクタのピン配置を表でまとめます。

項目	SMC-2P(PCI), SMC-4P(PCI)		SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	
	ピン名	ピン番号	ピン名	ピン番号
パルス出力部			CW+	61
	PCW	B08	CW-	60
			CCW+	63
	DCCW	B09	CCW-	62
エンコーダ入力部	A+	B49	A+	78
			A-	77
	B+	B46	B+	76
			B-	75
	Z+	B44	Z+	74
		Z-	73	
汎用出力部	OUT1	B10	OUT1	64
	OUT2	B11	OUT2	65
	OUT3	B12	OUT3	66
リミット入力部	+LIM	B27	+LIM	91
	-LIM	B26	-LIM	90
	ORG	B28	ORG	92
汎用入力部	IN1	B35	IN1	99
	IN2	B34	IN2	98
	IN3	B33	IN3	97
	IN4	B32	IN4	96
	IN5	B31	IN5	95
	IN6	B30	IN6	94
	IN7	B29	IN7	93
プラスコモン	P.COM	B36	P-COM	100
グランド	GND	B01	GND	59

## 4 CCB-SMC1 と CCB-SMC2 の相違点

「SMC-2P(PCI)」「SMC-4P(PCI)」の専用アクセサリ「CCB-SMC1」と「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」の専用アクセサリ「CCB-SMC2」の相違点を以下の表でまとめます。

項目	CCB-SMC1 仕様	CCB-SMC2 仕様
リミットセンサ用コネクタ	端子台	9 Pin D-Sub コネクタ
外形寸法[mm]	105.0x230.0	105.0x190.0

「CCB-SMC1」と「CCB-SMC2」のコネクタのピン配置を表でまとめます。

項目	CCB-SMC1		CCB-SMC2	
	ピン名	ピン番号	ピン名	ピン番号
パルス出力部			CW+	37
	PCW-	18	CW-	18
			CCW+	36
	DCCW-	17	CCW-	17
エンコーダ入力部	A+	32	A+	32
	A-	13	A-	13
	B+	31	B+	31
	B-	12	B-	12
	Z+	30	A+	30
	Z-	11	Z-	11
汎用出力部	OUT1	16	OUT1	16
	OUT2	15	OUT2	15
	OUT3	14	OUT3	14
リミット入力部	+LIM	2	+LIM	2
	-LIM	1	-LIM	1
	ORG	3	ORG	3
汎用入力部	IN1	29	IN1	29
	IN2	28	IN2	28
	IN3	27	IN3	27
	IN4	26	IN4	26
	IN5	25	IN5	25
	IN6	24	IN6	24
	IN7	23	IN7	23
プラスコモン	P.COM	20 ~ 22, 33 ~ 37	P-COM	20 ~ 22, 33 ~ 35
グラウンド	GND	4 ~ 10	GND	4 ~ 10

コネクタのピン配置については基本的に互換性がありますが、パルス出力形式、エンコーダ入力形式が異なりますので、接続状態の整合性を確認の上で結線してください。

## 5 ドライバ概要

「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」のドライバとして、WDM (Windows Driver Model) ドライバをご提供いたします。

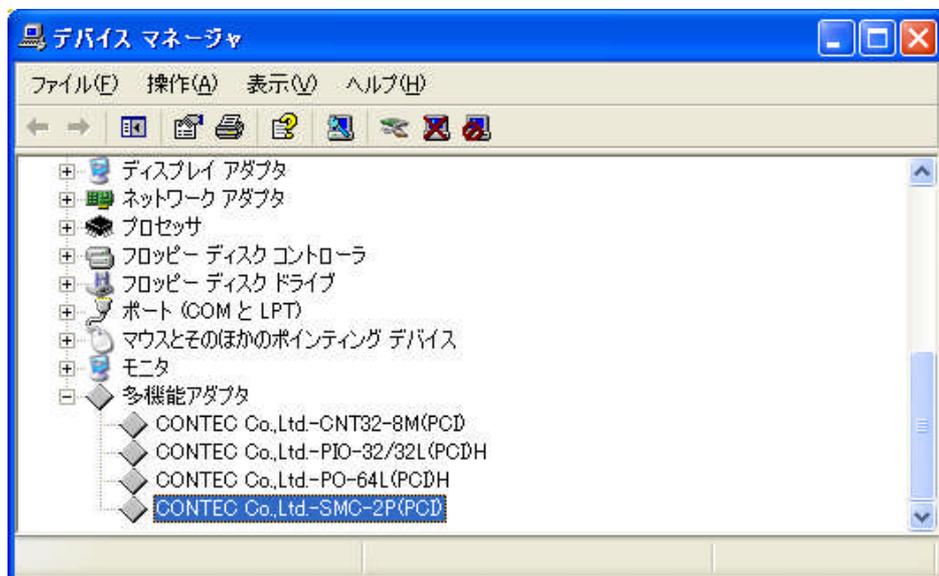
旧API-SMC(98/PC)ドライバでは、Windows 64bit への対応ができないことが判明しているため、WDMドライバでの提供とさせていただきます。

旧 SMC シリーズでのご要望を反映し関数体系の整理を行っております。既存システムからのスムーズな移行にご利用いただけるよう、以下に注意点/相違点をまとめました。

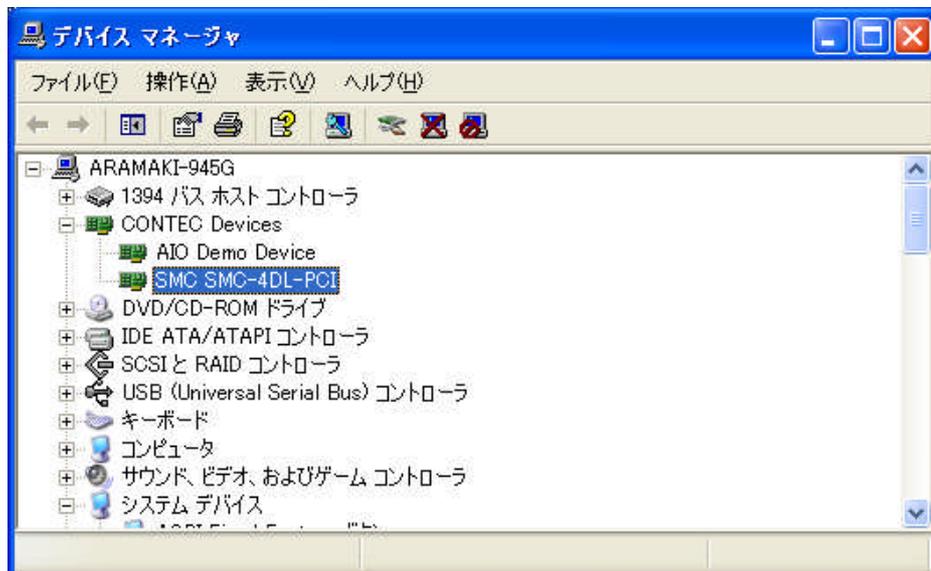
## 6 API-SMC(98/PC)と API-SMC(WDM)の相違点

### 6.1 デバイスマネージャでの扱いの違い

API-SMC(98/PC) では、多機能アダプタに分類されていました。

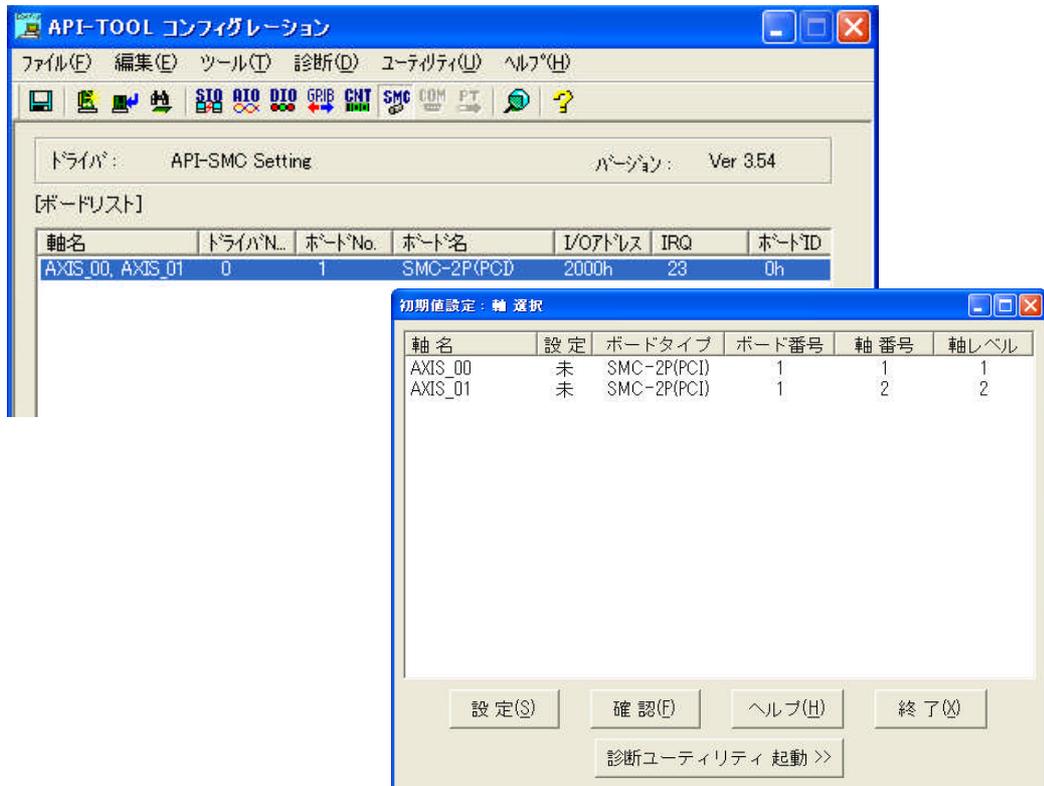


API-SMC(WDM) は、「CONTEC Device」という新規のカテゴリに分類されます。  
(アナログデバイスドライバ API-AIO(WDM) などと同じ分類になります)

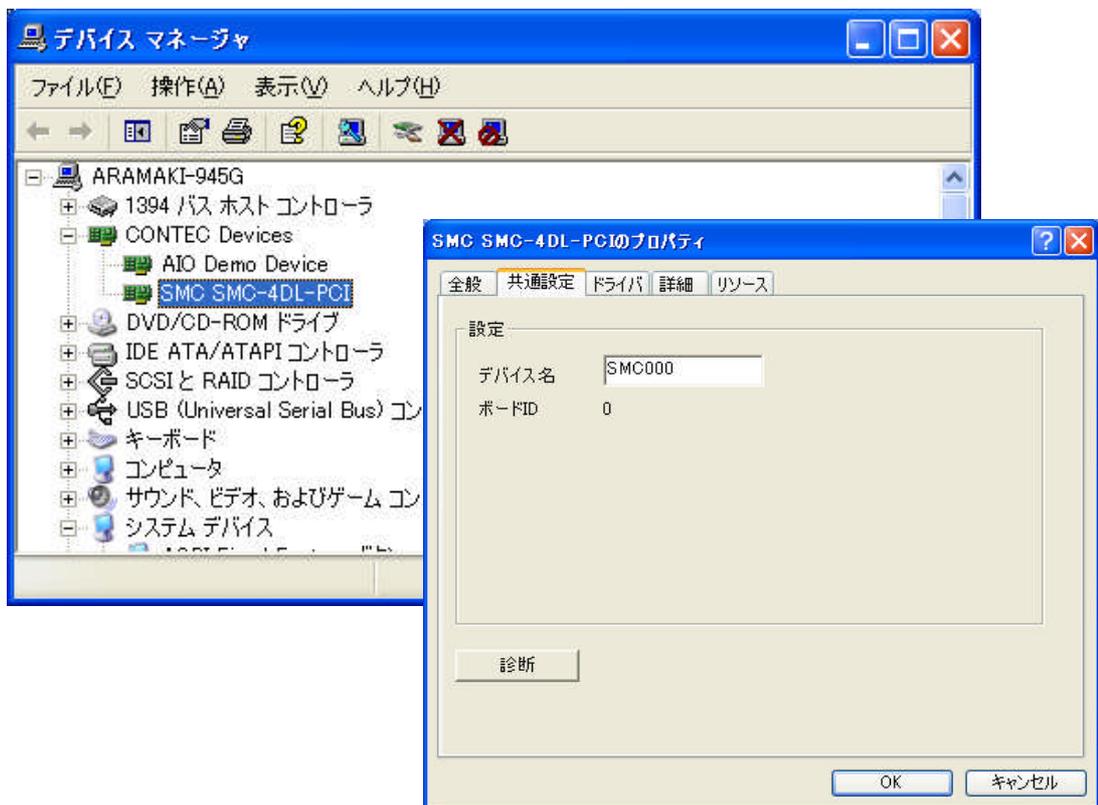


## 6.2 デバイス名称の定義について

API-SMC(98/PC) では、「API-TOOL コンフィグレーション」にて使用するデバイスを登録し、初期設定ユーティリティで軸ごとにデバイス名称(各軸の名前)を設定していました。



API-SMC(WDM) は、Windows 標準のデバイスマネージャからボードを選択し、「プロパティ」の「共通設定タブ」でボードごとにデバイス名を設定します。



### 6.3 API 関数について

API-SMC(98/PC) は、デバイスネーム(各軸の名前)をパラメータとして、各関数を呼び出しますが、API-SMC(WDM) では、デバイス名(文字列)で、デバイスの初期化(SmcWInit)を行い復帰値のデバイスハンドルと軸番号(数値)で関数を呼び出します。

例 >

API-SMC(98/PC)の場合

```
Ret = SmcInit("AXIS_00"); // 軸の初期化
:
Ret = SmcSetAccelRate("AXIS_00",50); // 加速レートの設定
Ret = SmcSetReady("AXIS_00",SMC_JOG,SMC_CW); // 開始準備パラメータの設定
Ret = SmcMotionStart("AXIS_00"); // 軸のモータ動作を開始
:
Ret = SmcExit("AXIS_00"); // 軸の終了処理
```

API-SMC(WDM)の場合

```
short Id;
Ret = SmcWInit("SMC00", &Id); // デバイスの初期化とデバイスハンドルの取得
Ret = SmcWSetPulseType( Id, AxisNo, PulseType, DirTimer); // パルスの出力モードを指定
:
Ret = SmcWSetInFilterType( Id, AxisNo, FilterType); // ±EL、SD、ORG、ALM、INP の入力フィルタ設定
Ret = SmcWSetInitParam( Id, AxisNo); // 初期設定内容をデバイスにセット
:
Ret = SmcWSetAccelRate(Id, AxisNo,50); // 加速時間の設定
Ret = SmcWSetReady( Id, AxisNo, CSMC_JOG, CSMC_CW); // 開始準備パラメータの設定
Ret = SmcWMotionStart( Id, AxisNo ); // 軸のモータ動作を開始
:
Ret = SmcWExit( Id ); // デバイスアクセスの終了処理
```

#### 6.4 初期設定関数について

API-SMC(98/PC) は、初期設定ユーティリティでのみ軸の初期設定可能でしたが、

API-SMC(WDM) では、API 関数でのご提供となります。

各種初期設定関数 (SmcWSetPulseType 等) を実行後、SmcWSetInitParam 関数呼び出しにて設定内容の反映を行ってください。本関数を実行するまで、ハードウェアに対し設定内容は反映されません。

PC 起動後 (ドライバ起動後)、一度実行すれば、設定内容は保持されます。

設定内容については SmcWGet ~ 関数にて取得可能です。

本ドライバに添付されているサンプル "Setting" を実行することで、初期設定を実行することも可能です。

例 >

API-SMC (WDM) の場合

```
Ret = SmcWInit("SMC00", Id);
```

```
Ret = SmcWSetPulseType( Id, AxisNo, PulseType, DirTimer); // パルスの出力モードを指定
```

```
    : <その他、デフォルト以外で使用する場合は設定します。>
```

```
    :
```

```
Ret = SmcWSetInFilterType( Id, AxisNo, FilterType); // ±EL、SD、ORG、ALM、INP の入力フィルタ設定
```

```
Ret = SmcWSetInitParam( Id, AxisNo);
```

## 7 新旧関数比較

新関数名は、旧関数よりの移行の際イメージしやすいよう、基本的には ” Smc ~ ” から “SmcW ~ ” の変更となっていますが、一部関数では同じ機能であっても関数名が変更されていますのでご注意ください。  
 なお、SmcWXxxReady 関数等の ” Xxx ” は Get/Set の両方が存在していることを意味します。

### 7.1 初期化関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、 ×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>初期化関数</b>			
SmcInit			SmcWInit
SmcInitAll		×	
SmcExit			SmcWExit
SmcExitAll		×	
	×		SmcWGetErrorString

 :追加関数

**SmcWInit** : 軸単位での初期化関数からボード単位での初期化関数に変更となります。

<API-SMC(98/PC)>

```
Ret =SmcInit("AXIS_00");
```

<API-SMC(WDM)>

```
Ret =SmcWInit( "SMC00", Id )
```

**SmcWExit** : 軸単位での終了処理関数からボード単位での終了処理関数に変更となります。

<API-SMC(98/PC)>

```
Ret =SmcExit("AXIS_00");
```

<API-SMC(WDM)>

```
Ret = SmcWExit( Id );
```

**SmcWGetErrorString** : エラーコードの内容を文字列に変換します。

## 7.2 基本動作設定関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>基本動作設定関数</b>			
SmcXxxMasterSlave		×	
SmcXxxReady			SmcWXxxReady
SmcXxxStartSpeed			SmcWXxxStartSpeed
SmcXxxTargetSpeed			SmcWXxxTargetSpeed
SmcXxxAccelRate			SmcWXxxAccelTime
SmcXxxDecelRate			SmcWXxxDecelTime
SmcXxxSSpeed			SmcWXxxSSpeed
SmcXxxStopPosition			SmcWXxxStopPosition
SmcXxxPreStopPosition		×	
SmcSetRevisePosition		×	
	×		SmcWSetResolveSpeed
SmcGetResolveSpeed			SmcWGetResolveSpeed
SmcGetMinSpeed		×	
SmcGetMaxSpeed		×	
SmcGetResolveRate		×	
SmcGetMinRate		×	
SmcGetMaxRate		×	

**追加関数**

**SmcXxxMasterSlave** : 同期スタート/ストップ時、および補間動作時にマスター/スレーブの概念が不要となったため、本関数は不要となります。

また、同期スタート条件として、下記の条件が存在します。

・1ボードにつき、1組の同期動作の組み合わせのみ使用可能です。

**SmcXxxAccelTime** : 設定内容は加速度から加速時間設定に変更となります。

<API-SMC(98/PC)>

速度変更時の加速レートの設定を行います。

Ret = SmcSetAccelRate("AXIS\_00", dbIAccelRate); // 単位は[ms/KPPS]です

<API-SMC(WDM)>

加速時間を設定します。

Ret = SmcWSetAccelTime( Id , AxisNo , AccelTime ); //単位は[ms]です。

**SmcXxxDecelTime** : 設定内容は加速度から減速時間設定に変更となります。

<API-SMC(98/PC)>

減速時の減速レートの設定を行います。

Ret = SmcSetDecelRate ("AXIS\_00", dbIDecelRate); // 単位は[ms/KPPS]です

<API-SMC(WDM)>

減速時間を設定します。

Ret = SmcWSetDecelTime ( Id , AxisNo , DecelTime); //単位は[ms]です。

**SmcXxxPreStopPosition** : サポートしません。

- SmcSetRevisePosition** : 相当機能は別途初期設定関数にて設定可能となります。
- SmcWSetResolveSpeed** : 初期設定ユーティリティでの設定のみでしたが、関数として追加しました。
- SmcGetMinSpeed** : サポートしません。
- SmcGetMaxSpeed** : サポートしません。
- SmcGetResolveRate** : 加速度分解能の概念がなくなる為、不要となります。
- SmcGetMinRate** : 加速度分解能の概念がなくなる為、不要となります。
- SmcGetMaxRate** : 加速度分解能の概念がなくなる為、不要となります。

### 7.3 動作ステータス取得関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 関数名変更、 x :使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>動作ステータス取得関数</b>			
SmcXxxOutPulse			SmcWXxxOutPulse
SmcXxxCountPulse			SmcWXxxCountPulse
SmcGetBankNo		x	SmcWGetBankNo
SmcGetPulseSts			SmcWGetPulseStatus
SmcGetMoveSts			SmcWGetMoveStatus
SmcGetStopSts			SmcWGetStopStatus
SmcGetLimitSts			SmcWGetLimitStatus
	x		SmcWGetLatchOutPulse
	x		SmcWGetLatchCountPulse
	x		SmcWGetMoveSpeed
	x		SmcWGetZCount

 : 追加関数

**SmcGetBankNo** : バンク動作関連の関数群です。本ボードでは、サポートされません。

**SmcGetStopSts** : 減速停止信号の方向が取得不可となります。

<API-SMC(98/PC)>

```
Ret = SmcGetStopSts ("AXIS_00", &bStopSts); // 軸名称で取得します。
```

<API-SMC(WDM)>

```
Ret = SmcWGetStopStatus ( Id , AxisNo , &StopSts); // デバイス名、軸番号で取得します。
```

**SmcGetLimitSts** : 取得できるリミット状態の SD 信号は ± 共通となります

**SmcWGetLatchOutPulse** : LTC 信号入力によりラッチされたフィードバック出力パルス数を取得します。

**SmcWGetLatchCountPulse** : LTC 信号入力によりラッチされたエンコーダカウンタ値を取得します。

**SmcWGetMoveSpeed** : 動作中のパルス出力速度を取得します。

**SmcWGetZCount** : 原点復帰動作に伴う Z 相カウント動作時の Z 相カウント数、もしくは Z 相カウント動作時の Z 相カウント数を取得します。

#### 7.4 モーション動作関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>モーション動作関数</b>			
SmcMotionStart			SmcWMotionStart
SmcMotionStartAll		×	
SmcMotionChange			SmcWMotionChange
SmcMotionStop			SmcWMotionStop
SmcMotionDecStop			SmcWMotionDecStop
SmcMotionBankStop		×	SmcWMotionBankStop
	×		<b>SmcWSyncMotionStart</b>

:追加関数

- SmcWMotionStart** : SmcWSetInitParam を実行していないと動作しません  
(実行した軸のみ動作可能)。
- SmcMotionStartAll** : 同期設定関数 SmcWSetSync で軸同期を設定し、  
SmcWSyncMotionStart で同期スタートします。  
サンプル " SyncMove"を参照してください。
- SmcMotionChange** : 目標速度のみ変更可能であったのが、目標位置の変更の変更も  
追加するため、選択の為の引数追加となりました。
- SmcMotionBankStop** : バンク動作関連の関数群です。本ボードでは、サポートされません。
- SmcWSyncMotionStart** : 同期スタート待ち状態の同期制御対象軸の動作を開始します。

#### 7.5 制御信号設定関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>制御信号設定関数</b>			
SmcSetAlarmCLR			SmcWSetAlarmClear
SmcXxxLimitMask			SmcWXxxLimitMask
SmcXxxDigitalOut			SmcWXxxDigitalOut
SmcGetDigitalIn			SmcWGetDigitalIn
SmcXxxStepMode		×	
SmcXxxServoPower		×	
SmcXxxHoldOFF			SmcWXxxHoldOFF
SmcGetAlarmCode			SmcWGetAlarmCode
SmcSetCountCLR			SmcWSetErcOut

- SmcWSetAlarmClear** : ワンショットパルス出力としてもご使用可能です。
- SmcXxxLimitMask** : マスク可能なリミットは、減速停止(SD)、アラーム(ALM)のみです。  
±LIM をマスクする場合は、±LIM リミットの論理を変更してください。
- SmcXxxStepMode** : 未サポートです。汎用出力を使用してください。

- SmcXxxServoPower** : 未サポートです。汎用出力を使用してください。
- SmcWSetErcOut** : 偏差カウンタクリア信号をレベル出力と設定した場合、本関数で ON/OFF を行います。

## 7.6 初期設定関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

\* これらの新関数で設定する項目は、旧ドライバの初期設定ユーティリティで設定したものです。

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
初期設定関数			
SmcXxxInitialParam		×	
	×		SmcWXxxPulseType
	×		SmcWXxxEncType
	×		SmcWXxxCtrlTypeOut
	×		SmcWXxxCtrlTypeIn
	×		SmcWXxxOrgLog
	×		SmcWXxxCtrlInOutLog
	×		SmcWXxxErcMode
	×		SmcWXxxErcAlmClearTime
	×		SmcWXxxOrgMode
	×		SmcWXxxSAccelType
	×		SmcWXxxInFilterType
	×		SmcWXxxSDMode
	×		SmcWXxxCounterMode
	×		SmcWXxxInitParam

**追加関数**

- SmcWXxxPulseType** : パルス出力モードの設定を行います。
- SmcWXxxEncType** : エンコーダ入力形式の設定を行います。
- SmcWXxxCtrlTypeOut** : 制御出力信号形式の設定を行います。
- SmcWXxxCtrlTypeIn** : 制御入力信号形式の設定を行います。
- SmcWXxxOrgLog** : 原点入力論理の設定を行います。
- SmcWXxxCtrlInOutLog** : 制御入出力信号論理の設定を行います。
- SmcWXxxErcMode** : 偏差カウンタクリア信号の設定を行います。
- SmcWXxxErcAlmClearTime** : 偏差カウンタクリア/アラームクリア信号幅、および偏差カウンタクリア信号 OFF タイマ時間の設定を行います。
- SmcWXxxOrgMode** : 原点復帰形式の設定を行います。
- SmcWXxxSAccelType** : S 字加減速の使用有無の設定を行います。
- SmcWXxxInFilterType** : ±EL、SD、ORG、ALM、INP の入力フィルタ設定を行います。
- SmcWXxxSDMode** : SD 信号入力時の動作設定を行います。
- SmcWXxxCounterMode** : カウンタの動作設定を行います。
- SmcWXxxInitParam** : 初期設定の内容をボードに反映します。

### 7.7 拡張動作設定関数

バンク動作関連の関数群です。「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」では、バンク機能を持たないため下記の関数はサポートされません。

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>拡張動作設定関数</b>			
SmcXxxBankReady		×	SmcWXxxBankReady
SmcXxxBankNumber		×	SmcWXxxBankNumber
SmcXxxBankDistance		×	SmcWXxxBankDistance
SmcXxxBankStartSpeed		×	SmcWXxxBankStartSpeed
SmcXxxBankTargetSpeed		×	SmcWXxxBankTargetSpeed
SmcXxxBankBaseSpeed		×	SmcWXxxBankBaseSpeed
SmcXxxBankTime		×	SmcWXxxBankTime
SmcXxxBankAccelRate		×	SmcWXxxBankAccelTime
SmcXxxBankDecelRate		×	SmcWXxxBankDecelTime
SmcXxxBankSSpeed		×	SmcWXxxBankSSpeed

### 7.8 パラメータ演算関数

バンク動作関連の関数群です。「SMC-4DL-PCI」「SMC-8DL-PCI」および「SMC-4DL-PE」「SMC-8DL-PE」では、バンク機能を持たないため下記の関数はサポートされません。

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>パラメータ演算関数</b>			
SmcCalcBankParam		×	
SmcCalcTimeRequired		×	

### 7.9 イベント関数

ご使用のボードで使用できる関数を下記に示します。

( :使用可、 :条件付で使用可 or 動作、引数変更、×:使用不可)

旧関数名	SMC-2P(PCI), SMC-4P(PCI)	SMC-4DL-PCI, SMC-8DL-PCI, SMC-4DL-PE, SMC-8DL-PE	新関数名
<b>イベント関数</b>			
SmcStopEvent			SmcWStopEvent
SmcBankEvent		×	SmcWBankEvent
SmcCountEvent			SmcWCountEvent

**SmcBankEvent** : バンク動作関連の関数群です。本ボードでは、サポートされません。

以上