

DIO98 プロバイダ CONTEC DIO ボード

Version 1.0.0

ユーザーズ ガイド

July 17, 2012

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0.0	2006-02-23	初版
1.0.0.1	2010-02-10	エラーコード追加
1.0.0.2	2011-03-11	プロバイダ登録ツールに関する追記
1.0.0	2012-07-17	ドキュメントのバージョンルールを変更

【対応機器】

機種	バージョン	注意事項

目次

1. はじめに	4
2. プロバイダの概要	5
2.1. 概要	5
2.2. メソッド・プロパティ	6
2.2.1. CaoWorkspace::AddController メソッド	6
2.2.2. CaoController::AddVariable メソッド	6
2.2.3. CaoVariable::get_Attribute プロパティ	7
2.2.4. CaoVariable::get_Value プロパティ	7
2.2.5. CaoVariable::put_Value プロパティ	7
2.3. 変数一覧	9
2.3.1. コントローラクラス	9
2.4. エラーコード	9
2.5. CAO-DIO API 対応表	10
3. サンプルプログラム	11

1. はじめに

本書は、CONTEC 製 DIO ボードにアクセスするためのプロバイダである、DIO98 プロバイダのユーザーズガイドです。

詳細については、CONTEC 社 API-DIO Help を参照して下さい。

注意: DIO プロバイダを使用するには、DIO ボードのドライバをインストールしなければなりません。
ドライバインストール後にプロバイダをレジストリ登録する必要があります。レジストリ登録の方法は表 2-1 を参照してください。

2. プロバイダの概要

2.1. 概要

DIO98 プロバイダは, CAO API を実行するときに対応する DIO (98PC) の API を実行します. CAO API と DIO(98PC) API の対応については表 2-7 を参照してください.

プロバイダの概要について記述します.

表 2-1 DIO プロバイダ

ファイル名	CaoProvDIO98.dll
ProgID	CaoProv.CONTEC.DIO98
レジストリ登録 ¹	regsvr32 CaoProvDIO98.dll
レジストリ登録の抹消	regsvr32 /u CaoProvDIO98.dll

¹ プロバイダの登録は regsvr32.exe または RegCOM.exe ([スタート]→[ORiN2]→[Tools])で実行できます. DIO ボードのドライバをインストールしていないと, DIO プロバイダの登録はできません.

2.2. メソッド・プロパティ

2.2.1. CaoWorkspace::AddController メソッド

DIO プロバイダでは Controller オブジェクトの生成時に DIO ボードとの接続処理を行います。接続時にオプション文字列でデバイスを指定します。

```

AddController
(
    "<コントローラ名>",           // コントローラ名.
    "CaoProv. CONTEC. DIO98",    // プロバイダ名. 固定.
    "<マシン名>",                // プロバイダの実行マシン名.
    "<オプション>"               // オプション文字列.
)

```

以下にオプション文字列に指定するリストを示します。

表 2-2 CaoWorkspace::AddController のオプション文字列

オプション	意味
Board=<Driver No.><Group No.>	接続先のデバイス番号, グループ番号の指定 省略時は, コントローラ名をデバイス名として接続
Mutex[=<0 1>]	マルチプロセス時の排他処理 0:OFF(デフォルト) 1:ON

2.2.2. CaoController::AddVariable メソッド

このメソッドでは, DIO ボードにアクセスする変数オブジェクトを生成します。

```

AddVariable
(
    "<変数名><論理番号>",       // 変数名.
    "<オプション>"               // オプション文字列.
)

```

変数名には, 2.3.1 の変数に 10 進数の論理番号を付加した名前を使用します。

2.3.1 以外の変数名を指定したとき, または論理番号を 10 進数以外で指定したときは, このメソッドはエラーを返します。

論理番号の意味は, 表 2-3 の Len オプションの値によって意味が変わります。

Len=1 のとき: 論理ビット

Len=8 のとき: 論理ポート

入力できる値の範囲については, CONTEC 社 API-DIO Help を参照して下さい。このメソッドでは, 入力の範囲外を指定してもエラーは発生しません。get_Value, put_Value を実行した時にエラーが発生します。

以下にオプション文字列に指定するリストを示します。

表 2-3 CaoController::AddVariable のオプション文字列

オプション	意味
Len [=<1 8>]	データ長の指定 (bit) (デフォルト:1)
BCD [= [+ -]<Figure>]	BCD 変換の桁数指定 (デフォルト:0) 有効範囲については, CONTEC 社 API-DIO Help を参照して下さい.
Filter=<時定数>	デジタルフィルタの時定数の指定 (デフォルト:デジタルフィルタなし) 有効範囲については, CONTEC 社 API-DIO Help を参照して下さい.

また BCD オプションは, 値の取得, 設定時に使用されるかどうかは変数によって異なる. 以下に変数ごとに値の取得, 設定時に BCD オプションが有効かどうかの一覧を示す.

表 2-4 変数毎の BCD オプションの使用状況一覧

	Get	Put
IN	○	-
OUT	-	○
IO	○	○

上記の表で“-”となっている場合は, BCD オプションは使用されません.

2.2.3. CaoVariable::get_Attribute プロパティ

変数の Read/Write の属性情報を取得します.

表 2-5 属性とその値の対応表

属性	値
Read	1
Write	2

2.2.4. CaoVariable::get_Value プロパティ

変数に対応する情報を取得します. 各変数の実装状況および取得データについては, 2.3.1 を参照して下さい.

2.2.5. CaoVariable::put_Value プロパティ

変数に対応する情報を設定します. 各変数の実装状況および設定データについては, 2.3.1 を参照して下さい.

さい。

2.3. 変数一覧

2.3.1. コントローラクラス

表 2-6 コントローラクラス ユーザ変数一覧

変数名	データ型	説明	属性	
			get	put
IN?	VT_I4, VT_UI1	データの取得を行います。 変数名の後ろに論理ポート番号をします。 例) “IN20” BCD オプションが ON の時は VT_I4, OFF の時は VT_UI1 でデータを取得します。	○	-
OUT?	VT_I4, VT_UI1	データの設定を行います。 変数名の後ろに論理ポート番号をします。 例) “OUT20” CaoVariable::put_Value 時に指定するデータ型は, BCD オ プションが ON の時は VT_I4, OFF の時は VT_UI1 でデー タを設定します。 CaoVariable::get_Value 時にはリードバックを行います。	○	○
IO?	VT_I4, VT_UI1	データの取得, 設定を行います。 変数名の後ろに論理ポート番号をします。 例) “IO20” CaoVariable::get_Value プロパティ時には変数名 “IN”, CaoVariable::put_Value プロパティ時には変数名 “OUT” と 同様の動作をします。	○	○

2.4. エラーコード

DIO98 プロバイダでは, 固有のエラーコードとして DIO(98PC) API で規定されたエラー番号を “0x8010000” でマスクした値を返します。

例) DIO(98PC) API のエラー: 0x0003 → CAO API のエラー: 0x80100003

DIO API の詳細については, CONTEC 社 API-DIO Help を参照してください。

ORiN2 共通エラーについては, 「[ORiN2 プログラミングガイド](#)」のエラーコードの章を参照してください。

2.5. CAO-DIO API 対応表

表 2-7 CAO-DIO API 対応表

CAO API	DIO API	備考
CaoWorkspace:: AddController()	DioOpen()	Board オプションがある場合
	DioOpenEx()	Board オプションがない場合
	DioUseMutex	Mutex オプションがある場合のみ
CaoWorkspaces:: Remove()	DioClose()	
CaoController:: AddVariable()	DioSFilter()	Filter オプションがある場合のみ
CaoVariable:: get_Value()	DioNInpBCD()	変数名“IN”又は“IO”の場合 BCD オプションで負の値を指定した場合
	DioInpBCD()	変数名“IN”又は“IO”の場合 BCD オプションで正の値を指定した場合
	DioInpBit()	変数名“IN”又は“IO”の場合 BCD オプションを指定しない場合 Len オプションで 1 を指定した場合
	DioInpByte()	変数名“IN”又は“IO”の場合 BCD オプションを指定しない場合 Len オプションで 8 を指定した場合
	DioEchoBackBit()	変数名“OUT”の場合 Len オプションで 1 を指定した場合
	DioEchoBackByte()	変数名“OUT”の場合 Len オプションで 8 を指定した場合
CaoVariable:: put_Value()	DioNOutBCD()	変数名“OUT”又は“IO”の場合 BCD オプションで負の値を指定した場合
	DioOutBCD()	変数名“OUT”又は“IO”の場合 BCD オプションで正の値を指定した場合
	DioOutBit()	変数名 OUT”又は“IO”の場合 BCD オプションを指定しない場合 Len オプションで 1 を指定した場合
	DioOutByte()	変数名 OUT”又は“IO”の場合 BCD オプションを指定しない場合 Len オプションで 8 を指定した場合

DIO API の詳細については、CONTEC 社 API-DIO Help を参照して下さい。

3. サンプルプログラム

以下に変数“DPM”でデュアルポートメモリにアクセスするサンプルを示します。

List 3-1**Sample.frm**

```
Private caoEng As CaoEngine
Private caoCtrl As CaoController
Private caoVar As CaoVariable

Private Sub Form_Load()

    Set caoEng = New CaoEngine
    Set caoCtrl = caoEng.Workspaces(0).AddController("DIOTest", " CaoProv. CONTEC.DIO ",
"", "Board=0:1")
    Set caoVar = caoCtrl.AddVariable("I020", "")
End Sub

Private Sub CmdPut_Click()

    Dim data As Byte
    data = 1

    caoVar.Value = data

End Sub

Private Sub cmdGet_Click()

    Dim Ret As Variant

    Text1.Text = caoVar.Value

End Sub
```