

1.1 DPCI-PLC 制御関数ライブラリ

WDPLCPCI . DLL (Windows95 /98 用)

NTDPLC . DLL (Windows NT 4.0 が提供しているコマンド(制御関数)は次の通りです。用)

(以下の説明では WDPLCPCI.DLL と記述しています。NT をご使用の方は適宜読み替えてください。)

例題は Visual Basic 5.0 の書式で記述しています。

関数名	機能
StartDplcDevice()	デバイスの開始
EndDplcDevice()	デバイスの終了
board_init()	インタフェースボードの初期化
unitrst()	各ユニットの初期化
Xin_byt()	Xデバイスの読み出し (1バイト単位)
Xin_bit()	Xデバイスの読み出し (1ビット単位)
Yout_byt()	Yデバイスの書き込み (1バイト単位)
Yout_bitrst()	Yデバイスのリセット
Yout_bitset()	Yデバイスのセット
From()	バッファメモリ読み出し
To_cmd()	バッファメモリ書き込み

1.1.1 デバイスの開始

StartDplcDevice

機能 デバイスの開始

形式 StartDplcDevice Lib "WDPLCPCI.DLL" () As Long

戻り値 : 0:異常 0以外:デバイスハンドル

解説 デバイスハンドルを取得します。

```
例
hDevice = StartDplcDevice()
If hDevice = 0 Then
    MsgBox "NG"
Else
    MsgBox "OK"
End If
```

1.1.2 デバイスの終了

EndDplcDevice

機能 デバイスの終了

形式 EndDplcDevice Lib "WDPLCPCI.DLL" () As Long

 戻り値 : 0:異常 0以外:正常

解説 デバイス ハンドルを解放します。

例 ret = EndDplcDevice(hDevice)
 If ret = 0 Then
 MsgBox "NG"
 Else
 MsgBox "OK"
 End If

1.1.3 インタフェースポートの初期化

bord_init

機能 インタフェースポートの初期化

形式 bord_init Lib "WDPLCPCI.DLL" (ByVal hDevice As Long) As Long

hDevice	:	デバイスハンドル
戻り値	:	0:正常 0以外:異常

解説 インタフェースポートの初期化をします。この関数を実行することでインタフェースポートのアクセスが可能となります。

例 ret = bord_init(hDevice)
 If ret = 0 Then
 MsgBox "OK"
 Else
 MsgBox "NG"
 End If

1.1.4各ユニットの初期化

unitrst

機能 各ユニットの初期化

形式 unitrst Lib "WDPLCPCI.DLL" (ByVal hDevice As Long) As Long

hDevice : デバイスハンドル
戻り値 : 0:正常 0以外:異常

解説 AnN, AnSなどの各ユニットの初期化をします。電源投入時と同等になります。

例
ret = unitrst(hDevice)
If ret = 0 Then
 MsgBox "OK"
Else
 MsgBox "NG"
End If

1.1.5 Xデバイスの読み出し (1バイト単位)

Xin_byt

機能 Xデバイスの読み出し (1バイト単位)

形式 Xin_bit Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal base As Byte, _
ByVal port As Byte, _
ByVal xдно As Byte) As Byte

hDevice	:	デバイスハンドル	
base	:	ベース番号	0~7
port	:	I/Oポート番号	0~7
xдно	:	XデバイスのNo.	0~63
戻り値	:	Xデバイスの読み出したデータ	

base,port,xдноで指定する値の範囲は使用するベース、ユニットにより異なります。

解説 base,port,xдноで指定されたXデバイスの入力番号を含む1バイト(8点)分のデータを読み出します。

例

システム構成



ユニット名 = A1SX41

ベース番号 = 0

I/Oポート番号 = 2

入力データ (X) = 1FH 18H 17H 10H 0FH 08H 07H 00H

00000000 00001000 0000

0000 00001001

00H

08H

00H

09H

data = Xin_byt(hDevice, 0, 2, 3)

(data = 9 X00~X07の1バイト分のデータ)

data = Xin_byt(hDevice, 0, 2, 0)

(data = 9 X00~X07の1バイト分のデータ)

xдноに0または3を指定しても、同じ内容が得られる。

```
data = Xin_byt(hDevice, 0, 2, 19)
      (data = 8 X10~ X170 1バイト分のデータ )
```

1.1.6 Xデバイスの読み出し (1ビット単位)

Xin_bit

機能 Xデバイスの読み出し (1ビット単位)

形式 Xin_bit Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal base As Byte, _
ByVal port As Byte, _
ByVal xдно As Byte) As Byte

hDevice	:	デバイスハンドル	
base	:	ベース番号	0~7
port	:	I/Oポート番号	0~7
xдно	:	Xデバイスの No.	0~63
戻り値	:	Xデバイスの読み出したデータ	

base, port, xдноで指定する値の範囲 使用するベース, ユニットにより異なります。

解説 base, port, xдноで指定された Xデバイスの入力信号のデータを1ビット(1点)分読み出します。

例

ユニット名 = A1SX41

ベース番号 = 0

I/Oポート番号 = 1

入力データ (X) =

1FH	18H	17H	10H	0FH	08H	07H	00H
00000000			00001000				0000
0000	00001001						
	00H		08H				00H

09H

data = Xin_bit(hDevice, 0, 1, 3)
(data = 1)

data = Xin_bit(hDevice, 0, 1, 1)
(data = 0)

data = Xin_bit(hDevice, 0, 1, 19)
(data = 1)

1.1.7 Yデバイスの書き込み (1バイト単位)

Yout_byt

機能	Yデバイスの書き込み (1バイト単位)		
形式	Yout_byt Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _ ByVal base As Byte, _ ByVal port As Byte, _ ByVal ydno As Byte, _ ByVal data As Byte) As Long		
	hDevice	:	デバイスハンドル
	base	:	ベース番号 0~7
	port	:	I/Oポート番号 0~7
	ydno	:	Yデバイスの No. 0~63
	data	:	書き込みデータ

base,port , ydnoで指定する値の範囲使用するベース,ユニットにより異なります。

解説 base,port,ydnoで指定されたYデバイスを含む1バイト分の出力データを書き込みます。(dataには常に現在のYデバイスの状態が記憶されていなければなりません。初期化[unitrst]を行うと同時に data をクリアしてください。)

例

ユニット名 = A1SY41

ベース番号 = 0

I/Oポート番号 = 2

ydev(0, 2, 0 ≠ 8) = 10

ret = Yout_byt(hDevice, 0, 2, 0, 10)

出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
 00 00000000 00000000 000000
 00 00001010 00H 00H 00H
 0AH

ydev(0, 2, 0 ≠ 8) = 11

ret = Yout_byt(hDevice, 0, 2, 0, 11)

出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
 00 00000000 00000000 000000
 00 00001011 00H 00H 00H
 0BH

ydev(0, 2, 19 ≠ 8) = 5


```
ret = Yout_byt(hDevice, 0, 2, 19, 5)
出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
              00000000 00000101 00000000
0 00001011
              0 0H 0 5H 0 0H
0BH
```

1.1.8 Yデバイスの リット

Yout_bitrst

機能 Yデバイスの リット

形式 Yout_bitrst Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal base As Byte, _
ByVal port As Byte, _
ByVal ydno As Byte, _
ByRef data As Byte) As Long

hDevice	:	デバイスハンドル	
base	:	ベース 番号	0~ 7
port	:	I/Oポート番号	0~ 7
ydno	:	Yデバイスの No.	0~ 63
data	:	ydno を含む Yデバイス 1バイト分の データ	

base,port , ydnoで 指定する 値の 範囲 使用する ベース ,ユニットに より異なります。

解説 base,port,ydnoで 指定された Yデバイスを リットし、その 内容を dataに 格納します。
(dataには 常に 現在の Yデバイスの 状態が、記憶されていなければなりません。初期化
[unitrst]を行うと同時に data をクリアしてください。)

例 **ユニット名 = A1SY41**
ベース 番号 = 0
I/Oポート番号 = 2

```
ret = Yout_bitrst(hDevice, 0, 2, 0, ydev(0, 2, 0 ≠ 8))
出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
               00000000 00000000 00000000
00 00110010
               00H 00H 00H
32H
```

```
ret = Yout_bitrst(hDevice, 0, 2, 1, ydev(0, 2, 1 ≠ 8))
出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
               00000000 00000000 00000000
00 00110000
               00H 00H 00H
30H
```

1.1.9 Yデバイスのセット

Yout_bitset

機能 Yデバイスのセット

形式 Yout_bitset Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal Board As Byte, _
ByVal port As Byte, _
ByVal ydno As Byte, _
ByRef data As Byte) As Long

hDevice	:	デバイスハンドル	
base	:	ベース番号	0~7
port	:	I/Oポート番号	0~7
ydno	:	Yデバイスの No.	0~63
data	:	ydno を含む Yデバイス 1バイト分の データ	

base,port , ydnoで指定する値の範囲 使用するベース ,ユニットにより異なります。

解説 base,port,ydnoで指定されたYデバイスをセット、その内容を dataに格納します。
(dataには常に現在のYデバイスの状態が記憶されていなければなりません。初期化
[unitrst]を行うと同時に dataをクリアしてください。)

例

ユニット名 = A1SY41

ベース番号 = 0

I/Oポート番号 = 2

ret = Yout_bitset(hDevice, 0, 2, 1, ydev(0, 2, 1 ≠ 8))

出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
00000000 00000000 00000000
00 10111010
00H 00H 00H

BAH

ret = Yout_bitset(hDevice, 0, 2, 17, ydev(0, 2, 17 ≠ 8))

出力データ = 1FH 18H 17H 10H 0FH 08H 07H 00H
00000000 00000010 00000000
00 10111010
00H 02H 00H

BAH

1.1.10 バッファメモリの読み出し

From

機能 バッファメモリの読み出し

形式 From Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal base As Byte, _
ByVal port As Byte, _
ByVal add As Integer, _
ByRef data As Integer, _
ByVal word As Integer, _
ByRef ddata As Byte) As Long

hDevice	:	デバイスハンドル
base	:	ベース番号 0~7
port	:	I/Oポート番号 0~7
add	:	バッファメモリアドレス
data	:	読み出しデータ格納バッファ
word	:	読み出しワード(16ビット数)
ddata	:	Y18 ~ Y1Fの1バイト分の格納バッファ
戻り値	:	1:正常 0:異常

base,portで指定する値の範囲は使用するベースにより異なります。

解説 base,portで指定されたユニットから addで示されるバッファメモリの内容を word分だけ読み出し dataに格納します。読み出しワード数のMax値は32767ワードです。ただし読み出しデータ格納バッファサイズ以上を設定しないでください。

例

ユニット名 = A1S64AD

ベース番号 = 0

I/Oポート番号 = 2

```
ret = From(hDevice, 0, 2, 10, rdbuf(0), 4, ydev(0, 2, &H1F ¥ 8))
```

rdbuf(0)~rdbuf(3)に、ch1~ch4のA/D変換された値が読み出されます。

1.1.11 バッファメモリ書き込み

To_cmd

機能 バッファメモリ書き込み

形式 To_cmd Lib "WDPLCPCI.DLL" (ByVal hDevice As Long, _
ByVal base As Byte, _
ByVal port As Byte, _
ByVal add As Integer, _
ByRef data As Integer, _
ByVal word As Integer, _
ByRef ddata As Byte) As Long

hDevice	:	デバイスハンドル
base	:	ベース番号 0~7
port	:	I/Oポート番号 0~7
add	:	バッファメモリアドレス
data	:	書き込みデータ格納バッファ
word	:	書き込みワード(16ビット数)
ddata	:	Y18 ~ Y1Fの1バイト分の格納バッファ
戻り値	:	1:正常 0:異常

base,portで指定する値の範囲は使用するベースにより異なります。

解説 base,portで指定されたユニットの addで示されたアドレスに dataの内容を書き込みます。
書き込みワード数の Max値は 32767ワードです。

例 ユニット= A1S62DA

ベース番号 = 0

I/Oポート番号 = 2

wrbuf(0)=128;

wrbuf(1)=9029;

ret = To_cmd(hDevice, 0, 2, 2, wrbuf(0), 2, ydev(0, 2, &H1F ¥ 8))

**A1S62DAの バッファメモリ “1”と“2”に デジタル値 128,9029
を書き込みます。**