

End Type

'API 関数宣言

```
Public Declare Function CTDSetIpParameter Lib "CTD20" _  
(ByVal wBsn As Integer, ByVal pDriveParameter As CTIPDRIVEPARAMETER, ByVal wIndex As Integer) As Long
```

VB2005 では以下の様に定義します。

VB2005

クラスファイルを使った場合の例 (プロジェクトの中に、Form ファイルとモジュールファイルを用意します。)

モジュールファイル内

```
Imports System.Runtime.InteropServices '必要
```

Module CT

```
Public Declare Function CTSetIpParameter Lib "CTD20" Alias "CTDSetIpParameter" _  
(ByVal wBsn As UInt16, <MarshalAs(UnmanagedType.Struct)> ByVal pDriveParameter As CTIPDRIVEPARAMETER, ByVal windex As UInt32) As UInt32  
<StructLayout(LayoutKind.Sequential)> Public Structure CTIPDRIVEPARAMETER  
<VBFixedArray(1, 4), MarshalAs(UnmanagedType.ByValArray, SizeConst:=4)> Dim blpAxis(.) As Byte  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim blpKind() As Byte  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim bMovekind() As Byte  
<VBFixedArray(1, 2), MarshalAs(UnmanagedType.ByValArray, SizeConst:=2)> Dim lObjPoint(.) As UInt32  
<VBFixedArray(1, 2), MarshalAs(UnmanagedType.ByValArray, SizeConst:=2)> Dim lCenterPoint(.) As UInt32  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim IN2Data1() As UInt32  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim IN2Data2() As UInt32  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim IN2Data3() As UInt32  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim blpconsin() As Byte  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim bEndkind() As Byte  
<VBFixedArray(1), MarshalAs(UnmanagedType.ByValArray, SizeConst:=1)> Dim bDrawIn() As Byte  
Public Sub Initialize()  
    ReDim blpAxis(0, 0 To 3)  
    ReDim blpKind(0)  
    ReDim bMovekind(0)  
    ReDim lObjPoint(0, 0 To 1)  
    ReDim lCenterPoint(0, 0 To 1)  
    ReDim IN2Data1(0)  
    ReDim IN2Data2(0)  
    ReDim IN2Data3(0)  
    ReDim blpconsin(0)  
    ReDim bEndkind(0)  
    ReDim bDrawIn(0)  
End Sub
```

End Structure

End Module

フォームファイル内

Public Class Form1

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click  
    CTD.Initialize() '構造体配列の初期化  
    CTD.blpAxis(0, 0) = 1 '1軸X軸指定  
    CTD.blpAxis(0, 1) = 1 '2軸X軸指定  
    (省略)  
    If CTSetIpParameter(wBsn, CTD, 0) = 0 Then Exit Sub 'パラメータ設定  
End Sub
```

End Class

4.VB6 AddressOf 演算子

VB2005 では Delegate を使用しないと VB6 AddressOf と同等な機能は使用できません。
以下の例を参考に記述する必要があります。

例.フォーム内に以下を記述します。

```
Public Class DelegateSample
{
    'BSN=0 Axis=0 の場合
    Private Delegate Sub sampleCallDelegate()
    Private workerDelegate As sampleCallDelegate

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        workerDelegate = New sampleCallDelegate(AddressOf ExeEnd)
        Call doWork()
    End Sub

    Private Sub dowork()
        dim wBsn as Short
        dim wAxis as Short
        wBsn = 0
        wAxis = 0

        If xxxwDllOpen() = False Then Exit Sub           '必要な製品のみ記述します
        If xxxwCreate(wBsn) = False Then Exit Sub        'PCI 製品の場合は" False" と記述します
                                                    'USB 製品の場合は" -1" と記述します

        If xxxwMode1Write(wBsn, wAxis, &H40) = False Then Exit Sub    'MODE1 設定
        If xxxwMode2Write(wBsn, wAxis, &H3F) = False Then Exit Sub    'MODE2 設定
        If xxxwDataHalfWrite(wBsn, wAxis, 0, 100) = False Then Exit Sub 'RANGE DATA 設定
        If xxxwDataHalfWrite(wBsn, wAxis, 2, 100) = False Then Exit Sub 'START/STOP SPEED DATA 設定
        If xxxwDataHalfWrite(wBsn, wAxis, 4, 2000) = False Then Exit Sub 'OBJECT SPEED DATA 設定
        If xxxwDataHalfWrite(wBsn, wAxis, 6, 4000) = False Then Exit Sub 'RATE-1 DATA 設定
        If xxxwDataFullWrite(wBsn, wAxis, &H20, 200000) = False Then Exit Sub '+PRESET PULSE DRIVE 実行

        While 1                                           'BUSY がOFF になるまでのループ
            If xxxwGetDriveStatus(wBsn, wAxis, b) = False Then Exit Sub 'DriveStatus 取得
            If (b And 1) = 0 Then
                workerDelegate()
                Exit While
            End If
            System.Windows.Forms.Application.DoEvents()
        End While
    End Sub

    Private Sub ExeEnd()
        If xxxwClose(0) = False Then Exit Sub
        If xxxwDllClose() = False Then Exit Sub          '必要な製品のみ記述します
    End Sub
}
```

5.コンバート機能を使用する場合の注意点

VB6 で作成したプロジェクトを VB2005 で使用する場合、アップグレードウィザードを使用すると簡単に変換できる為非常に便利ですが、正常に変換されずにエラーとなる場合がありますので、ご注意ください。

Visual C++ 6 から Visual C 2005 へ移行する際の注意点

2007/9/4

Visual C++ 6(以降 VC6) から Visual C++ 2005(以降 VC++2005)へ移行する際の注意点は以下の通りです。

1.CRT 関数宣言

VC++2005 では、多くの C ランタイム関数(strcpy など)が、「推奨されない関数」としてコンパイル時に警告”warning C4996 …”が発生するようになりました。

殆どの「推奨されない関数」は利用する事は可能ですが、非常に多くの警告が発生する為、デバッグ作業に支障がでる場合があります。

この状態を回避する方法として以下の方法があります。

セキュリティ強化版に変更

殆どの「推奨されない関数」は、新関数に移行する事で回避できます。多くの新関数は、旧来の関数の後ろに”_s”を付加した記述になります。

例)

Strcpy

↓

Strcpy_s

特定の警告を無効にする方法

警告” warning C4996”を無効にする事で、回避する事もできます。

```
#pragma warning (disable : 4996)
```

もしくは

```
#define _CRT_SECURE_NO_DEPRECATED
```