



Development Guide & Tutorial

Getestet? Du willst es?

Angebot

Die folgenden Codeausschnitte bedeuten, dass Sie die Datei [Helper.bas](#) verwenden, die in den Samples enthalten ist und mit der Installation der Bibliothek installiert wird.

Device Provider

Verwenden Sie einen der verschiedenen SPS-Geräteanbieter, indem Sie eine Instanz der entsprechenden PlcDevice-Klassenableitungen instanziiieren.

```
Dim device As PlcDevice
Set device = New_SiemensDevice(New_IPDeviceEndPoint("192.168.0.80"))
```

Device Konfiguration

Nachdem eine Instanz von (z. B.) die SiemensDevice-Klasse erstellt wurde, verwenden Sie einfach die von der Klasse bereitgestellte Property, um die entsprechenden Geräte-Metadaten einzurichten.

```
Dim device As SiemensDevice
Set device = New_SiemensDevice(New_IPDeviceEndPoint("192.168.0.80"))

device.Type = SiemensDeviceType_S71200
device.ChannelType = SiemensChannelType_OperationPanel
```

In den meisten Szenarien ist es schon genug, um ein Gerät mit dem entsprechenden Gerätetyp im Konstruktor des Geräts zu erstellen, da der Default-Kanal meistens den gemeinsamen Anforderungen entspricht.

```
Dim device As SiemensDevice
Set device = New_SiemensDevice(New_IPDeviceEndPoint("192.168.0.80"),
SiemensDeviceType_S71200)
```

Device End Point

Das Framework bietet die Möglichkeit, verschiedene Typen von end points über die PlcDeviceEndPoint-Klasse zu definieren. Die am häufigsten verwendete End point-Implementierung ist die IPDeviceEndPoint-Klasse. Mit dieser Klasse können Sie die IP-Adresse und optional die Rack- und Steckplatznummer der SPS angeben.

```
Dim endPoint As PlcDeviceEndPoint
Set endPoint = New_IPDeviceEndPoint("192.168.0.80")

' Alternatively also specify the rack.
Dim endPointWithRack As PlcDeviceEndPoint
Set endPointWithRack = New_IPDeviceEndPoint("192.168.0.80", 0)

' Alternatively also specify the rack and slot.
Dim endPointWithRackSlot As PlcDeviceEndPoint
Set endPointWithRackSlot = New_IPDeviceEndPoint("192.168.0.80", 0, 2)
```

Device Connection

Nach dem Erstellen einer Instanz eines der PlcDevice-Klassen-Derivate erstellen Sie einfach eine neue Verbindung, die mit dem dargestellten Gerät verknüpft ist. Wenn Sie eine Instanz mit dieser Factory-Methode erstellen, wird die vom Provider abhängige Implementierung der PlcDeviceConnection-Klasse zurückgegeben.

```
Dim device As PlcDevice
Set device = New_SiemensDevice(New_IPDeviceEndPoint("192.168.0.80"))

Dim connection As PlcDeviceConnection
Set connection = device.CreateConnection()
```

Verbindungsstatus behandeln

Um den aktuellen Konnektivitätsstatus der PlcDeviceConnection-Klasseninstanz abzurufen, verwenden Sie die State-Property, um das entsprechende PlcDeviceConnectionState-Enumerationmember zu erhalten.

```
If (connection.State = PlcDeviceConnectionState_Connected) Then
    ' ...
End If
```

Um die Zustandsübergänge zu behandeln, verwenden Sie eines oder mehrere der Status spezifischen Events der PlcDeviceConnection-Klasseninstanz.

```
Private WithEvents m_connection As PlcDeviceConnection
```

Um das Event zu behandeln, deklarieren Sie den übereinstimmenden Event-Handler wie folgt:

```
Private Sub m_connection_Connected(ByRef sender As Object, ByVal e As IComEventArgs)
    Debug.Print "Connection established!"
End Sub
```

Werte lesen

Verwenden Sie zum Lesen eines einzelnen Werts eine der Read-Methoden der PlcDeviceConnection-Klasse.

```
Dim value As Long
value = connection.ReadInt32("DB1.DB1 1")
```

Um ein Array von Werten zu lesen, verwenden Sie die zusätzlichen Leseüberladungen:

```
Dim values() As Long
values = connection.ReadInt32Array("DB1.DB1 1", 3)
```

Werte schreiben

Um einen einzelnen Wert zu schreiben, verwenden Sie eine der Write-Methoden der PlcDeviceConnection-Klasse.

```
Call connection.WriteInt32("DB1.DBD 1", 123)
```

Um ein Array von Werten zu schreiben, verwenden Sie die zusätzlichen Write-Überladungen:

```
Dim values(3) As Long  
values(0) = 123  
values(1) = 456  
values(2) = 789  
  
Call connection.WriteInt32Array("DB1.DBD 1", values)
```

Inhaltsverzeichnis

Getestet? Du willst es?	1
Device Provider	2
Device Konfiguration	2
Device End Point	2
Device Connection	3
Verbindungsstatus behandeln	3
Werte lesen	3
Werte schreiben	3

