

# W&T OPC-Server für Netzwerk-E/A-Geräte

Version 2.05

Copyright © 1999 - 2001 Wiesemann & Theis GmbH. Basiert auf dem OPC Toolkit von FactorySoft.

Die beigefügte Datei "FSServer.dll" unterliegt dem Copyright © 1997 von FactorySoft, Inc. Sie erhalten sie ausschließlich zur Benutzung als Teil des oben genannten Softwareproduktes, auf Grundlage eines Lizenzvertrages zwischen FactorySoft, Inc. und der Wiesemann & Theis GmbH. Sie sind nicht berechtigt, diese Datei für andere Zwecke zu verwenden oder an Dritte weiterzugeben.

## Allgemeines über OPC-Server

OPC (OLE for Process Control) ist eine Softwareschnittstelle zum Zugriff auf Prozeßdaten, auf Basis der OLE-Technologie von Microsoft. Anwendungsprogramme wie z. B. Visualisierungssysteme, die diese Schnittstelle benutzen, heißen OPC-Clients. Auf der anderen Seite der Schnittstelle stehen ihnen OPC-Server gegenüber. Das sind Gerätetreiber, die eine bestimmte Hardware in abstrakter Form als einen Satz von OPC-Variablen darstellen.

Der hier vorliegende OPC-Server dient der Ansteuerung von W&T Digital-E/A-Servern und seriellen Com-Servern (Produktfamilien 50xxx bzw. 58xxx).

## Installation

Starten Sie das Setup-Programm von der Installationsdiskette (indem Sie z. B. aus dem Startmenü "Ausführen..." wählen und dann eingeben: "A:\Setup"). Es installiert und registriert den OPC-Server auf ihrem Rechner.

Der OLE-Servername, den OPC-Clients später angeben müssen, um sich mit dem Server zu verbinden, lautet: "Wiesemann-Theis.DigitalEA". Der OPC-Server startet auf solche Anforderungen hin automatisch. Um ihn zu konfigurieren, können Sie ihn aber auch von Hand ausführen. Einen entsprechenden Eintrag "W&T OPC-Server" finden Sie unter "Programme" im Startmenü.

**Hinweis:** Unter Windows 95 wird die Installation möglicherweise mit Meldungen über fehlende Funktionen in der OLE32.dll fehlschlagen. Eine aktuellere Version dieser DLL, die das Problem behebt, ist in dem Update-Paket "DCOM für Windows 95" enthalten, das Microsoft zum kostenlosen Download im Internet anbietet, unter der Adresse [http://www.microsoft.com/com/dcom/dcom95/dcom1\\_3.asp](http://www.microsoft.com/com/dcom/dcom95/dcom1_3.asp).

## Deinstallation

Den OPC-Server entfernen können Sie über die Systemsteuerungskomponente "Software". Er steht dort in der Liste unter "W&T OPC-Server für Digital-EA".

## Gerätekonfiguration

Das Dialogfeld "Endgerät" erscheint jedesmal, wenn Sie einen neuen E/A-Server oder Com-Server zur Konfiguration hinzufügen oder die Angaben zu einem existierenden Server nachträglich ändern wollen. Die einzelnen Eingabefelder haben folgende Bedeutung:

**Hostname oder IP-Adresse:** Muß mit der IP-Adresse übereinstimmen, die am Gerät eingestellt wurde (vier Zahlen, durch Punkte getrennt, wie z. B. 172.16.232.77). Falls der Adresse über DNS oder einen ähnlichen Namensdienst auch ein Hostname zugeordnet wurde, können Sie auch diesen Namen statt der IP-Adresse verwenden.

**Servertyp:** "Mit Meßfunktion" meint die E/A-Server-Modelle 50211/50311. Der Typ "seriell mit 2+2 E/A" ist für Com-Server mit dem speziellen Schnittstellenmodul 18804 gedacht.

Einen besonderen Typ wie etwa "seriell mit 12+12 E/A", der die Modelle 50310/50311 beschreiben würde, gibt es *nicht*. Aus Sicht des OPC-Servers ist die serielle Schnittstelle eines E/A-Servers ein eigenständiger Com-Server. Legen Sie stattdessen zwei Server mit derselben IP-Adresse an, von denen der eine ein E/A-Server ist, der andere einfach vom Typ "seriell".

**TCP-Portnummer:** Ist 8000 für serielle Ports (bzw. 8100, 8200, 8300 für die seriellen Ports B, C, D eines Mehrfach-Com-Servers). Die Portnummer zur Ansteuerung der Geräte mit 12+12 E/A ist normalerweise 49153 (voreingestellt ab Werk).

Hinweis: Für jeden seriellen Port werden in Wirklichkeit zwei TCP-Verbindungen geöffnet: Eine Datenverbindung unter der angegebenen Portnummer, sowie eine Steuerverbindung unter einer um 1094 höheren Portnummer, also 9094, 9194, 9294 oder 9394.

**Fortlaufende Gerätenummer:** Diese Nummer wird beim Hinzufügen eines E/A-Servers automatisch zugewiesen und soll garantieren, daß alle Geräte verschiedenen Namen erhalten (Box1, Box2 usw.). Änderungen von Hand können sinnvoll sein, um z. B. den E/A-Ports und dem seriellen Port eines E/A-Servers (s. Anmerkungen zu "Servertyp") dieselbe Gerätenummer zuzuweisen. Die resultierenden Gerätenamen wären dann z. B. "Box1" und "Com1".

**Aktualisierung mindestens alle ... Sek.:** Manche Ereignisse melden die E/A- bzw. Com-Server von sich aus (das Eintreffen von RS232-Empfangsdaten, Änderungen an binären Eingängen der Geräte mit 12+12 E/A), für andere ist man auf regelmäßiges aktives Abfragen angewiesen (u. a. für die Eingänge der seriellen Server mit 2+2 E/A). Den Abstand zwischen zwei solchen Abfragen legt dieser Wert fest, wobei die Eingabe auf ein Vielfaches von 0.1 Sekunden gerundet wird.

Diese regelmäßigen Abfragen dienen außerdem dazu, eine Störung der TCP-Verbindung zum Endgerät möglichst bald zu erkennen. Eine Störung wird dann vermutet (und die Verbindung wird zurückgesetzt), wenn zwischen zwei Anfragen keine Antwort vom Gerät eingetroffen ist. Vorsicht: Ein zu kleiner Wert an dieser Stelle kann darum einen Verbindungsaufbau unmöglich machen. Konkrete Gefahr dazu dürfte allerdings nur in Extremfällen bestehen, z. B. bei einem Intervall von 0.1 Sek., wenn außerdem ein sehr weitläufiges Netzwerk zwischen PC und E/A-Server liegt.

**Deaktiviert:** Zu einem deaktivierten Server wird keine TCP-Verbindung aufgebaut. Die zugehörigen OPC-Variablen werden zwar angelegt, ergeben beim Auslesen aber immer nur OPC\_QUALITY\_NOT\_CONNECTED.

**RS232-Parameter:** Ein Initialisierungsstring wie z. B. "19200,n,8,1,x", der die Schnittstellenparameter beim Start des OPC-Servers festlegt. Bedeutung der Parameter innerhalb des Strings: Baudrate, Paritätsbit (n = keins, e = gerade, o = ungerade), Anzahl Datenbits (5 - 8), Anzahl Stopbits (1 oder 2), Handshakeverfahren (x = Software, XOn/XOff, h = Hardware, RTS/CTS).

Wenn kein Initialisierungsstring angegeben ist, bleiben die aktuellen seriellen Einstellungen des Com-Servers unverändert.

**Textempfang komplett bei...:** Um den empfangenen RS232-Datenstrom als OPC-Variable darstellen zu können, muß er auf irgendeine Weise in Strings, oder anders gesagt: Pakete, zerlegt werden. Paketgrenzen werden entweder anhand von Pausen oder am Auftreten spezieller Zeichenfolgen erkannt.

Wenn eine Zeichenfolge angegeben wird, sollte gleichzeitig die Erkennung von Pausen abgeschaltet werden (durch Eingabe eines Wertes von 0 Sekunden). Eine Kombination der beiden Kriterien ist zwar möglich, aber nicht unbedingt empfehlenswert.

Zeichenfolgen werden als Dezimalzahlen angegeben, niemals direkt als String. Beispielsweise wäre also für drei Pluszeichen als Paketgrenze ("+++") die korrekte Eingabe "43,43,43".

**Beim Empfang ausfiltern:** Das Ausfiltern von Sonderzeichen findet erst statt, nachdem die empfangenen Daten bereits in Pakete zerlegt sind. Ausfiltern von Zeichen, die gleichzeitig Trennzeichen darstellen, ist also zulässig. Null-Bytes werden in jedem Fall ausgefiltert, auch wenn hier nicht ausdrücklich "0" mit angegeben wird, da sie nicht innerhalb von Strings auftauchen dürfen.

**Beim Senden anhängen:** Sinnvoll kann insbesondere das Anhängen von CR (13), LF (10) oder beiden (13,10) als Zeilenendezeichen sein.

## Direktsteuerung über Bedienfelder

Für jeden Server in der Liste können Sie ein Bedienfeld öffnen, das im wesentlichen dieselben Zugriffsmöglichkeiten bietet wie die OPC-Schnittstelle. Insbesondere sind die einzelnen Steuerelemente mit den Namen der entsprechenden OPC-Variablen beschriftet.

Nützlich ist das vor allem, um sich mit dem Verhalten der Endgeräte vertraut zu machen, und um z. B. bei seriellen Schnittstellen die Wirkung der diversen Filteroptionen zu sehen. Für eine echte Fernsteuerung der Geräte ist diese Funktion dagegen vermutlich nicht sinnvoll, da sich nicht mehr als ein Bedienfeld gleichzeitig öffnen läßt.

## OPC-Variablen für E/A-Server

Für jeden E/A-Server, der in die Konfiguration eingefügt wird, legt der OPC-Server die folgenden Variablen an. Mehrere E/A-Server unterscheiden sich dabei von Gerät zu Gerät durch den ersten Teil des Namens: Box1, Box2, usw.

**Box1.E.0 — Box1.E.11** (VT\_BOOL, R): Zustand der Eingänge.

**Box1.A.0 — Box1.A.11** (VT\_BOOL, R/W): Zustand der Ausgänge.

Darüber hinaus gibt es diese Variablen nur für E/A-Server mit Meßfunktion:

**Box1.I.0 — Box1.I.11** (VT\_R4, R): Ausgangsströme, in Ampere.

**Box1.T** (VT\_R4, R): Gerätetemperatur, in Grad Celsius.

**Box1.U** (VT\_R4, R): Versorgungsspannung, in Volt.

**Box1.F** (VT\_BOOL, R/W): Fehler, Ausgänge wegen Überlast abgeschaltet. Zurücksetzen der Variable schaltet wieder ein.

## Abkürzungen

OLE-Datentypen:

VT\_BOOL: Binärwert

VT\_I2, VT\_I4: Ganzzahl

VT\_R4: Fließkommazahl

VT\_BSTR: Zeichenkette

Zugriffsrechte:

R/W: lesen und schreiben

R: nur lesen

W: nur schreiben

## OPC-Variablen für serielle Com-Server

Die Gerätenamen für serielle Com-Server lauten Com1, Com2, usw., und die Variablen im einzelnen sind:

**Com1.TxD** (VT\_BSTR, W): RS232-Sendedaten, zugewiesene Werte werden über die serielle Schnittstelle ausgegeben.

**Com1.RxD** (VT\_BSTR, R): RS232-Empfangsdaten (das zuletzt von der seriellen Schnittstelle eingetroffene Textpaket).

**Com1.N** (VT\_I4, R/W): Paketzähler, wird bei jedem Empfang eines Textpakets um 1 erhöht.

Außerdem, bei seriellen Ports mit 2+2 E/A (am sinnvollsten für Ports mit speziellem E/A-Modul 18804, aber auch gewöhnliche COM-Ports lassen sich so ansteuern):

**Com1.E.0** (VT\_BOOL, R): binärer Eingang (DSR, bzw. Pin 4 am E/A-Modul).

**Com1.E.1** (VT\_BOOL, R): binärer Eingang (CTS, bzw. Pin 7 am E/A-Modul).

**Com1.A.0** (VT\_BOOL, R/W): binärer Ausgang (DTR, bzw. Pin 6 am E/A-Modul).

**Com1.A.1** (VT\_BOOL, R/W): binärer Ausgang (RTS, bzw. Pin 8 am E/A-Modul).

Die folgenden Variablen geben lediglich Zugriff auf die Schnittstellenparameter, wie sie auch über den RS232-Initialisierungsstring bei der Gerätekonfiguration eingestellt werden können:

**Com1.Line.Baud** (VT\_I4, R/W): Baudrate, in Bit pro Sekunde.

**Com1.Line.Bits** (VT\_I2, R/W): Anzahl Datenbits pro Zeichen (5, 6, 7 oder 8).

**Com1.Line.Stop** (VT\_I2, R/W): Anzahl Stopbits (1 oder 2).

**Com1.Line.Parity** (VT\_I2, R/W): Paritätsbit: 0 = keins, 1 = ungerade, 2 = gerade Parität

**Com1.Line.Flow** (VT\_BOOL, R/W): soll ein Handshake-Protokoll zur seriellen Flußkontrolle verwendet werden?

**Com1.Line.Hard** (VT\_BOOL, R/W): Hardware- oder Software-Handshake? (Bei COM-Ports, die für 2+2 E/A konfiguriert sind, ist diese Variable nicht vorhanden, und es ist nur Software-Handshake möglich.)

## Ein E/A-Server, mehrere Client-Rechner

Die TCP-Verbindung, die der OPC-Server zu einem ihm zugeordneten E/A-Server oder Com-Server aufbaut, gibt ihm exklusiven Zugriff auf das Gerät. Eine Konfiguration, in der OPC-Server von mehreren Rechnern aus auf ein- und denselben E/A-Server zugreifen, ist darum nicht sinnvoll und würde nicht zuverlässig funktionieren.

Einen vergleichbaren Effekt erzielt man stattdessen mit einem einzelnen OPC-Server, auf den Clients von mehreren Rechnern aus zugreifen, über DCOM (Distributed COM). Beachten Sie nur, daß Sie für die Benutzung von DCOM erst noch Zugriffsrechte erteilen müssen. Das geschieht mit dem Programm "DcomCnfg.exe", das zu Windows NT 4.0 standardmäßig dazugehört.

Um DCOM auch unter Windows 95 benutzen zu können, müssen Sie erst die Update-Pakete DCOM95.EXE und DCM95CFG.EXE installieren. (Ob das bereits geschehen ist, können Sie feststellen, indem Sie von der Eingabeaufforderung aus versuchen, DcomCnfg.exe auszuführen.) Setzen Sie dann noch in der Registry den Wert HKEY\_LOCAL\_MACHINE\Software\Microsoft\OLE\EnableRemoteConnect auf "Y". Und beachten Sie, daß es unter Windows 95 nicht möglich ist, einen OLE-Server auf einen DCOM-Aufruf von außerhalb hin starten zu lassen. Der OPC-Server müßte auf einem Win95-Rechner also von Hand gestartet werden, bevor sich OPC-Clients von anderen Rechnern aus mit ihm verbinden können.

Ausführliche Informationen zum Thema DCOM finden Sie auch in der Microsoft Knowledge Base, u. a. in den Artikeln:

Q176799, INFO: Using DCOM Config (DCOMCNFG.EXE) on Windows NT  
Q165101, HOWTO: Use Win95 as a DCOM Server  
Q182248, HOWTO: Use DCOM Config (DCOMCNFG.EXE) with Windows 95  
Q158508, FAQ: COM Security Frequently Asked Questions  
Q174024, FAQ: DCOM95 Frequently Asked Questions

Diese Artikel finden Sie auf der Homepage von Microsoft: Entweder indem Sie dort die Suchmaschine benutzen, <http://search.microsoft.com>, und als Suchbegriff eine Artikelnummer wie "Q165101" eingeben sowie als Kategorie "Support & Knowledge Base" auswählen. Oder direkt über eine URL der Form <http://support.microsoft.com/support/kb/articles/q165/1/01.asp>.

## Zugriff über Visual Basic bzw. VBA

Dieser OPC-Server bietet, abgesehen von dem grundlegenden Custom-Interface, nur ein Automatisierungs-Interface nach dem neuen OPC-Standard 2.0 an. Falls Sie mit dem alten OPC Automatisierungs-Standard 1.0 vertraut sind: Die Unterschiede zwischen den beiden Versionen sind erheblich. Außerdem ist zur Benutzung des neuen Interface mindestens Visual Basic Version 5.0 bzw. Office 97 erforderlich. Die komplette Schnittstellen-Spezifikation finden Sie auf der Homepage der OPC Foundation, <http://www.opcfoundation.org>. Für einen ersten Eindruck sind aber auch vielleicht die folgenden Beispiele (Excel-Makros) hilfreich.

Beispiel 1 benutzt den Namensraum-Browser, um alle vorhandenen Variablen des OPC-Servers zu ermitteln, und trägt sie in Spalte 1 der aktuellen Excel-Tabelle ein.

Beispiel 2 liest für alle Variablennamen, die es in Spalte 1 der aktuellen Excel-Tabelle vorfindet, die Eigenschaften "Wert", "Maßeinheit" und "Beschreibung", und trägt sie in die benachbarten Spalten ein.

Eine besondere Bedeutung hat dabei die Eigenschaft "Signalqualität", die im wesentlichen aussagt, ob der OPC-Server einen gültigen Wert für eine bestimmte Variable hat. Ein möglicher und leider gar nicht so unwahrscheinlicher Grund für Probleme in dieser Hinsicht kann sein, daß der OPC-Server gerade erst gestartet ist (automatisch aufgrund der Anforderung durch das Makro), und noch keine TCP-Verbindung zu seinen Endgeräten aufbauen konnte.

Beispiel 3 liest in der aktuellen Excel-Tabellenzeile aus Spalte 1 einen Variablennamen, aus Spalte 2 einen Wert, und weist den Wert an die Variable zu.

Anders als das Auslesen von Werten ist dieser Vorgang nicht über die "Item Properties" des OPC-Servers realisierbar, sondern nur direkt über das entsprechende OPCItem-Objekt. Übrigens sollte aus Effizienzgründen normalerweise auch das Lesen von Variablen auf diese Weise erfolgen, und nicht mit der in Beispiel 2 gezeigten Methode. (Und anders als in diesem Makro würde man die OPCItem-Objekte normalerweise auch nicht für jeden Zugriff erst anlegen und dann sofort wieder freigeben.)

Damit Visual Basic das OPC-Interface benutzen kann, muß in der Liste der Verweise der Eintrag "OPC Automation 2.0" aktiviert sein. (Zu dieser Liste gelangen Sie bei Visual Basic 6.0 über den Menüpunkt "Projekt/Verweise...", bei Excel 97 über den Menüpunkt "Extras/Verweise..." im Visual Basic Editor.)

## Beispiel 1

Option Base 1

```
Sub OpcGetNames()  
    ' Trägt die verfügbaren Variablennamen in Spalte 1 ein  
    ' Wir haben es mit einem hierarchischen Namensraum zu tun,  
    ' max. 2 Ebenen tief.  
    Dim TheOpcServer As OPCServer  
    Dim MyBrowser As OPCBrowser  
    Set TheOpcServer = New OPCServer  
    TheOpcServer.Connect ("Wiesemann-Theis.DigitalEA")  
    Set MyBrowser = TheOpcServer.CreateBrowser  
  
    Dim RootName, BranchName As String  
    Dim i, j, k, n As Integer  
    ' Inhalt von Spalte 1 zunächst löschen.  
    Columns(1).ClearContents  
    n = 0 ' Anzahl gefundene Variablen  
    MyBrowser.ShowBranches  
    For i = 1 To MyBrowser.Count  
        RootName = MyBrowser.Item(i)  
        MyBrowser.MoveDown (RootName)  
        ' Die Blätter direkt am Stamm  
        MyBrowser.ShowLeafs  
        For j = 1 To MyBrowser.Count  
            n = n + 1  
            Cells(n, 1) = RootName + "." + MyBrowser.Item(j)  
        Next j  
        ' In die Unterzweige abtauchen  
        MyBrowser.ShowBranches  
        For j = 1 To MyBrowser.Count  
            BranchName = MyBrowser.Item(j)  
            MyBrowser.MoveDown (BranchName)  
            ' Die untersten Blätter  
            MyBrowser.ShowLeafs  
            For k = 1 To MyBrowser.Count  
                n = n + 1  
                Cells(n, 1) = RootName + "." + BranchName + "." +  
MyBrowser.Item(k)  
            Next k  
            MyBrowser.MoveUp  
            MyBrowser.ShowBranches  
        Next j  
        MyBrowser.MoveUp  
        MyBrowser.ShowBranches  
    Next i  
  
    Set MyBrowser = Nothing  
    TheOpcServer.Disconnect  
    Set TheOpcServer = Nothing  
End Sub
```



## Beispiel 2

Option Base 1

```
Sub OpcUpdate()  
    ' Fragt zu allen Variablenennamen aus Spalte 1 Beschreibung  
    ' und aktuellen Inhalt ab  
    Dim TheOpcServer As OPCServer  
    Set TheOpcServer = New OPCServer  
    TheOpcServer.Connect ("Wiesemann-Theis.DigitalEA")  
  
    Dim PropertyIDs(5) As Long  
    Dim Data() As Variant  
    Dim Errors() As Long  
    Dim i, j As Integer  
    PropertyIDs(1) = 3      ' OPC_PROP_QUALITY  
    PropertyIDs(2) = 2      ' OPC_PROP_VALUE  
    PropertyIDs(3) = 100    ' OPC_PROP_UNIT  
    PropertyIDs(4) = 101    ' OPC_PROP_DESC  
    PropertyIDs(5) = 4      ' OPC_PROP_TIME  
    Columns("B").ClearContents  
    Columns("E:F").ClearContents  
    i = 1  
    While Cells(i, 1) <> ""  
        TheOpcServer.GetItemProperties Cells(i, 1), 5, PropertyIDs, Data,  
Errors  
        For j = 2 To 5  
            Cells(i, j) = Data(j)  
        Next j  
        If Data(1) = 20 Then      ' OPC_QUALITY_LAST_KNOWN  
            Cells(i, 6) = "VERALTET"  
        ElseIf Data(1) <> 192 Then ' OPC_QUALITY_GOOD  
            Cells(i, 6) = "FEHLER"  
            Range(Cells(i, 2), Cells(i, 3)).ClearContents  
        End If  
        n = n + 1  
    Wend  
  
    TheOpcServer.Disconnect  
    Set TheOpcServer = Nothing  
End Sub
```

### Beispiel 3

Option Base 1

```
Sub OpcWrite()  
    ' Für die aktuelle Zeile dem Variablennamen (erste Spalte) einen  
    ' Wert (aus der zweiten Spalte) zuweisen.  
    Dim TheOpcServer As OPCServer  
    Dim MyGroup As OPCGroup  
    Dim MyItem As OPCItem  
    Set TheOpcServer = New OPCServer  
    TheOpcServer.Connect ("Wiesemann-Theis.DigitalEA")  
    Set MyGroup = TheOpcServer.OPCGroups.Add("group")  
  
    Set MyItem = MyGroup.OPCItems.AddItem(Cells(ActiveCell.Row, 1), 1234)  
    MyItem.Write (Cells(ActiveCell.Row, 2))  
  
    TheOpcServer.OPCGroups.Remove (MyGroup.ServerHandle)  
    TheOpcServer.Disconnect  
    Set TheOpcServer = Nothing  
End Sub
```