



TAPPERTZHOFEN

rs232server

Internet RS232Server
mit integrierter Script Engine

rs232server (Deutsch)
Rev 2
Oct. 2005

www.fdos.de/rs232server

rs232server

Rev 2 – Oct. 2005

© 2004 – 2005 EDV TAPPERTZHOFEN; www.fdos.de/rs232server

EINFÜHRUNG.....	5
EINRICHTUNG.....	6
Wahl des LAN/WAN Ports.....	6
RS232 Port und Settings.....	6
Log Bericht.....	6
404-Fehler Seiten.....	7
HTTP Pfad.....	7
Nur interne IPs.....	7
Standard Config.ini.....	7
SCRIPT ENGINE.....	8
VBScript.....	8
JScript.....	9
RS232 Modul.....	10
Server Modul.....	10
Response Modul.....	12

Einführung

Der RS232Server ist eine Softwarelösung um eine Schnittstelle zwischen einer seriellen RS232 Verbindung und dem Internet oder Intranet herzustellen.

Der Server basiert dabei nicht auf einem festen Protokoll zum Austausch von Daten zwischen dem RS232 Netz und dem Internet, sondern verfügt über eine eigene Script Engine, die mit einer Reihe von Standard Funktionen den Datenaustausch ermöglicht. Der Benutzer ist daher in der Lage die Kommunikation selbst zu gestalten und flexibel anzupassen. Als Scriptsprachen bietet der Server Microsoft ® VBScript und JScript an.

Nach dem Start des Servers minimiert sich das Programm und versteckt sich im System Tray (als Weltkugel neben der Uhr). Den Server können Sie durch einen Rechtsklick auf den Globus und dann „Beenden“ herunterfahren.

Nach der Installation ist der Server so eingerichtet, dass Sie ein kleines Beispiel ausprobieren können.

Einrichtung

Nach der Installation ist der Server weitestgehend fertig eingerichtet und kann gestartet werden. Für manuelle Eingriffe kann die im Installationsordner befindliche „config.ini“ verändert werden.

Wahl des LAN/WAN Ports

Standardmäßig kann der Server mit einem Webbrowser über den Port 80 angesprochen werden. Das ist auch der normale HTTP Port. In manchen Anwendungsfällen ist sinnvoll einen anderen Port zu verwenden (z.B. 8080). Ändern Sie hierfür die config.ini. Im Abschnitt „[server]“ befindet sich die Einstellung „port“. Den hinter dem Gleichzeichen folgenden Wert können Sie frei einstellen, um den LAN/WAN Port manuell zu wählen.

RS232 Port und Settings

Mit großer Wahrscheinlichkeit werden Sie gezwungen sein die RS232 Port und Settings Einstellungen zu ändern. Geben Sie hierfür die passenden Daten im Abschnitt „[rs232]“ für „port“ und „settings“ in der config.ini ein.

Log Bericht

Der Server kann bei Bedarf einen Log Bericht (log.txt) erstellen, in dem alle Zugriffe dokumentiert werden. In der config.ini kann die Erzeugung des Log Berichts mit dem Wert 1 bzw. 0 im Abschnitt „[log]“, „dologs“ eingestellt bzw. abgestellt werden.

404-Fehler Seiten

Der Webserver bietet Ihnen die Möglichkeit eigene 404-Fehler Seiten zu erstellen. Erstellen Sie hierfür im Installationsordner eine „404.htm“ mit beliebigem Text.

HTTP Pfad

Sie können den Hauptverzeichnispfad für den Webserver unter „[server]“, „path“ in der config.ini frei wählen. Standardmäßig nimmt der Server das Installationsverzeichnis.

Nur interne IPs

Unter „[server]“, „internaliponly“ in der config.ini können Sie mit einer 1 bzw. einer 0 bestimmen, ob der Server nur Anfragen aus dem internen Netz beantworten soll oder nicht.

Standard Config.ini

```
[server]
port = 80
path = examples\
onlyinternalip = 0
listenforscript = 1
jspport = 8001
vbport = 8002
[rs232]
port = 2
settings = 9600,N,8,1
[log]
dolog = 1
```

Script Engine

Mit der eingebauten Script Engine können Sie sowohl in VBScript als auch JScript programmieren. Die Konventionen für die Script Sprachen richten sich ausschließlich nach den Vorgaben von Microsoft und wurden nur über einige spezifische Funktionen erweitert. Scripte können nur innerhalb von „.htm“ Dateien eingebettet werden. Grundsätzlich sind einzeilige Befehle wie in PHP oder ASP nicht möglich.

Die beiden Script Sprachen sind zwar identisch mit den im Internet verbreiteten VBScript und JScript, werden hier jedoch Serverseitig ausgeführt, so dass sie für den Client nicht sichtbar sind.

VBScript

VBScript ist weitestgehend identisch mit Visual Basic™ oder Visual Basic of Application™. Eingeleitet wird ein VBScript Block mit „<%“ und beendet mit „%>“.

Beispiel:

```
<html><head><title>AD[1]</title></head>
<body>
AD[1]: <%

rs232.portopen
rs232.SendByte 4
rs232.SendByte 1
response.Send asc(rs232.GetData)
rs232.portclose

%>
</body></html>
```

Der Server kann optional auch per TCP/IP VBScript empfangen, ausführen und das Resultat zurücksenden. In der „config.ini“ können Sie unter „server“, „vbport“ den TCP/IP Port wählen (Standard 8002), mit dem der Server Scriptcode empfangen soll. Der Scriptcode ähnelt dem des normalen VB Scriptcodes eingeschlossen in „<%“ und „%>“ jedoch ohne Script Tags und HTML Code.

JScript

JScript ist weitestgehend identisch Java Script. Eingeleitet wird ein JScript Block mit „<?“ und beendet mit „?>“.

Beispiel:

```
<html><head><title>AD[1]</title></head>
<body>
AD[1]: <?

RS232.PortOpen();
RS232.SendByte(4);
RS232.SendByte(1);
Response.Send(RS232.GetData());
RS232.PortClose;

?>
</body></html>
```

Der Server kann optional auch per TCP/IP JScript empfangen, ausführen und das Resultat zurücksenden. In der „config.ini“ können Sie unter „server“, „jsport“ den TCP/IP Port wählen (Standard 8001), mit dem der Server Scriptcode empfangen soll. Der Scriptcode ähnelt dem des normalen JS Scriptcodes eingeschlossen in „<?“ und „?>“ jedoch ohne Script Tags und HTML Code.

RS232 Modul

Public TimeOut As Long

Variable zum Einstellen der max. Wartezeit.

Public Sub SendData(ByVal strString As String)

Sendet einen String über RS232.

Public Sub PortOpen()

Öffnet eine Verbindung.

Public Sub PortClose()

Schließt eine Verbindung.

Public Sub SendByte(ByVal byt As Byte)

Sendet ein Byte über RS232.

Public Function GetData() As Variant

Wartet die max. Wartezeit auf Daten von der seriellen Schnittstelle und gibt diese ggf. zurück.

Public Function ASCII2String(ByVal byt As Byte) As String

Wandelt ein ASCII Byte in einen String um.

Server Modul

Public Function GetVersion() As String

Gibt die Version des Servers zurück.

Public Function GetMajor() As String

Gibt die Hauptversion des Servers zurück.

Public Function GetMinor() As String

Gibt die Nebenversion des Servers zurück.

Public Function GetRevision() As String

Gibt die Revision des Servers zurück.

Public Function GetExeName() As String

Gibt den EXE Namen des Servers zurück.

Public Function GetCompanyName() As String

Gibt den Hersteller des Servers zurück.

Public Function GetPath() As String

Gibt den Programmpfad des Servers zurück.

Public Function GetProductName() As String

Gibt den Produktnamen des Servers zurück.

Public Function GetQuery() As String

Liefert die Argumente im Query String (Werte, die mit „?“ in der Adresszeile übergeben werden).

Public Function GetQueryArg(ByVal lngArgIndex As Long) As String

Liefert ein Argument eines Index aus dem Query String (Werte, die mit „?“ in der Adresszeile übergeben und durch „&“ getrennt werden).

Public Function GetQueryValue(ByVal Name As String) As String

Liefert den Wert eines Argument mit dem übergebenen Namen aus dem Query String (Werte, die mit „?“ in der Adresszeile übergeben und durch „&“ getrennt werden).

Public Function ServerDateTime() As String

Gibt Datum und Zeit des Servers zurück.

Public Function ServerTime() As String

Gibt Zeit des Servers zurück.

Public Function ServerDate() As String

Gibt Datum des Servers zurück.

Public Function GetLocalIP() As String

Gibt die Lokale IP des Servers zurück.

Public Function GetTimer() As String

Gibt den Timer des Servers zurück.

Public Sub RegDLL(ByVal DLLName As String)

Registriert eine ActiveX DLL

Public Sub UnRegDLL(ByVal DLLName As String)

Löscht die Registrierung einer ActiveX DLL

Public Sub ExecuteJScript(ByVal strCode As String)

Führt JScript Code aus

Public Sub ExecuteVBScript(ByVal strCode As String)

Führt VBScript aus

Response Modul

Public Sub Send(ByVal strString As String)

Gibt einen String auf der Webseite aus.

Public Sub SendLine(ByVal strString As String)

Gibt einen String mit nachfolgendem Zeilenumbruch auf der Webseite aus.

Public Sub ClearDocument()

Löscht den Inhalt des kompletten Dokuments.

Kontakt

EDV TAPPERTZHOFEN
Heinsenstraße 32
DE 40221 Düsseldorf

Weitere Informationen:

<http://www.fdos.de>

Alle Informationen in dieser Dokumentation wurden nach besten Wissen zusammengestellt und aufgearbeitet. Eventuelle Verbesserungen oder Veränderungen jeglicher Art können ohne vorherige Ankündigung vorgenommen werden.

Es wird keinerlei Haftung für eventuelle Schäden übernommen.

Jede Art von nicht erlaubter elektronischer Vervielfältigung der Dokumentation oder Auszüge der Dokumentation können strafrechtlich verfolgt werden.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zur Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürfen

Copyright © 2004 - 2005 EDV Tappertzhofen; Alle Rechte vorbehalten.



TAPPERTZHOFEN