

**AR WebServer**



# Contents (9.7.2005)

<b>1 B&amp;R Web Server.....</b>	<b>1</b>
1.1 Web Server Installation .....	1
1.1.1 Web Server – System Requirements.....	1
1.1.1.1 Operating System Version.....	1
1.1.1.2 Copy Protection (Dongle).....	2
1.1.1.3 System Requirements Overview.....	2
1.1.2 Establishing the TCP/IP Connection to the Target System.....	2
1.1.2.1 AR Settings.....	2
1.1.2.2 PC Settings.....	4
1.1.2.3 Testing the TCP/IP Connection.....	5
1.1.3 Initialization File – BrWebSvr.ini.....	6
1.1.3.1 Creating the "BrWebSvr.ini" Initialization File.....	6
1.1.4 Web Server Configuration File.....	7
1.1.5 Web Server Start Page.....	7
1.1.5.1 Creating the Start Page.....	7
1.1.6 Downloading Web Server.....	8
1.1.6.1 Integrating the Web Server in the Software Tree.....	8
1.1.7 Transferring Web Pages Using FTP.....	10
1.1.7.1 Publish FrontPage Web.....	10
1.1.7.2 FTP Transfer.....	10
1.1.8 Displaying Web Pages Using HTTP.....	11
1.1.9 B&R Web Server on the Internet.....	11
1.1.9.1 What is Required?.....	11
1.1.9.2 Accessing B&R Web Servers Via Domains.....	12
1.1.9.3 Saving Public IP Addresses.....	12
1.1.9.4 NAT Network Address Translation.....	12
1.1.9.5 PAT Port Address Translation und IP Masquerading.....	12
1.2 ASP Functions.....	13
1.2.1 Start Page with Process Variables.....	13
1.2.2 ReadPLC – Reading PLC Variables.....	14
1.2.2.1 Command syntax.....	14
1.2.2.2 Function.....	14
1.2.2.3 Example.....	14
1.2.3 WebPrint – Displaying Parameters.....	14
1.2.3.1 Command syntax.....	14
1.2.3.2 Function.....	14
1.2.3.3 Example.....	15
1.2.4 ReadWrite – Reading / Writing PLC Variables.....	15
1.2.4.1 Command syntax.....	15
1.2.4.2 Function.....	15
1.2.4.3 Command syntax.....	15
1.2.4.4 Function.....	15
1.2.4.5 Command syntax.....	15
1.2.4.6 Function.....	16
1.2.4.7 Command syntax.....	16
1.2.4.8 Function.....	16
1.2.4.9 Example.....	16
1.2.5 UserLogin – User Login.....	17
1.2.5.1 Command syntax.....	17
1.2.5.2 Function.....	17
1.2.5.3 Command syntax.....	17
1.2.5.4 Function.....	17
1.2.5.5 Command syntax.....	17
1.2.5.6 Function.....	17
1.2.5.7 Command syntax.....	18
1.2.5.8 Function.....	18
1.2.6 UserLogout – User Logout.....	18
1.2.6.1 Command syntax.....	18
1.2.6.2 Function.....	18
1.2.6.3 Command syntax.....	18

# Contents (9.7.2005)

## 1 B&R Web Server

1.2.6.4 Function.....	18
1.2.6.5 Command syntax.....	19
1.2.6.6 Function.....	19
1.2.7 UserGroup – User Groups.....	19
1.2.7.1 Command syntax.....	19
1.2.7.2 Function.....	19
1.2.7.3 Example.....	20
1.2.8 ErrorMsg – Error Message.....	20
1.2.8.1 Command syntax.....	20
1.2.8.2 Function.....	20
1.2.8.3 Example.....	20
1.2.9 ReadALARM – Alarm List History.....	21
1.2.9.1 Command syntax.....	21
1.2.9.2 Function.....	21
1.2.9.3 Example.....	21

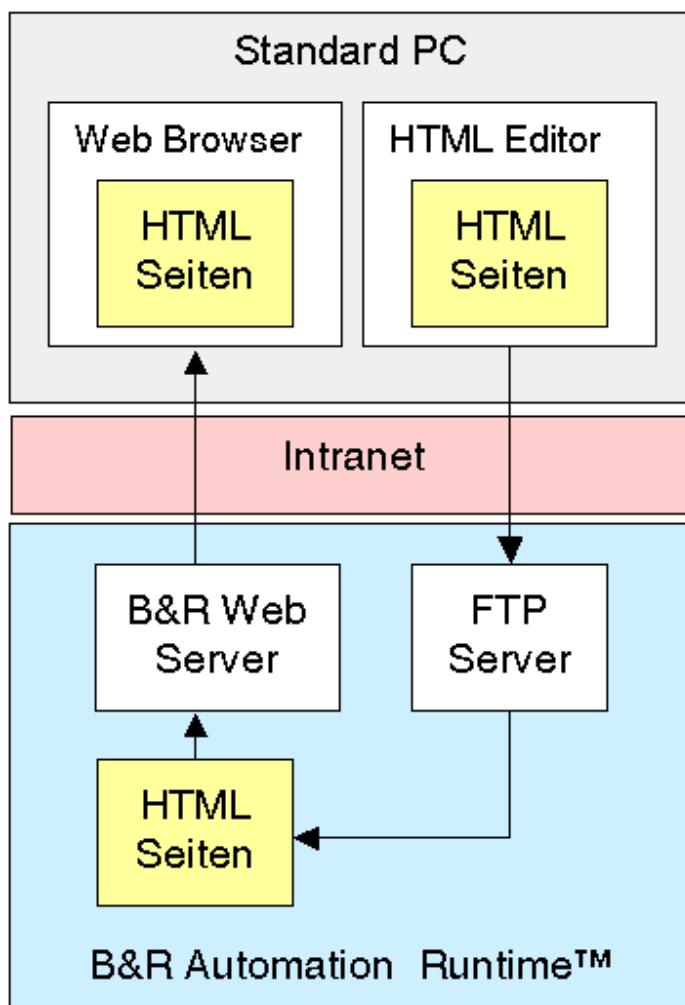
# 1 B&R Web Server

The B&R Automation Runtime platforms AR102, AR105 and CP360 are expanded to include B&R Web Server. B&R Web Server enables the display and input of process values on a standard internet browser using intranet connections.

B&R Web Server is executed as a reloadable module and transferred along with the B&R Automation Studio project to B&R Automation Runtime. In addition to B&R Web Server, an FTP server is also installed on B&R Automation Runtime.

HTML pages are created using an HTML editor (i.e. MS FrontPage) and transferred directly via FTP or the "Publish FrontPage Web..." function to B&R Automation Runtime.

A standard internet browser enables these pages to be called and displayed from the B&R Web Server. B&R Automation Runtime process variables can be output or modified using HTML pages.



## 1.1 Web Server Installation

### 1.1.1 Web Server – System Requirements

#### 1.1.1.1 Operating System Version

The B&R Web Server can be used with Automation Runtime AR102 and AR105 beginning with operating system Y1.48; on the CP360, beginning with version Y1.52.

### 1.1.1.2 Copy Protection (Dongle)

Using simple copy protection (Model No. 1A4601.02 or 1A4601.05) only allows the operation of Automation Runtime operating systems; the Web Server is not enabled by this copy protection.

Web Server operation requires the installation of a special copy protection (Model No. 1A4601.02–1 or 1A4601.05–1) which enables the use of the Automation Runtime operating system as well as B&R Web Server.

The CP360 is an exception; no copy protection (dongle) is required.

### 1.1.1.3 System Requirements Overview

	<i>Operating System</i>	<i>Dongle</i>
<b>AR102</b>	B&R Automation Runtime Y1.48	(Type AR102 dongle + Web Server (Model No. 1A4601.02–1) is necessary)
<b>AR105</b>	B&R Automation Runtime Y1.48	AR105 (Type AR105 dongle + Web Server (Model No. 1A4601.02–1) is necessary)
<b>CP360</b>	B&R Automation Runtime Y1.52	not required

## 1.1.2 Establishing the TCP/IP Connection to the Target System

### 1.1.2.1 AR Settings

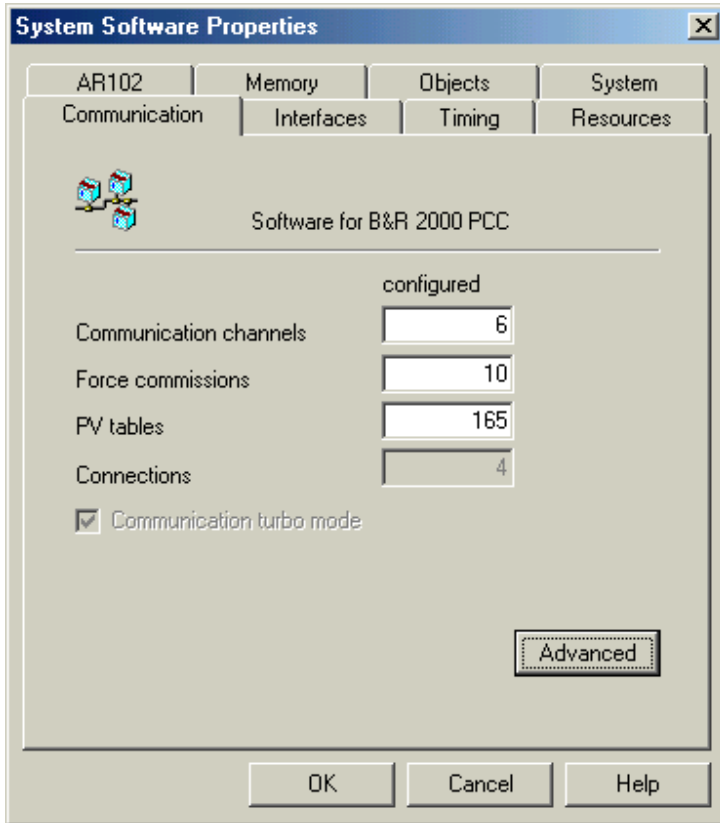
Displaying HTML pages on a web browser assumes there is an active TCP/IP connection to the B&R Automation Runtime Web Server.

For this to happen, the Ethernet TCP/IP interface on Automation Runtime needs to be configured. This takes place in Automation Studio using the TCP/IP address and subnet mask settings.

Ethernet parameters are set in the system software properties of the CPU object in Automation Studio versions up to and including V2.1. Beginning with Automation Studio V2.2, settings parameters is substantially simplified by using the configuration dialog box.

#### 1.1.2.1.1 Settings in Automation Studio V2.1

"System Software Properties" is started by selecting the CPU object and clicking on the "Edit -> Properties" menu.



When the "Advanced" button is pressed, an editor appears in which the Ethernet is activated and the IP address and net mask are input.

Attribute	Datatype	Value
StartMode	VT_UI4	1
FbDriverName	VT_BSTR	FbEthUdp
Device	VT_BSTR	
InaNode	VT_UI4	11
InaParameter	VT_BSTR	

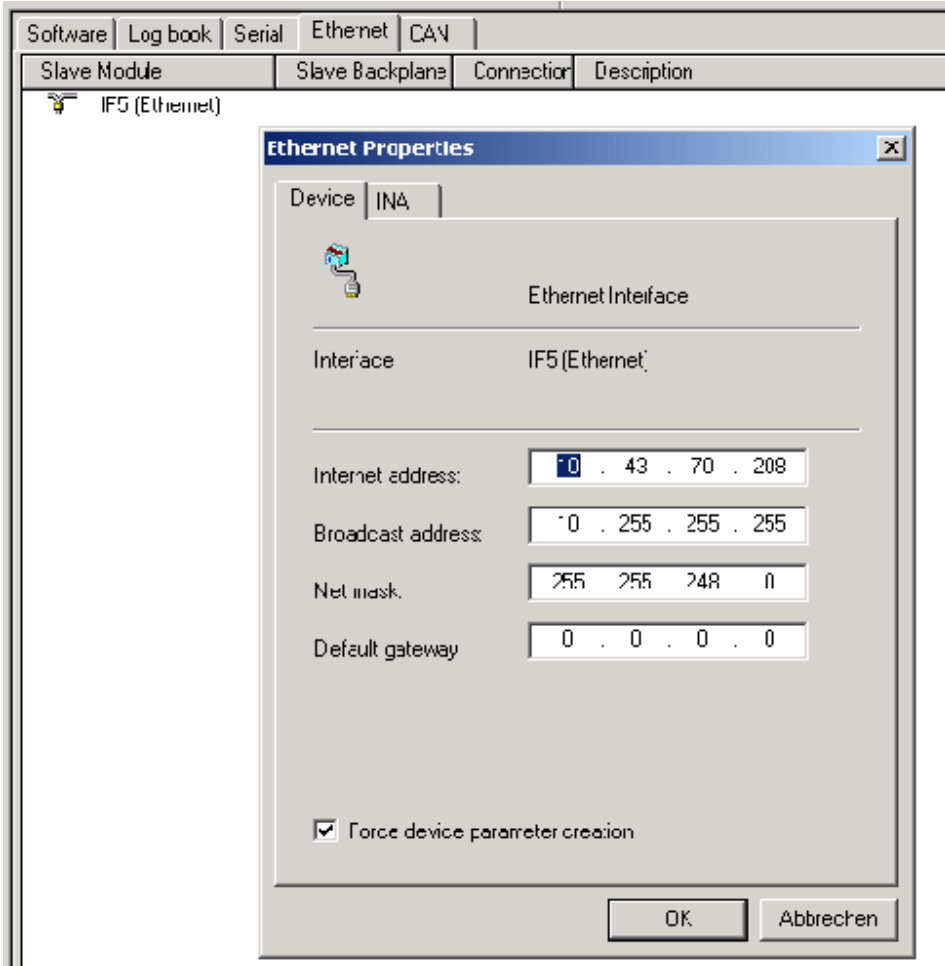
Attribute	Datatype	Value
StartMode	VT_UI4	1
InterfaceName	VT_BSTR	ene0
InternetAddress	VT_BSTR	10.43.70.208
BroadcastAddress	VT_BSTR	10.255.255.255
NetMask	VT_UI4	4294965248
DefaultGateway	VT_BSTR	

The IP address is set to 10.43.70.208 in this example; the NetMask (Subnet Mask) is set to the decimal 4294965248 which corresponds to the Subnet Mask "255.255.248.00".

After downloading the project with these settings and a system reboot, an active TCP/IP connection is made.

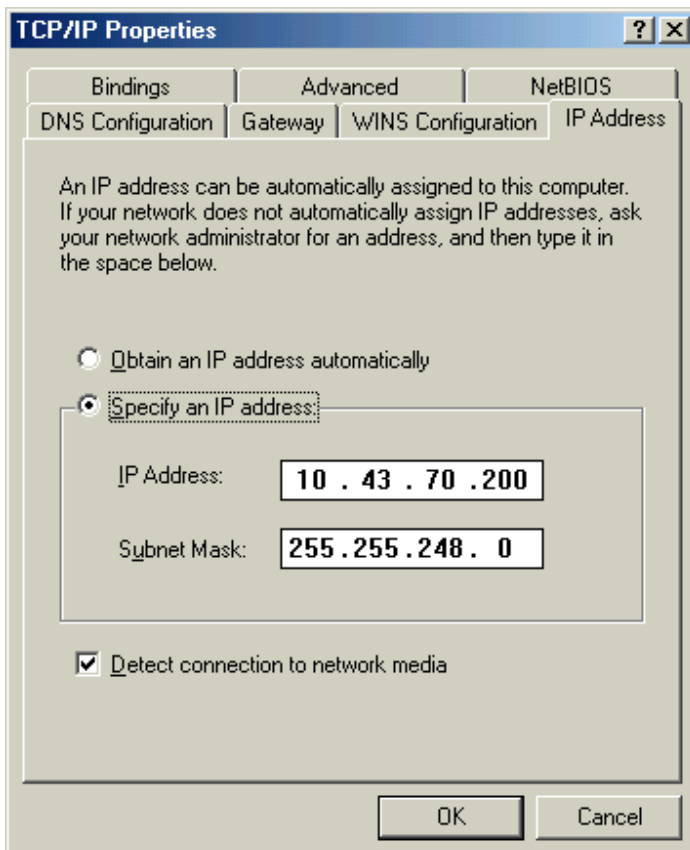
**1.1.2.1.2 Settings beginning with Automation Studio V2.2**

In the project software configuration, the "Ethernet" tab is chosen and the Ethernet interface selected, i.e. IF5. The "Ethernet Properties" dialog box for setting Ethernet parameters appears under "Edit -> Properties"



### 1.1.2.2 PC Settings

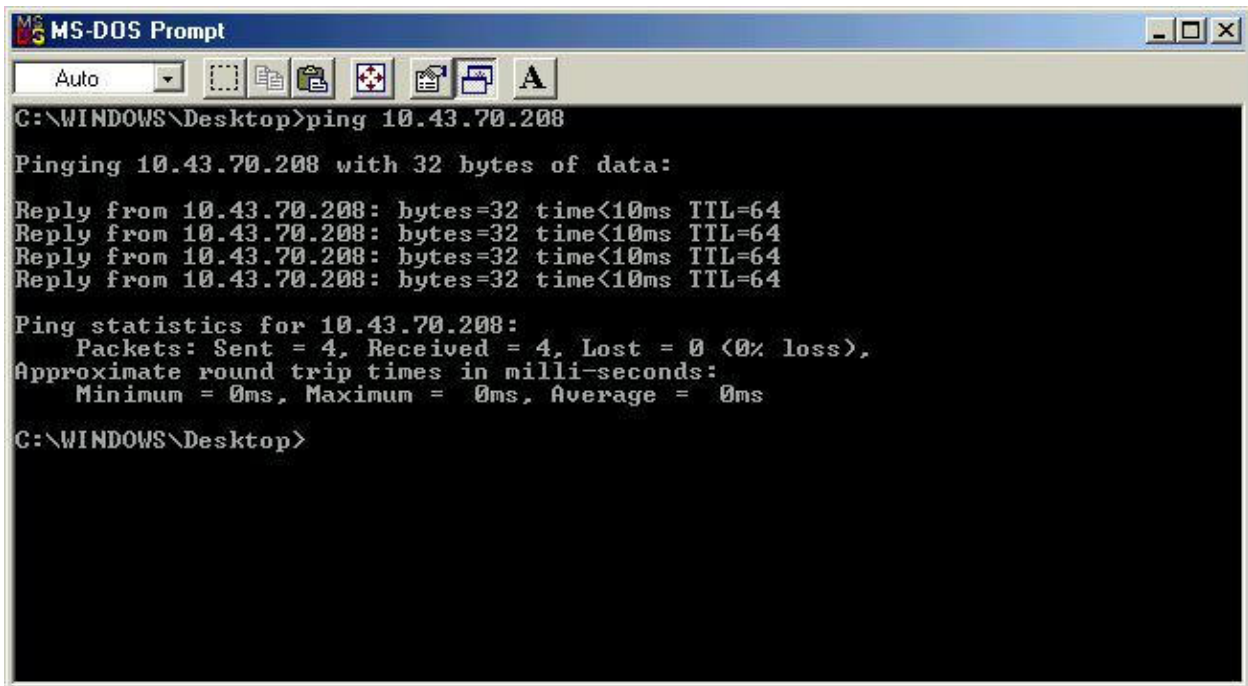
In this example, the following IP setting is used on Windows PCs which have the web browser installed.





### 1.1.2.3 Testing the TCP/IP Connection

The "ping" function can be used to test whether an active IP connection to Automation Runtime exists. On a Windows PC, this program is started in a DOS box:



```

MS-DOS Prompt
Auto
C:\WINDOWS\Desktop>ping 10.43.70.208

Pinging 10.43.70.208 with 32 bytes of data:

Reply from 10.43.70.208: bytes=32 time<10ms TTL=64
Reply from 10.43.70.208: bytes=32 time<10ms TTL=64
Reply from 10.43.70.208: bytes=32 time<10ms TTL=64
Reply from 10.43.70.208: bytes=32 time<10ms TTL=64

Ping statistics for 10.43.70.208:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

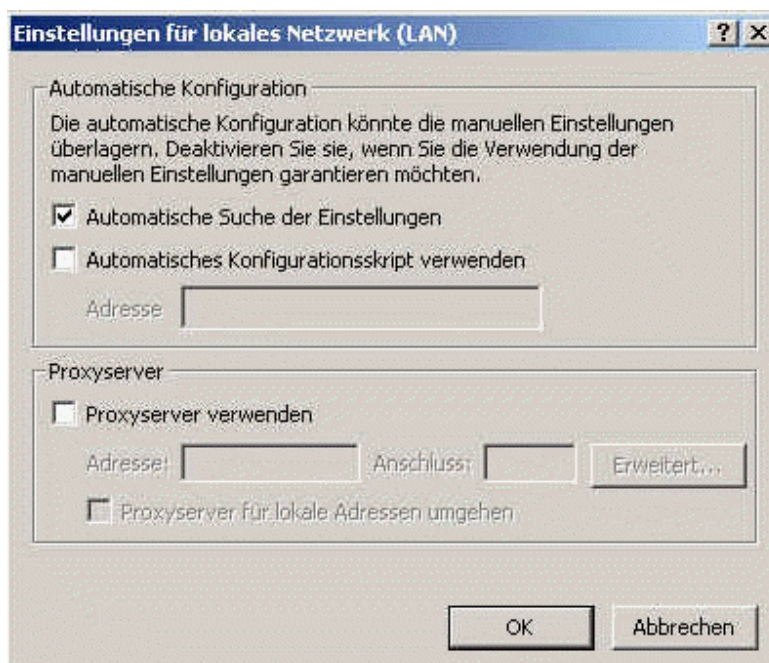
C:\WINDOWS\Desktop>

```

**Note:**

If there are problems establishing a connection between the browser and the target system, then the following browser settings should be checked and possibly changed before retrying the connection:

- ◆ Deactivate the proxy server in the proxy server settings section (LAN settings). Here is an example of settings for IE5:



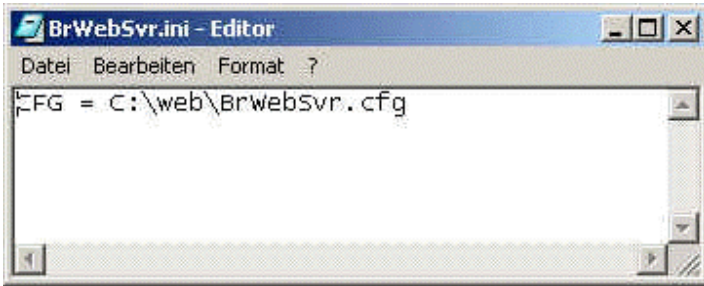
## 1.1.3 Initialization File – BrWebSvr.ini

At startup, the Web Server searches for the "C:\BrWebSvr.ini" initialization file. The name of this file (BrWebSvr.ini) and the path in which this file is stored (C:\) may not be changed by the user. In this initialization file, an entry exists with the key word "CFG = " along with the name of the path of the configuration file. Using this configuration file, the Web Server can initiate further startup procedures and then display the start page.

### 1.1.3.1 Creating the "BrWebSvr.ini" Initialization File

The "BrWebSvr.ini" file is either created in an editor and copied to the C:\ drive with FTP (File Transfer Protocol) or created during the setup generator.

#### 1.1.3.1.1 Setting Manually



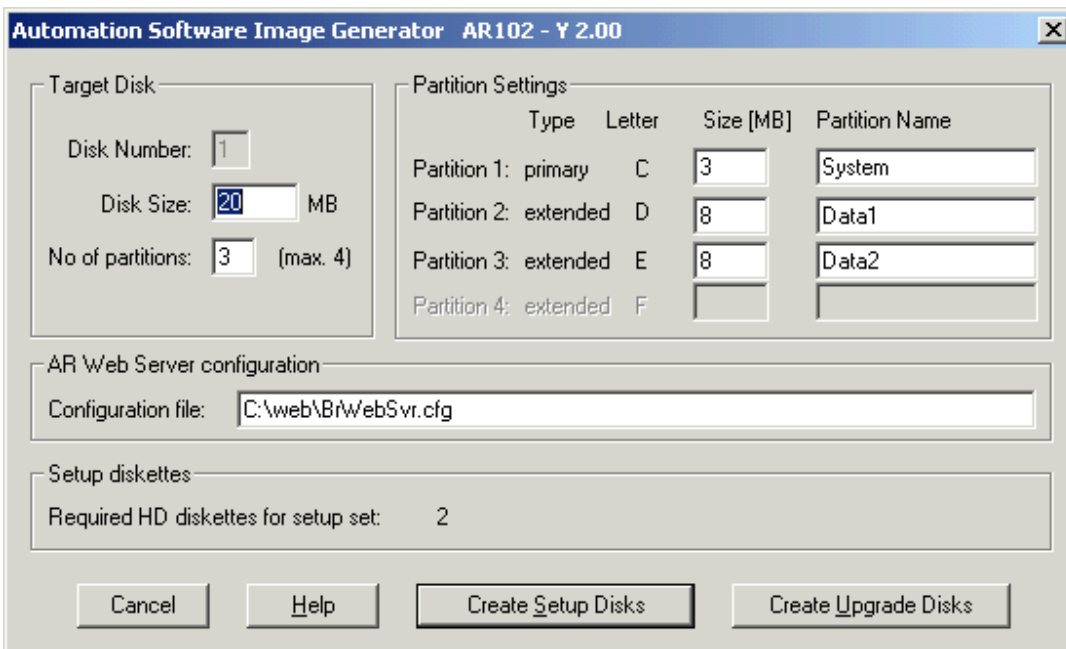
A text file with the name BrWebSvr.ini is created using a text editor (i.e. Windows editor). Only the link to the configuration file is found in this file.

#### Note:

The initialization file must be manually created and saved to the C:\ root directory in every case on the **CP360 target system!**

#### 1.1.3.1.2 Setup Generator

If installation takes place using the Automation Runtime Setup Generator (AR102 and AR105), the "BrWebSvr.ini" initialization file will be saved in the system root directory.



## 1.1.4 Web Server Configuration File

In order to operate, the Web Server requires some configuration data from the configuration file.

Creating the configuration file takes places using any desired text editor; the file must be stored in the same path as specified in the "C:\BrWebSvr.ini" file.

The configuration file must contain the following parameters:

Parameter	Description
Port	Web channel port number
Retries	Repeats
Priority	Priority
BaseDir	Main directory in which the web files are stored
WebDir	Directory containing the start page
StartPage	Name of the start page which should be opened when starting the Web Server
UserCount	Max. number of user logins

### **Example for the configuration file "BrWebSvr.cfg" :**

```
Port          = 80          ;Port
Retries       = 5           ;Retries
Priority      = 78          ;Priority

BaseDir       = C:         ;Base Directory
WebDir        = web        ;Web Directory
StartPage     = index.htm ;Start Page
```

---

```
;Userlist
```

```
UserCount    = 3          ;User Count
```

```
NAME[1]      = user1      ;User 1
PASSWORD[1]  = password
RIGHTS[1]    = 35
```

```
NAME[2]      = user2      ; User 2
PASSWORD[2]  = password
RIGHTS[2]    = 105
```

```
NAME[3]      = user3      ; User 3
PASSWORD[3]  = password
RIGHTS[3]    = 129
```

## 1.1.5 Web Server Start Page

After the Web Server is initialized, the start page entered in the configuration file is displayed.

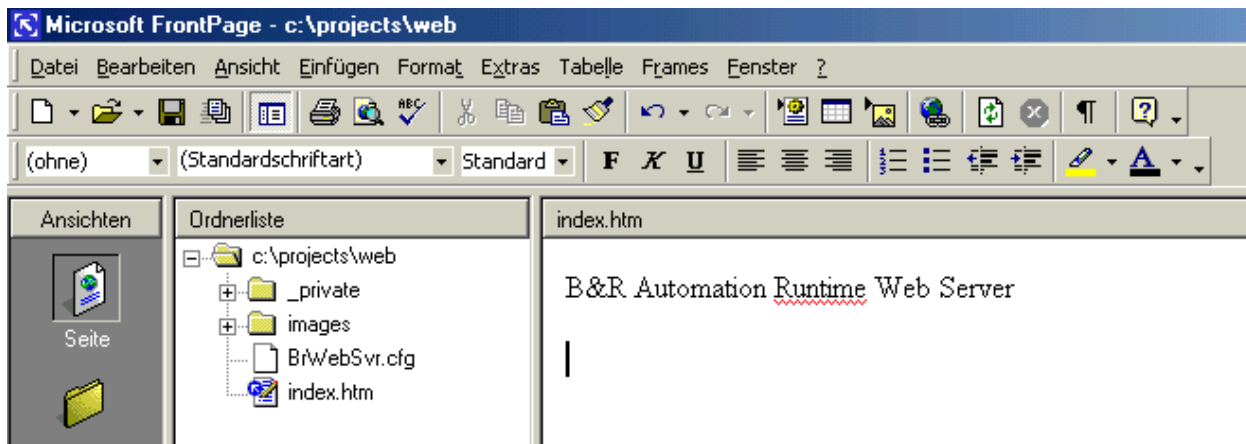
Example:

```
BaseDir       = C:         ;Base Directory
WebDir        = web        ;Web Directory
StartPage     = index.htm ;Start Page
```

### 1.1.5.1 Creating the Start Page

Creating and individually designing this page takes place using an HTML editor (i.e. FrontPage).

**Example for a simple start page:**



**HTML source code** (can be inserted into the HTML editor using the clipboard)

```
<html>

<head>
  <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
  <meta http-equiv="Content-Language" content="eng">
  <title>Homepage</title>
  <meta name="GENERATOR" content="Microsoft FrontPage 5.0">
  <meta name="ProgId" content="FrontPage.Editor.Document">
</head>

<body>
  <p>B&R Automation Runtime Web Server</p>
</body>

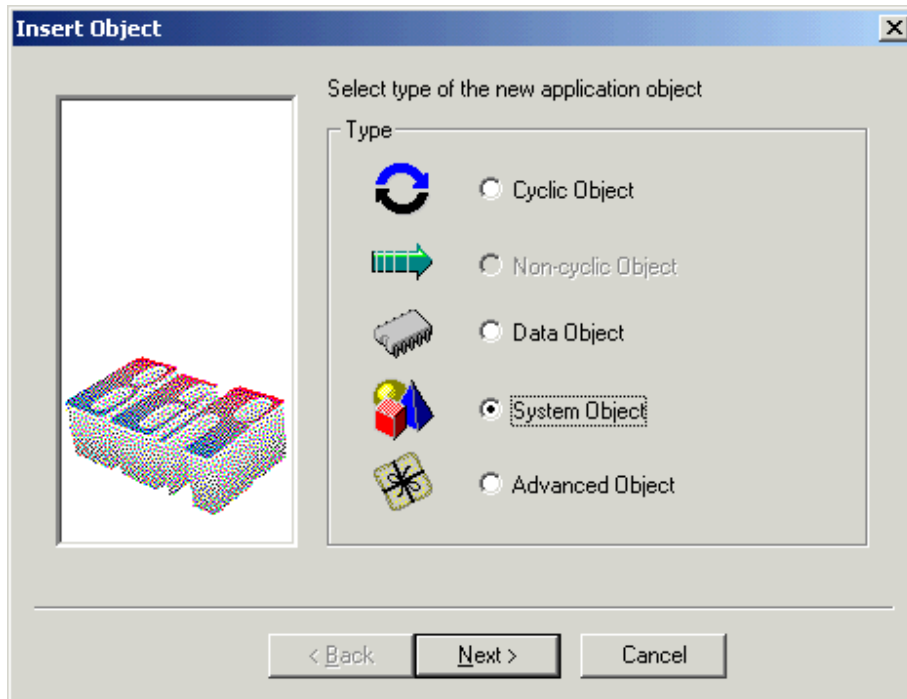
</html>
```

## 1.1.6 Downloading Web Server

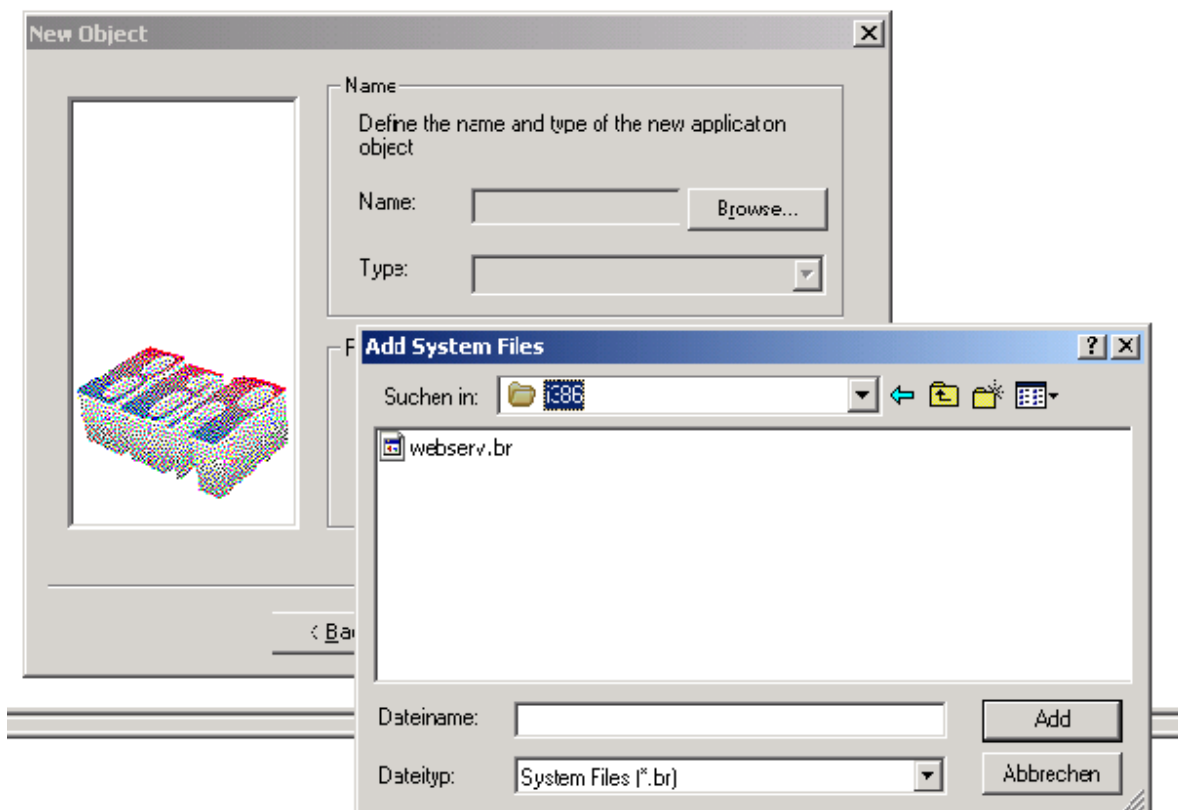
The Ethernet settings for the TCP/IP connection and the B&R Web Server module are loaded to Automation Runtime with the Automation Studio project.

### 1.1.6.1 Integrating the Web Server in the Software Tree

A new system object can be integrated into the project by selecting the CPU object in the software tree and calling up the menu "Insert -> New Object".



The "webserv.br" module can be selected by clicking "Next" and "Browse" in the "New Object" dialog box.



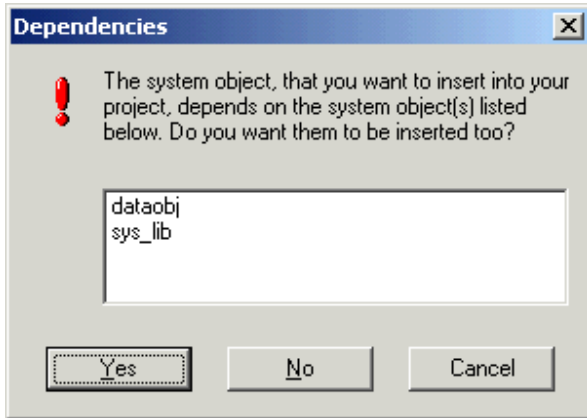
In Automation Studio version 2.1, the module is included in the following directory:

<Version Directory>/AS/WebServer/i386

Beginning with Automation Studio version 2.2, the module is included in the standard system directory:

<Version Directory>/AS/WebServer/Y0200/i386

The "webserv.br" accesses other system dependent libraries which are necessary for operation.



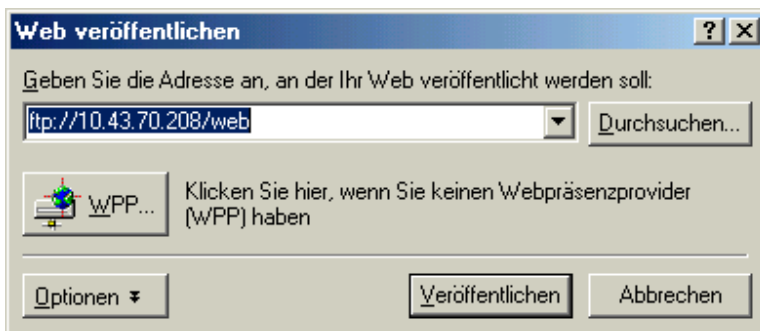
All necessary system components are now included in the project for operating the B&R Web Server.

System	Version	Use	Size
sysconf	V3.00	Use: ROM	5204
runtime	V1.09	Use: HUM	31244
webserv	V1.12	Use: ROM	168456
sys_lib	V1.37.1	Use: ROM	14756
dataobj	V1.01.2	Use: ROM	15512

## 1.1.7 Transferring Web Pages Using FTP

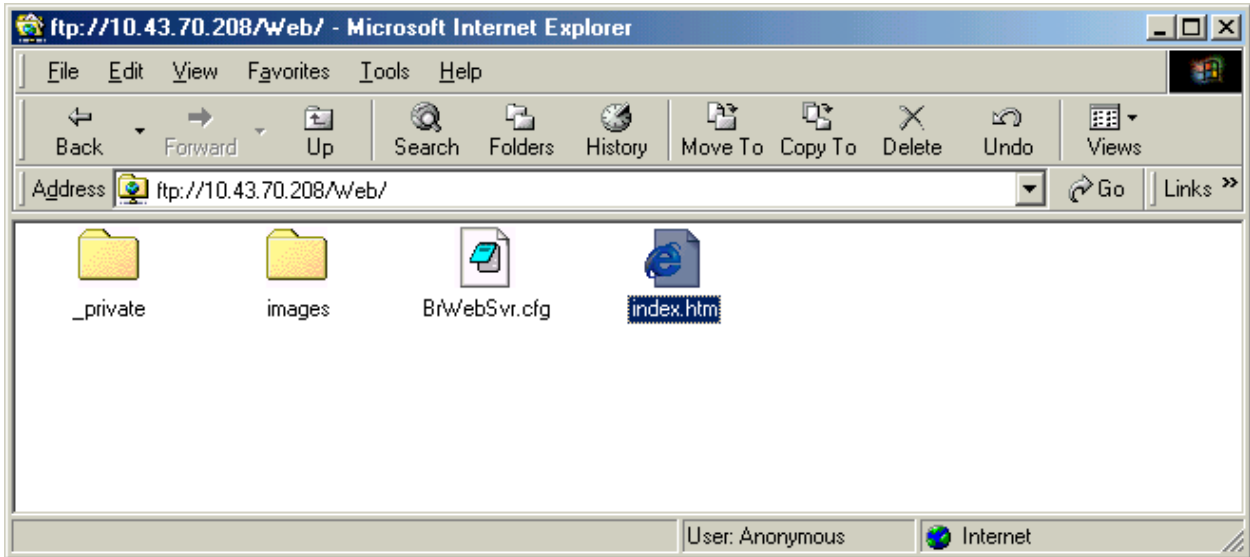
A functioning Ethernet connection to the target system is a requirement for transferring web pages to the Web Server. HTML pages can be directly transferred using FTP or by using the "Publish FrontPage Web..." function in FrontPage.

### 1.1.7.1 Publish FrontPage Web...



### 1.1.7.2 FTP Transfer

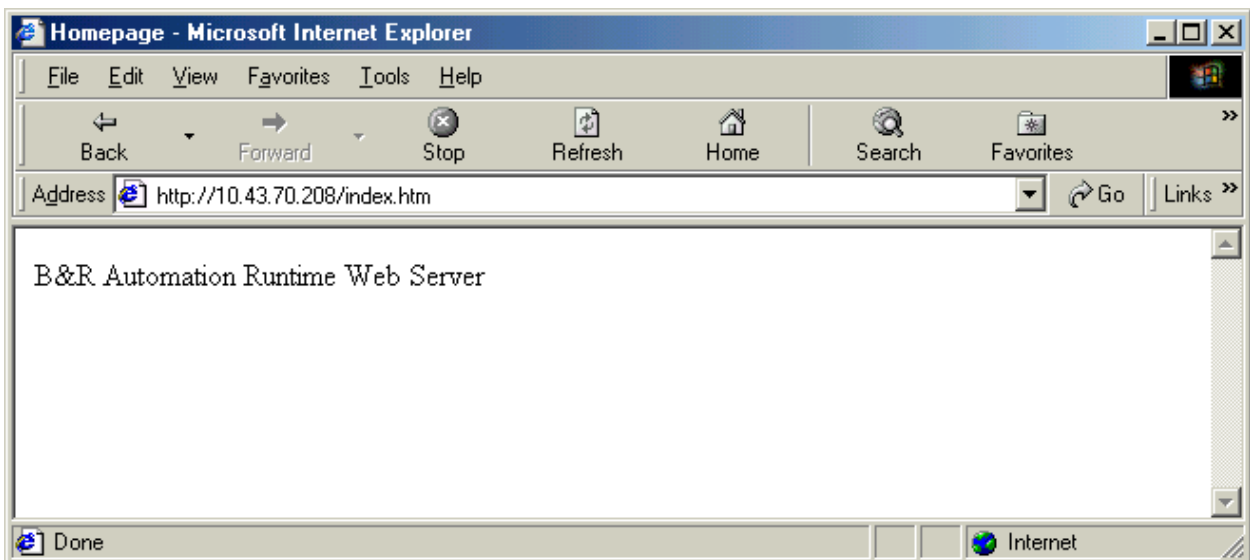
A connection to the FTP server on the target system can be made by entering the FTP address (IP address of the target system) in an FTP client (i.e. IE).



If the connection is made, the web pages can be transferred to the target system using Explorer.

### 1.1.8 Displaying Web Pages Using HTTP

Displaying HTML pages can be done by entering the HTTP address (IP address of the target system) in an HTML browser (i.e. IE).



### 1.1.9 B&R Web Server on the Internet

#### 1.1.9.1 What is Required?

##### 1.1.9.1.1 1. Permanent connection to the internet

A static, publicly registered IP address (i.e. 195.230.44.101) must be arranged so that the B&R Automation Runtime Web Server can be directly accessed from the internet.

Therefore, a constant connection to the internet is required (i.e. xDSL or dedicated line). If a dialup connection is used (i.e. ISDN), then the ISP (internet service provider) would have to dial the modem of the B&R Web Server, which providers won't usually do.

##### 1.1.9.1.2 2. Static IP address

When establishing a dedicated line, at least one IP address is obtained for the internet router to create a connection to the ISP and, therefore, the internet. Depending on the number of B&R Web Servers configured for internet usage, a corresponding number of static IP addresses need to be established for internet routing and

accessibility.

### 1.1.9.2 Accessing B&R Web Servers Via Domains

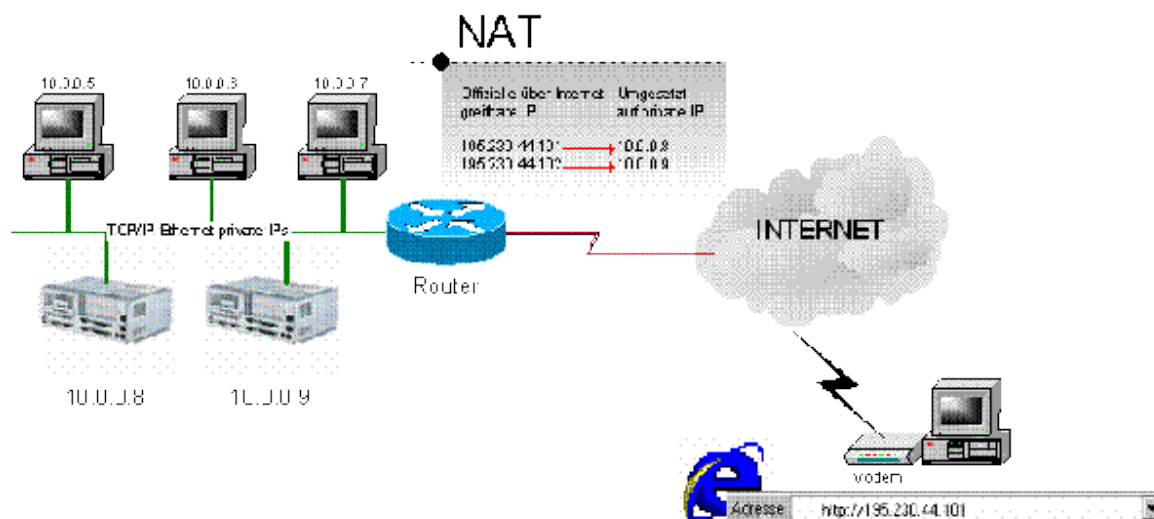
A registered domain name is required to access Web Servers using domain names (i.e. companyname.com). In these zones a name can be assigned to an IP address. If the name "plc1" is assigned to an IP address (i.e. 195.230.44.101), this must be entered in the name server zone (by the provider which is hosting the domain). In this way, the B&R Web Server can be accessed over the internet at <http://plc1.companyname.com>. The DNS server translates the name into the IP address.

### 1.1.9.3 Saving Public IP Addresses

The limited availability of static IP addresses has led to methods such as NAT, PAT and IP Masquerading which allow connections from the internet to private networks using fewer publicly registered IP addresses. All of these methods portray private addresses in accordance with RFC 1918 or a propriety (non-registered) network address area of publicly registered IP addresses.

### 1.1.9.4 NAT Network Address Translation

NAT (Network Address Translation) allows the assignment of addresses in a private network using a table of publicly registered IP addresses. One advantage of this is that computers communicating with each other within a private network do not require a public IP address. Internal computer IP addresses which need to communicate with targets on the internet obtain a table entry on the router between the internet service provider (ISP) and the private network. Using this one-to-one assignment allows computers not only to connect to targets on the internet, but also allows their access from the internet. However, the internal structure of the company network remains outwardly hidden.



A device which should be accessible over the internet can use the router's static IP address to convert this to a private IP address; if several IP addresses (or devices) are addressed in the internal network, then even more IP addresses are required on the router's "internet site". Therefore, the outer interface of the router contains several virtual IP addresses which can be reached over the internet. These IP addresses can be converted into private IP addresses in the NAT table on the router.

### 1.1.9.5 PAT Port Address Translation und IP Masquerading

If various ports are used, then the possibility exists to use the internet site router IP address for several internal IP addresses to eliminate further IP addresses. For example, the web server (http) uses network port 80. The possibility exists now on B&R Web Servers to configure various port addresses and assign them to the router IP address.

Internet IP Addresses	Port	Private IP address	Port
195.230.44.101	80	10.0.0.9	80
195.230.44.101	8080	10.0.0.8	8080
192.230.44.101	8888	10.0.0.7	8888



If PAT is used, then then the port number must be specified when accessing the web server:

Adresse

## 1.2 ASP Functions

.HTM files can only use static texts and displays.

If the B&R Web Server start page (or other pages) are to contain displays from current Automation Runtime process variables, this needs to be done using Active Server Page (.asp) files.

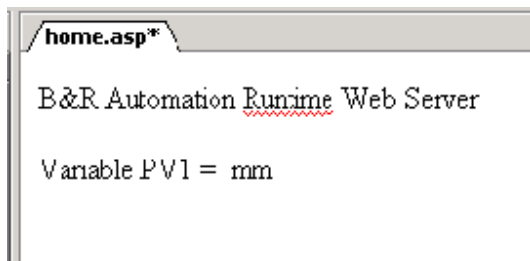
### 1.2.1 Start Page with Process Variables

The entry in the configuration file (BrWebSvr.cfg) is changed from "index.htm" to "home.asp",

```
BaseDir      = C:           ;Base Directory
WebDir       = web         ;Web Directory
StartPage    = home.asp    ;Start Page
```

and a "home.asp" file is created with the content from the value of the "PV1" process variable.

**Example of a start page with a process variable:**



**HTML source code**

```
home.asp*
<html>

<head>
<meta http-equiv="Content-Language" content="de-at">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Startseite</title>
<meta name="GENERATOR" content="Microsoft FrontPage 5.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<meta name="Microsoft Theme" content="none, default">
<meta name="Microsoft Border" content="none, default">
</head>

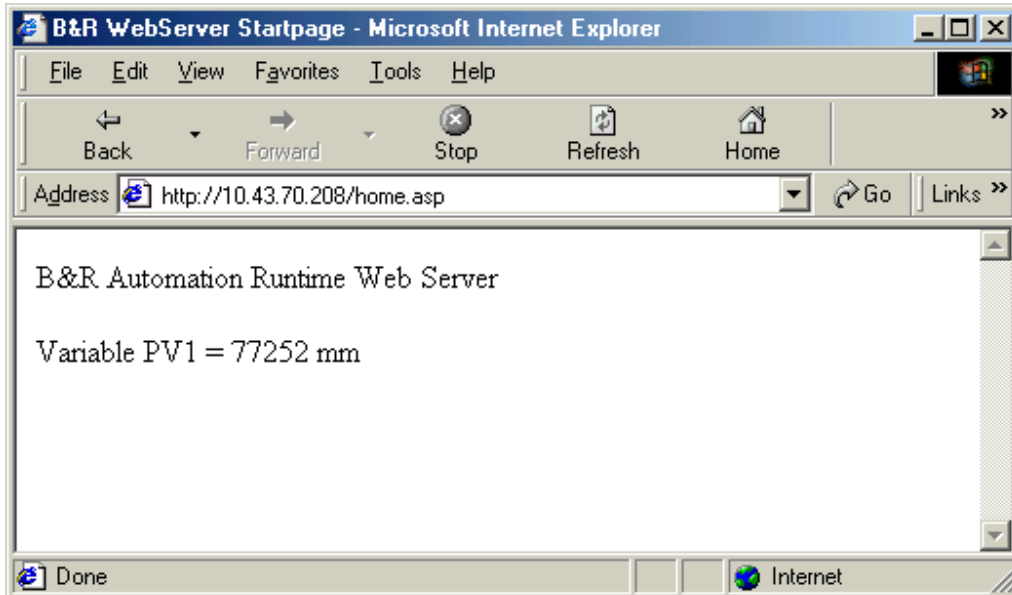
<body>

<p>B&R Automation Runtime Web Server</p>
<p>Variable PV1 = <% ReadPI.C ("a"): %> mm</p>

</body>

</html>
```

When calling this page on the HTML browser, the current value of the PV1 process variable is shown next to the static text "PV1 = ".



## 1.2.2 ReadPLC – Reading PLC Variables

### 1.2.2.1 Command syntax

```
<% ReadPLC(variable_name); %>
```

### 1.2.2.2 Function

The contents of the specified variable are inserted instead of the READPLC() ASP function. Only global variables from the PLC can be used. If the variable does not exist, then the error message "Unknown variable is displayed.

### 1.2.2.3 Example

*In the PLC program:*

```
_GLOBAL int iOp_Type;
.
..
iOp_Type = 5;
```

*In ASP file:*

```
<body>
<p>Current type of operation: <% ReadPLC ("iOp_Type"); %> </p>
</body>
```

*Result:*

Current type of operation: 5

## 1.2.3 WebPrint – Displaying Parameters

### 1.2.3.1 Command syntax

```
<% WebPrint(PARAMETER); %>
```

### 1.2.3.2 Function

The value of the specified ASP PARAMETER is inserted instead of the WebPrint() ASP function.

### 1.2.3.3 Example

Calling an ASP site: home.asp?start=30

*In ASP file:*

```
<body>
  <p>START = <% WebPrint("start"); %> </p>
</body>
```

*Result:*

START = 30

## 1.2.4 ReadWrite – Reading / Writing PLC Variables

Some parameters which allow reading and writing of PLC variables are predefined in B&R Web Server.

Parameter	Description
/goform/ReadWrite	Executes Read/Write operations
redirect	Name of the page to which the result(s) of read/write operations is/are sent.
variable	Name of the PLC variable to be written to / read from
value	Value to be read from / written to the PLC variable
write	Writing of the PLC variable
read	Reading of the PLC variable
var	Name of the read/written PLC variable
val	Value of the read/written PLC variable

Using the form technique, parameters are given to the Web Server to allow it to know which variables ("variable" and "value") should be read/written. The Web Server reads/writes ("read" or "write") the corresponding value to them and sends them to a defined page ("redirect") as parameters ("var" and "val"). Using the ASP function WebPrint(), these parameters can be evaluated and further processed.

### 1.2.4.1 Command syntax

```
<form method="POST" action="/goform/ReadWrite">
```

### 1.2.4.2 Function

With the action "/goform/ReadWrite", the Web Server is instructed by the "submit" function to execute read and write operations.

### 1.2.4.3 Command syntax

```
<input type="hidden" name="redirect" value="home.asp">
```

### 1.2.4.4 Function

The result of the read/write operation is routed to the "home.asp" page with the "var" and "val" parameters by the "submit" function (i.e. home.asp?var=a&val=25).

### 1.2.4.5 Command syntax

```
<input type="text" name="variable">
<input type="text" name="value">
```

### 1.2.4.6 Function

The contents of the "variable" and "value" input fields are sent to the Web Server by the "submit" function and transferred there from/to the PLC variables.

### 1.2.4.7 Command syntax

```
<input type="submit" value="Read PV" name="read">
<input type="submit" value="Write PV" name="write">
```

### 1.2.4.8 Function

If the "Read PV" or "Write PV" button is activated, the "submit" function is triggered which executes the corresponding read/write operation from the Web Server.

### 1.2.4.9 Example

#### Read/Write variables

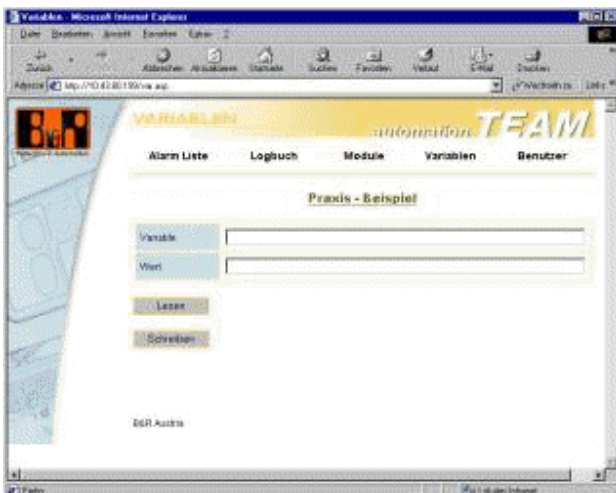
Variable  
" size="20">

Value  
" size="20">

*HTML source code (can be inserted into the HTML editor using the clipboard)*

```
<form method="POST" action="/goform/ReadWrite">
  <input type="hidden" name="redirect" size=50 value="home.asp">
  <b>Read / Write variables</b>
  <p>Variable<br>
  <input type="text" name="variable" value="<% WebPrint("var"); %>"></p>
  <p>Value<br>
  <input type="text" name="value" value="<% WebPrint("val"); %>"></p>
  <p><input type="submit" value="Reading" name="read"></p>
  <p><input type="submit" value="Writing" name="write"></p>
</form>
```

*Application example: (Reading / Writing Variables)*



## 1.2.5 UserLogin – User Login

B&R Web Server provides an authorization system to protect certain websites or parts of websites from unauthorized access.

Parameter	Description
/goform/UserLogin	Use of the Login function
username	User name (see configuration file)
password	Assigned password (see configuration file)
redirect	Following page (optional)
error_redirect	Following page in case of error (optional)

A certain user logs into B&R Web Server ("user" and "password"). User names, passwords, and rights can be assigned in the "BrWebSvr.cfg" configuration file.

Only one user per IP address can be logged in at the same time. Pages or parts of a page can be closed off to certain user groups by evaluating rights with the UserGroup() function. It is possible to create a notification page (error message) if a user is denied access to a certain page. The login remains until the user logs out or the Web Server is restarted. If the login is successful the "following page" will be shown; if not successful, the "following page in case of error" will be displayed instead. If no following page is defined, a default reply will be output.

### 1.2.5.1 Command syntax

```
<form method="POST" action="/goform/UserLogin">
```

### 1.2.5.2 Function

With the action "/goform/UserLogin", the Web Server is instructed by the "submit" function to execute the login function.

### 1.2.5.3 Command syntax

```
<input type="hidden" name="redirect" value="user1.asp">
<input type="hidden" name="error_redirect" value="user1e.asp">
```

### 1.2.5.4 Function

After the login operation and the "submit" function, the "user1.asp" page will be called if the the user is authorized; "user1e.asp" will be called if the login is refused. The "user" and "password" parameters will be transferred. (i.e. user1.asp?user=user2&password=\*\*\*\*\*).

### 1.2.5.5 Command syntax

```
<input type="text" name="username">
<input type="password" name="password">
```

### 1.2.5.6 Function

The "username" and "password" input fields will be sent to the Web Server after the "submit" function. Authorization will be checked corresponding to the entries in the configuration file.

### 1.2.5.7 Command syntax

```
<input type="submit" value="User Login">
```

### 1.2.5.8 Function

If the "Login" button is pressed, the "submit" function is triggered and the corresponding login operation is begun by the Web Server.

#### ***Login form example:***

##### **User Login:**

Name

Password

*HTML source code (can be inserted into the HTML editor using the clipboard)*

```
<form method="POST" action="/goform/UserLogin">
  <input type="hidden" name="redirect" value="user1.asp">
  <input type="hidden" name="error_redirect" value="error1.asp">
  <p><b>User-Login:</b></p>
  <p>Name<br>
  <input type="text" name="username"></p>
  <p>Password<br>
  <input type="password" name="password"></p>
  <p><input type="submit" value="login" ></p>
  <p><input type="reset" value="reset"></p>
</form>
```

## 1.2.6 UserLogout – User Logout

Parameter	Description
redirect	Following page (optional)

A logged in user is logged out. The logout must take place from the same computer (IP address) as the login. If the user is logged in many times (on several IP addresses), only the current computer will be logged out. A global logout is not possible. The "following page" will be displayed after successfully logging out.

### 1.2.6.1 Command syntax

```
<form method="POST" action="/goform/UserLogout">
```

### 1.2.6.2 Function

With the action "/goform/UserLogout", the Web Server is instructed by the "submit" function to execute the logout function.

### 1.2.6.3 Command syntax

```
<input type="hidden" name="redirect" value="user1o.asp">
```

### 1.2.6.4 Function

The "user1o.asp" page will be called as a result of the logout operation and the submit function.

### 1.2.6.5 Command syntax

```
<input type="submit" value="User Logout">
```

### 1.2.6.6 Function

If the "Logout" button is pressed, the "submit" function is triggered and the corresponding logout operation is begun by the Web Server.

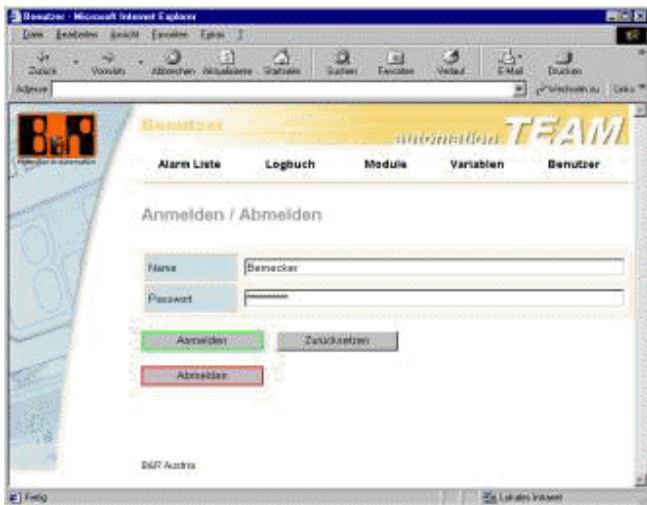
#### Logout form example:

##### User Logout:

HTML source code (can be inserted into the HTML editor using the clipboard)

```
<form method="POST" action="/goform/UserLogout">
<input type="hidden" name="redirect" value="userlo.asp">
<p><b>User-Logout:</b></p>
<p><input type="submit" value="Logout" ></p>
</form>
```

User management example: (combination of login and logout forms)



## 1.2.7 UserGroup – User Groups

### 1.2.7.1 Command syntax

```
<% UserGroup(rights); %>
```

### 1.2.7.2 Function

The HTML code which follows the function only appears on the client browser if the logged-on user has the necessary rights. Checking authorization takes place by comparing user rights bits (BrWebSvr.cfg configuration file) and the transferred "rights" parameter. By repeating the call to this ASP function, the rights are rechecked; the current authorization determines whether the following HTML code is displayed or not. To reactivate the display for all users, the function must be called with the "rights" parameter set to 0 (rights = 0).

### 1.2.7.3 Example

All users can see this text.

All users in Group 2 can see this text.  
Allowed authorization: all users for which bit 2 (dec. 2) is set.

All users in Group 8 can see this text.  
Allowed authorization: all users for which bit 3 (dec. 8) is set.

All users in Group 20 can see this text.  
Allowed authorization: all users for which bits 3 and 5 (dec. 20) are set.

All users can see this text (again).

*HTML source code (can be inserted into the HTML editor using the clipboard)*

```
<p>
All users can see this text.</p>
<% UserGroup(2); %>
<p>All users in Group 2 can see this text.<br>Allowed authorization: all users for which bit 2 (dec. 2) is set.
<% UserGroup(8); %>
<p>All users in Group 8 can see this text.<br>
Allowed authorization: all users for which bit 3 (dec. 8) is set. </p>
<% UserGroup(20); %>
<p>All users in Group 20 can see this text.<br>
Allowed authorization: all users for which bits 3 and 5 (dec. 20) are set. </p>
<% UserGroup(0); %>
<p>All users can see this text (again).</p>
```

## 1.2.8 ErrorMsg – Error Message

### 1.2.8.1 Command syntax

```
<% ErrorMsg(); %>
```

### 1.2.8.2 Function

The ASP function ErrMsg() is used in connection with the UserGroup() ASP function to display error messages regarding insufficient user rights. The authorizations of the last UserGroup function are inverted.

### 1.2.8.3 Example

All users can see this text.

All users in Group 2 can see this text.  
Allowed authorization: all users for which bit 2 (dec. 2) is set.

**You are not a user in Group 2.**

All users in Group 8 can see this text.  
Allowed authorization: all users for which bit 3 (dec. 8) is set.

**You are not a user in Group 8.**

All users in Group 20 can see this text.  
Allowed authorization: all users for which bits 3 and 5 (dec. 20) are set.

**You are not a user in Group 20.**

All users can see this text (again).



HTML source code (can be inserted into the HTML editor using the clipboard)

```
<p>All users can see this text.</p>
<% UserGroup(2); %>
<p>All users in Group 2 can see this text.<br>Allowed authorization: all users for which bit 2 (dec. 2) is s
<% ErrorMsg(); %> <p><font color="#FF0000">You are not a user in Group 2.</font></p>
<% UserGroup(8); %>
<p>All users in Group 8 can see this text.<br>Allowed authorization: all users for which bit 3 (dec. 8) is s
<% ErrorMsg(); %> <p><font color="#FF0000">You are not a user in Group 8.</font></p>
<% UserGroup(20); %>
<p>All users in Group 20 can see this text.<br>Allowed authorization: all users for which bits 3 and 5 (dec.
<% ErrorMsg(); %> <p><font color="#FF0000">You are not a user in Group 20.</font></p>
<% UserGroup(0); %>
<p>All users can see this text (again)</p>
<p> </p>
```

## 1.2.9 ReadALARM – Alarm List History

If B&R Web Server is running alongside a Visual Components application, the alarm history can be displayed using the Web Server.

This function is only possible in connection with VA\_GetAlarmList or VA\_GetExAlarmList (VISAPI)!

### 1.2.9.1 Command syntax

```
<% ReadALARM("Errorstring",index); %>
```

### 1.2.9.2 Function

The respective replacement text ("index") is inserted in place of the ReadALARM() ASP function. The error string is the result of the VA\_GetAlarmList (or VA\_GetExAlarmList) VISAPI functions. The semicolon (;) must be used as a separating character.

Parameter	Description
Errorstring	An error string separated by a ";" (VA_GetExAlarmList!!)
index	1 = date and time 2 = group number 3 = alarm number 4 = alarm text 5 = alarm status

Text appearance is also dependent on the settings in the Visual Components alarm system.

Observe the VISAPI function documentation!

### 1.2.9.3 Example

ASP file:

```
<body>
  <% ReadALARM("Errorstr",3); %> -> <% ReadALARM("Errorstr",4); %>
</body>
```

Result:

14 -> Tank 2 full

