SIEMENS

SIMATIC

S7-1500
Web server

Function Manual

01/2013

A5E03484625-01

Legal information

Warning notice system

This manual contains notices you have to observe in order to ensure your personal safety, as well as to prevent damage to property. The notices referring to your personal safety are highlighted in the manual by a safety alert symbol, notices referring only to property damage have no safety alert symbol. These notices shown below are graded according to the degree of danger.

⚠DANGER
indicates that death or severe personal injury **will** result if proper precautions are not taken.

⚠WARNING
indicates that death or severe personal injury **may** result if proper precautions are not taken.

⚠CAUTION
indicates that minor personal injury can result if proper precautions are not taken.

NOTICE
indicates that property damage can result if proper precautions are not taken.

If more than one degree of danger is present, the warning notice representing the highest degree of danger will be used. A notice warning of injury to persons with a safety alert symbol may also include a warning relating to property damage.

Qualified Personnel

The product/system described in this documentation may be operated only by **personnel qualified** for the specific task in accordance with the relevant documentation, in particular its warning notices and safety instructions. Qualified personnel are those who, based on their training and experience, are capable of identifying risks and avoiding potential hazards when working with these products/systems.

Proper use of Siemens products

Note the following:

⚠WARNING
Siemens products may only be used for the applications described in the catalog and in the relevant technical documentation. If products and components from other manufacturers are used, these must be recommended or approved by Siemens. Proper transport, storage, installation, assembly, commissioning, operation and maintenance are required to ensure that the products operate safely and without any problems. The permissible ambient conditions must be complied with. The information in the relevant documentation must be observed.

Trademarks

All names identified by ® are registered trademarks of Siemens AG. The remaining trademarks in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owner.

Disclaimer of Liability

We have reviewed the contents of this publication to ensure consistency with the hardware and software described. Since variance cannot be precluded entirely, we cannot guarantee full consistency. However, the information in this publication is reviewed regularly and any necessary corrections are included in subsequent editions.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
GERMANY

A5E03484625-01
Ⓟ 01/2013 Technical data subject to change

Preface

Purpose of the documentation

This documentation supports you in the operation of the Web server.

The Web server of the S7-1500 CPU offers, among other things, website access to diagnostic data and to process data of the CPU.

Basic knowledge required

The following knowledge is required in order to understand the documentation:

- General knowledge in the field of automation technology
- Knowledge of the industrial automation system SIMATIC
- Experience of working with Windows-based computers
- Experience of working with STEP 7

Conventions

- This documentation contains illustrations of the Web server user interface. The user interface can deviate in details from the illustrations depending on the browser you use.
- The term "User-defined websites" used in the STEP 7 user interface refers to the "User pages" described in this documentation.

Please also observe notes marked as follows:

Note

A note contains important information on the product described in the documentation, on the handling of the product and on the section of the documentation to which particular attention should be paid.

Additional support

- The range of technical documentation for the individual SIMATIC products and SIMATIC systems is available on the Internet (http://www.siemens.com/simatic-tech-doku-portal).
- The online catalog and the ordering system are available on the Internet (http://mall.automation.siemens.com).

Table of contents

Documentation guide 1

Introduction

This modular documentation of the SIMATIC products covers diverse topics concerning your automation system.

The complete documentation for the S7-1500 system consists of a system manuals, function manuals and manuals.

The STEP 7 information system (Online Help) also helps you to configure and program your automation system.

Overview of additional documentation on the topic Web server

The table below includes additional documentation which supplements this description on the topic of Web servers.

Table 1- 1 Web Server documentation

Topic	Documentation	Most important contents
Overview information PROFINET	PROFINET (http://www.automation.siemens.com/mcms/infocenter/content/en/Pages/order_form.aspx?InfoType=brochures&nodeKey=key_8097497) brochure	• Overview • Application examples • Hardware • Software
STEP 7 (TIA Portal)	STEP 7 Professional V12 online help	Configuring and programming with the engineering software
CPU	CPU 1511-1 PN (http://support.automation.siemens.com/WW/view/en/59402190)	• Block diagram • Wiring • Parameter assignment/addressing • Interrupts/diagnostic alarms • Technical specifications • Dimensional drawings
	CPU 1513-1 PN (http://support.automation.siemens.com/WW/view/en/59186494)	
	CPU 1516-3 PN/DP (http://support.automation.siemens.com/WW/view/en/59191914)	

SIMATIC manuals

All current manuals for SIMATIC products are available for download free of charge on the Internet (http://support.automation.siemens.com).

General information 2

2.1 Properties of the Web server

Benefits of the Web server

The Web server gives you the option to monitor your CPU across a network. This permits evaluation and diagnostics over long distances. Monitoring and evaluation is possible without STEP 7; all you need is a web browser. Make sure that you protect the CPU from being compromised through the use of different methods (e.g., limiting network access, using firewalls).

Activating the Web server

The Web server is deactivated in the delivery state of the CPU. This means that you must load a project in which the Web server is activated to enable access using the web browser.

Safety functions

The Web server provides the following security functions:

- Access via secure transmission protocol "https"
- User authorizations you can configure by means of user list
- Enabling for specific interfaces

Web browser

You need a web browser to access the HTML pages of the CPU.

The following web browsers have been tested explicitly for communication with the CPU:

- Internet Explorer (Version 8)
- Mozilla Firefox (Version 5.0)
- mobileSafari (iOS5)

Reading out information

You can read out the following information from the CPU with the Web server:

- Start page with general CPU information (Page 19)
- Identification information (Page 22)
- Content of the diagnostic buffer (Page 23)
- Module information (Page 24)
- Alarms (Page 30) (without acknowledgment option)
- Information about Communication (Page 32)
  - Important interface parameters
  - Port statistics
  - Display of the communication resources
- PROFINET- Topology (Page 34)
  - Display of the actual topology
  - Display of the target topology specified in configuration data
- User pages (Page 44)
- File browser (Page 67)

On the following pages you will find detailed information on the HTML pages.

Web access to the CPU via PG/PC, HMI devices and mobile terminal devices

Proceed as follows to access the Web server:

1. Use STEP 7 to download a project to S7-1500 in which the Web server is enabled.
2. Enable WLAN on the display device and establish a connection to the access point (e.g., SCALANCE W788-1RR or SCALANCE W784-1, etc.).
   If you are not working with WLAN, connect the display device (PG/PC, HMI, mobile terminal device, such as tablet PC or smartphone) with the CPU or a communication module using a PROFINET interface.
3. Open the web browser on the display device.
4. Enter the IP address of the CPU in the "Address" field of the web browser in the format: http://a.b.c.d or https://a.b.c.d (exemplary input: http://192.168.3.141).
   The CPU intro page opens. From the intro page you can navigate to additional information. Additional information on access using the secure transmission protocol "https" is available in the chapter Configuring the Web server (Page 11) in the section "Permit access only with HTTPS only".

For HMI devices operating with the preinstalled Windows CE operating system (V 5.x or earlier), the CPU information is processed in a browser especially developed for Windows CE. The information appears in a simplified format in this browser.

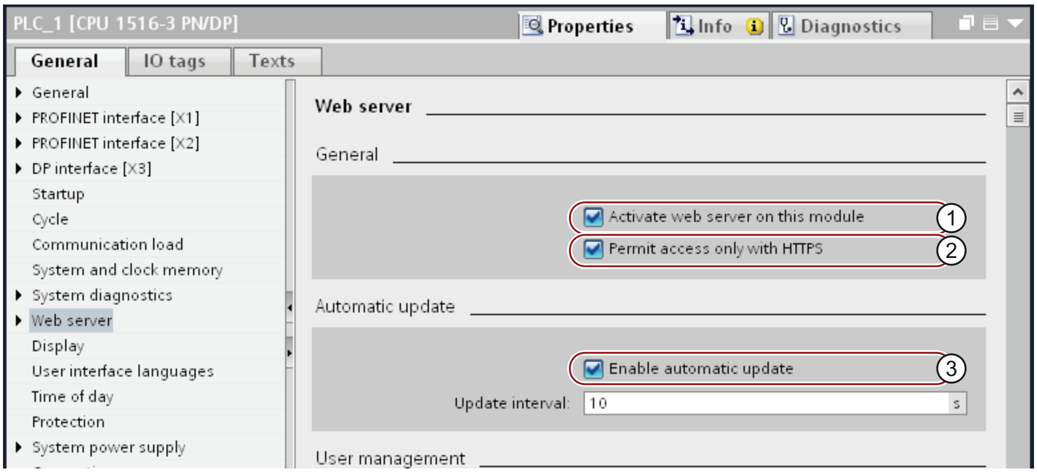
The Web server offers websites with reduced complexity that can be used on devices with smaller screens and less computing power.

2.2 Configuring the Web server

Requirements

You have opened the properties dialog of the CPU in the STEP 7 project view.

Procedure



Web server

General

☑ Activate web server on this module

☑ Permit access only with HTTPS

Automatic update

☑ Enable automatic update

Update interval: 10 s

User management

Figure 2-1 Web server settings in STEP 7

To use the full range of functions offered by the Web server, you need to make the following listed settings in STEP 7:

- ① Activating the Web server

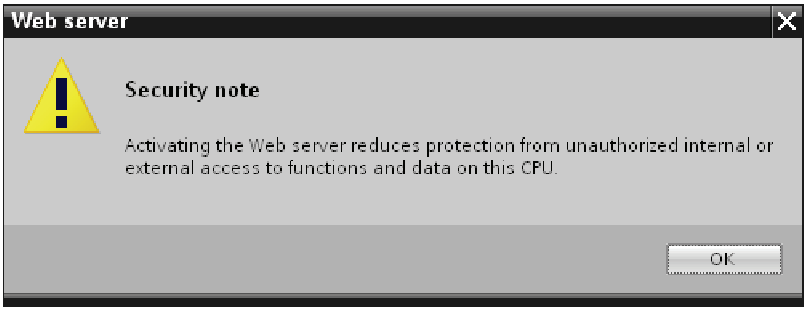
  The Web server is disabled in the basic setting of a configured CPU; to enable the Web server, follow these steps:

  – Open the "Devices & Networks" view with a double-click in the STEP 7 project tree.

  – Select the required CPU in the device view, network view or topology view.

  – Navigate to the "Web server" area in the Inspector window properties, "General" tab.

  – Select the "Enable Web server on this module" check box.
  The following notice is output:



Security note

Activating the Web server reduces protection from unauthorized internal or external access to functions and data on this CPU.

Figure 2-2 Security notice upon activation of the Web server in STEP 7

Note

When projects from deliveries are applied in which the Web server was already activated and configured on the module, this security notice is not shown.

- ② Permit access only with HTTPS

  The websites are transmitted by default via insecure connection and are not protected from attacks by third parties. If you always want to transmit websites and login information to the browser in encrypted form, enable the "Permit access only with HTTPS" option in the CPU properties. Note that the URL of the CPU starts with "https://" in this case.

  For error-free https access to the CPU, the following conditions must be met:

  – The current time is set on the CPU

  – The IP address of the CPU is set (example: https://192.168.3.141)

  – A valid certificate is installed in your web browser.

    If no certificate is installed, a warning is output recommending that you do not use the page. To view this page, you must explicitly "Add an exception".

    A valid certificate (Certification Authority) is available for download from the "Intro" website, "Download certificate". Instructions for installing the certificate are available in the help system of your Web browser.

  Note

  To protect against manipulation from the outside, download the certificate only in an environment that is guaranteed not to be compromised. You have to download the certificate once for each display device you are going to use.

- Access protection
  The certificate establishes an encrypted connection that prevents wiretapping or distortion of the communication but does not provide access protection. This means you have to protect your CPU from unauthorized access with the corresponding configuration in the user administration.

- ③ Automatic updating

  Automatic updating is activated in the basic setting of a configured CPU.

  The following websites are updated automatically:

  – Start page

  – Diagnostic buffer

  – Module information

  – Alarms

  – Information about communication

  – Topology

  Note

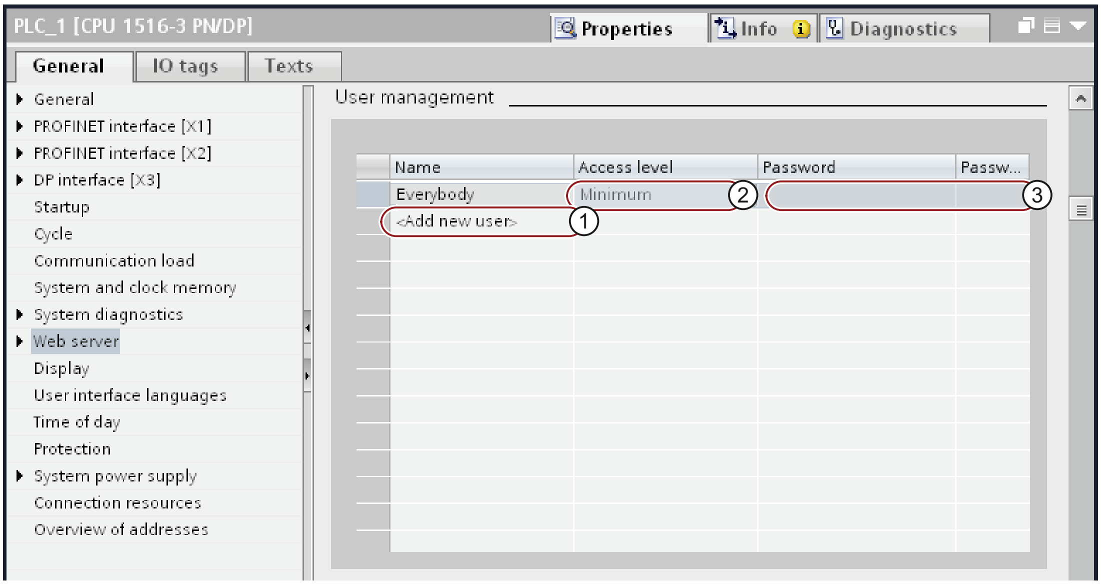
  The preset update interval is 10 seconds.
  Larger amounts of data or several http-/https connections increase the update time.

- **Setting the language for the web**

  Activate the project languages you want to use and then select up to two languages for the web. Assign one of the activated project languages to each of the languages for the web.

  You can find additional information in chapter Language settings (Page 16).

- **Amending user administration**



Name	Access level	Password	Passw...
Everybody	Minimum		
<Add new user>			

Figure 2-3 User administration in STEP 7

In STEP 7, you manage the user list in the area "Web server > User administration".

The user list provides the following options:

– ① Create user

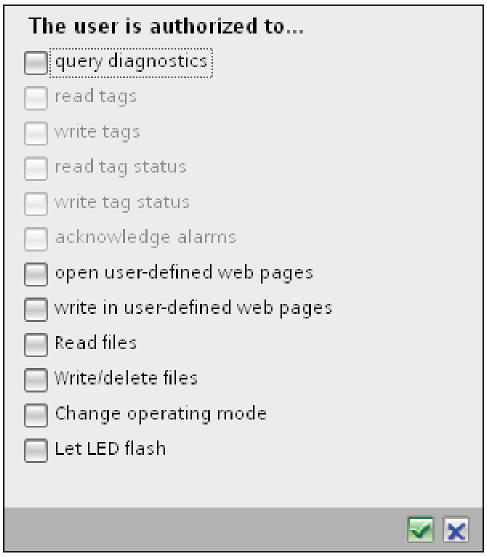
– ② Specify access permissions

– ③ Assign passwords

Users only have access to the options that are permanently linked to the access permissions.

You can assign different user rights depending on the used CPU and firmware. Rights that your CPU does not support cannot be activated.

The available rights can be available for selection as follows in STEP 7:



The user is authorized to...

□ query diagnostics
□ read tags
□ write tags
□ read tag status
□ write tag status
□ acknowledge alarms
□ open user-defined web pages
□ write in user-defined web pages
□ Read files
□ Write/delete files
□ Change operating mode
□ Let LED flash

Figure 2-4 Assignment of user rights in STEP 7

If you have not configured an additional user, access as user "Everybody" is set by default.

If users have been configured, any user that is not logged on accesses the Web server as user "Everybody".

User "Everybody"

A user with the name "Everybody" is preset in the user list; this user has minimum access rights. These are read-only access to the intro page and start page. The user "Everybody" is defined without a password. However, you can assign all access authorizations available in STEP 7 to this user.

You can create a maximum of 20 users and a user "Everybody".

Because the user "Everybody" is defined in STEP 7 without assignment of a password, be careful of the access authorizations which you assign to this user.
Certain authorizations, such as the possibility of changing the operating state, could represent a security risk.
For the assignment of security-relevant authorizations, we recommend that you configure a user and always assign a password in STEP 7.

Assign secure passwords to users during configuration. An example of a secure password is one which is only used for a single application, is more than 8 characters long, and consists of lower- and upper-case letters as well as symbols and numbers (?!+%$1234...). In addition, passwords based on common keyboard sequences or words from the dictionary should be avoided.
Change the password at regular intervals.

Whenever possible, select the option "Permit access only with HTTPS" as soon as you have assigned a password to at least one user.

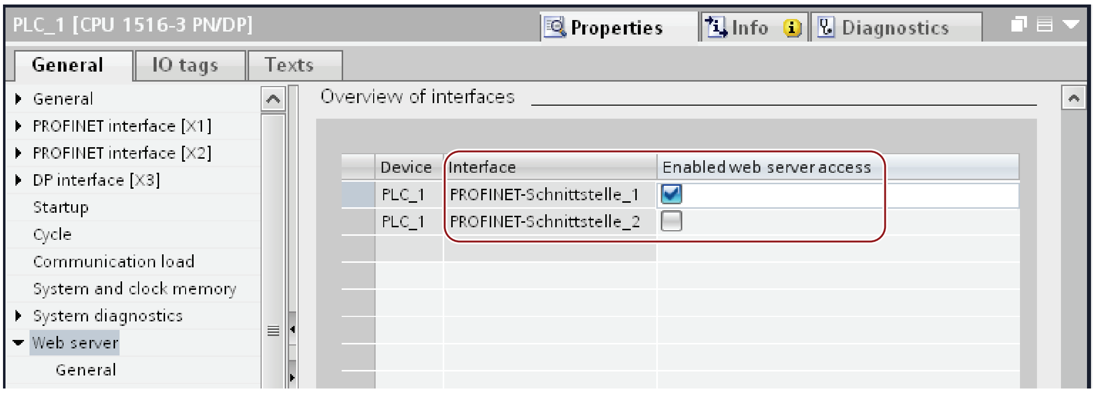
- **User pages**

  In the "User pages" area you can download your own websites to the CPU and make available your own web applications by means of the web browser.

  You can find additional information, in chapter User pages (Page 44). It also includes references for creating user pages for the Web servers of the S7-300 or S7-1200 CPUs.

- **Activation of the Web server for specific interfaces**

  In the area "Overview of interfaces", you have the option to release or block access to the Web server.



Overview of interfaces

Device	Interface	Enabled web server access
PLC_1	PROFINET-Schnittstelle_1	☑
PLC_1	PROFINET-Schnittstelle_2	☐

Figure 2-5 Activation of access to the Web server via the interfaces

2.3 Language settings

Introduction

The Web server provides messages and diagnostic information in the following languages:

- German (Germany)
- English (U.S.)
- French (France)
- Italian (Italy)
- Spanish (traditional sorting)
- Chinese (Simplified)
- Japanese

The two Asian languages can be combined as follows:

- Chinese with English
- Japanese with English

Requirements for the availability of the Asian languages

The following requirements must be met for the Chinese and Japanese languages:

- The corresponding support package for East Asian languages is installed on the display device (e.g., PC).
  For this purpose, activate the "Install files for East Asian languages" option in the "Regional and Language Options > Languages" tab of the Windows Control Panel.
- STEP 7 for Asian languages is installed on the programming device used to configure the CPU.

Note

Asian languages are not supported for SIMATIC HMI devices with Windows CE operating system.

Requirements for multilingual output of text

The following two language settings must be made in STEP 7 to permit that the Web server displays the different languages correctly:

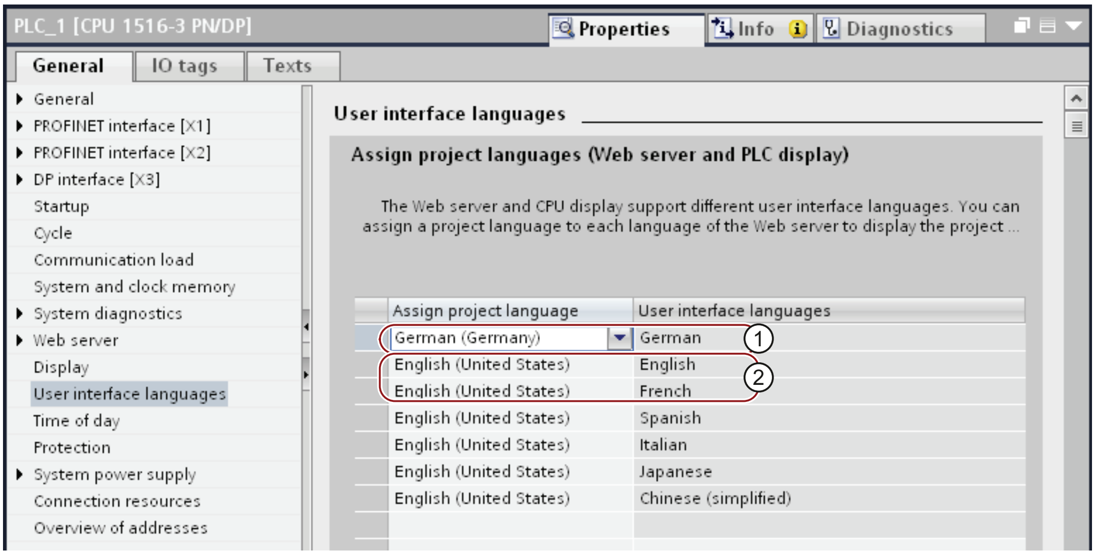
- Setting the Web server languages in the properties dialog of the CPU
- Assign one project language each to the selected languages

Note

The project languages you want to assign must be activated and the corresponding texts (translations) must be available in the project. The project languages selection is available in the project tree under "Languages & Resources".

Setting the language for the web

Once you have activated the Web server on your module, set up to two languages for the Web server and assign them a project language each from the drop-down list.



User interface languages

Assign project languages (Web server and PLC display)

The Web server and CPU display support different user interface languages. You can assign a project language to each language of the Web server to display the project ...

Assign project language	User interface languages
German (Germany)	German
English (United States)	English
English (United States)	French
English (United States)	Spanish
English (United States)	Italian
English (United States)	Japanese
English (United States)	Chinese (simplified)

Figure 2-6 Language settings for the Web server in STEP 7

You can assign all relevant languages the same project language:

- ① Language for Web server: German, assigned project language: German,
- ② Language for Web server: English (U.S.), assigned project language: English (U.S.), Language for Web server: French, assigned project language: English (U.S.)

If you want to use French as second language for the Web server instead of English, English (U.S.) remains set as project language because of the selected assignment.

Reference

Additional information on how to set the project language in STEP 7, is available in the online help for STEP 7, keyword: "Selecting project languages".

See also

Configuring the Web server (Page 11)

2.4 Updating and storing information

Update status of the screen content

Automatic updating is activated in the basic setting. The preset update time is 10s.

You update the web pages manually using either the function key <F5> or the button 🔄.

Disabling automatic updates for an individual website

Select the button 🔄 Off to temporarily deactivate automatic updating of a website.

You reactivate automatic updates with the <F5> function key or the 🔄 On button.

Note

If the CPU is subject to high load, for example, when processing a large number of PROFINET interrupts or extensive communication jobs, the update of websites may take a lot longer for the duration of such high CPU load.

Saving alarms and diagnostic buffer entries

You can save alarms and diagnostic buffer entries to a csv file for further processing in a spreadsheet calculation program or database program. Save the data by clicking the 💾 icon.

A dialog opens in which you can specify the file name and target directory.

Websites 3

3.1 Start page with general CPU information

Connecting to the Web server

Set up a connection to the Web server by entering the IP address of the configured CPU in the address bar of the web browser, for example, http://192.168.3.141 or https://192.168.3.141. The connection is set up and the "Intro" page opens.

The examples in the next section provide information about the different websites.

Intro

The figure below shows the first page (Intro) called by the Web server.

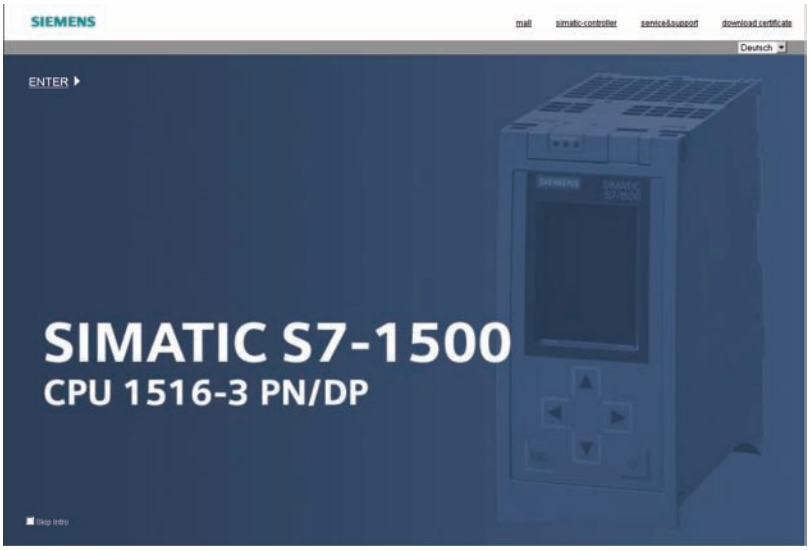


Figure 3-1 Welcome page of the Web server of the CPU 1516-3 PN/DP

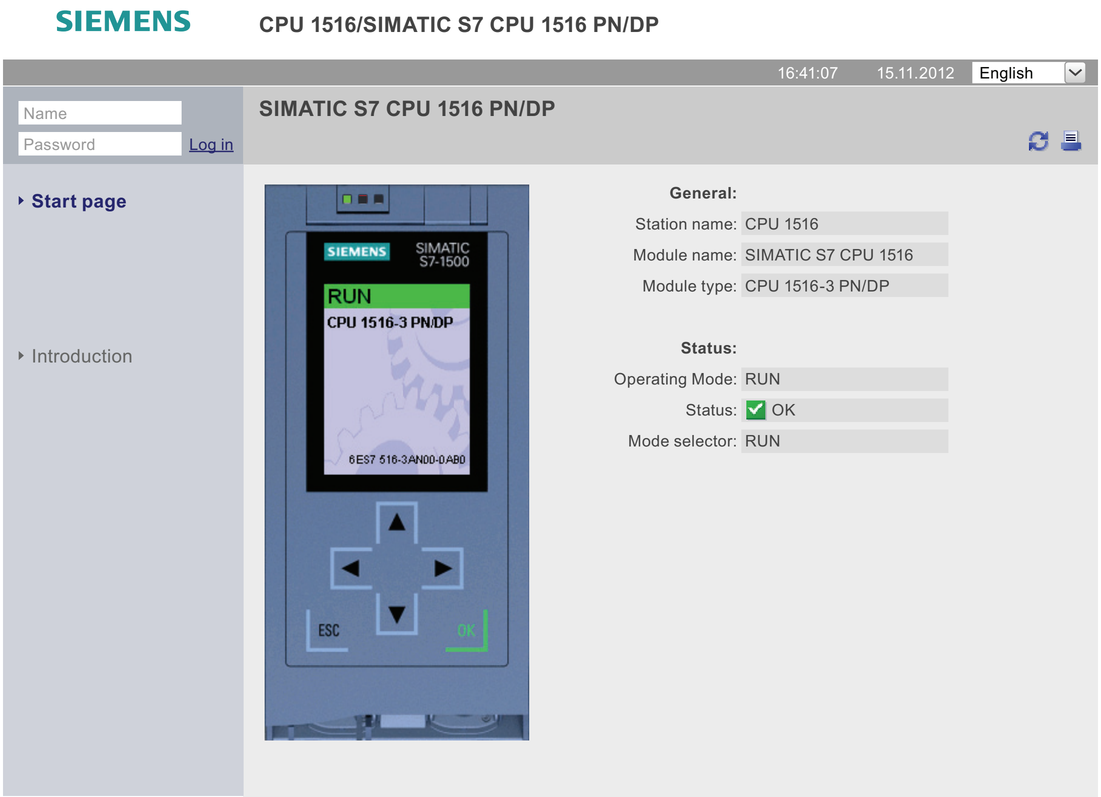
Click the ENTER link to go to the Web server pages.

Note

Select the "Skip Intro" check box in order to skip the Intro. As of now, the Web server will take you directly to its start page. You can undo the setting "Skip Intro" by clicking the "Intro" link on the Start page.

Start page

Before the log in, the Start page offers information as shown in the figure below. The CPU image with LEDs returns the actual CPU status at the time of the data request.



SIEMENS CPU 1516/SIMATIC S7 CPU 1516 PN/DP

16:41:07 15.11.2012 English

Name
Password Log in

SIMATIC S7 CPU 1516 PN/DP

- Start page
- Introduction

General:

Station name: CPU 1516
Module name: SIMATIC S7 CPU 1516
Module type: CPU 1516-3 PN/DP

Status:

Operating Mode: RUN
Status: ✓ OK
Mode selector: RUN

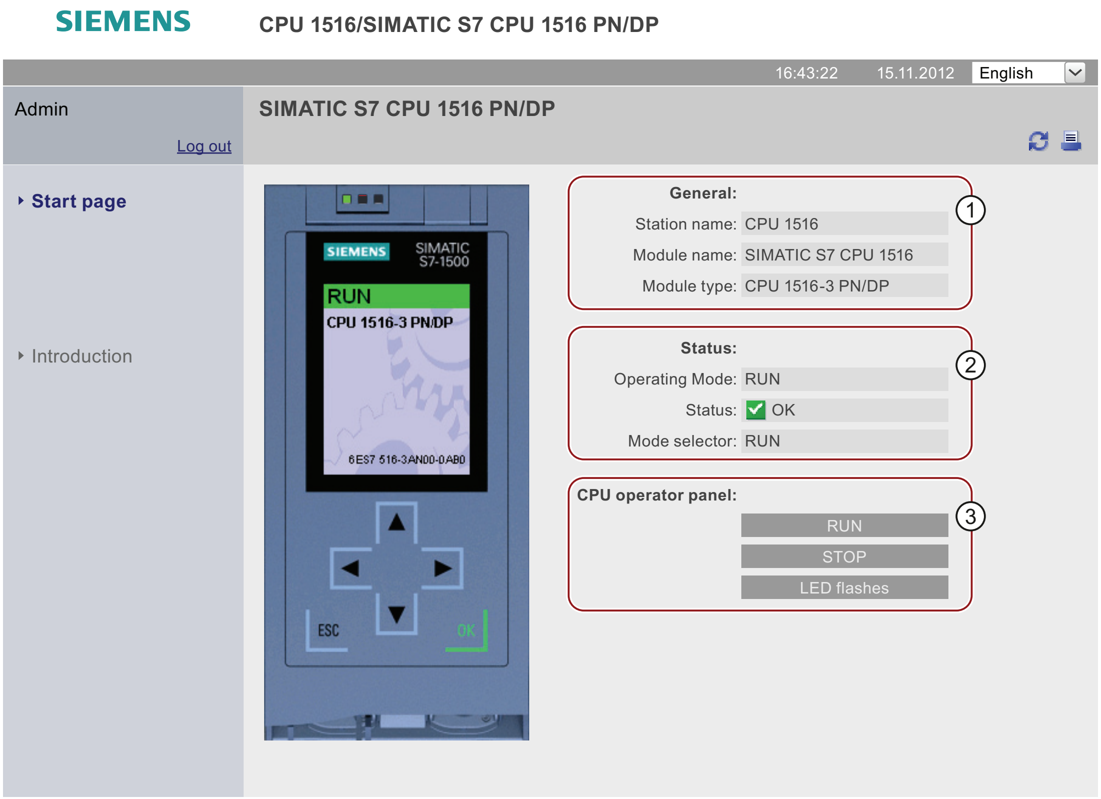
Figure 3-2 Start page before log in

Log in

You must be logged in in order to use the full functionality of the websites. Log in with a user name and password specified in the web configuration in STEP 7. You now have corresponding permissions to access the websites released for this user. If you have not configured a user, read-only access is granted to welcome and home pages by default.

Note

After carrying out your required actions, log out explicitly from the Web server by clicking "Logout" in order to minimize the risk of unauthorized external access.



SIEMENS CPU 1516/SIMATIC S7 CPU 1516 PN/DP

16:43:22 15.11.2012 English

Admin

SIMATIC S7 CPU 1516 PN/DP

General:
Station name: CPU 1516
Module name: SIMATIC S7 CPU 1516
Module type: CPU 1516-3 PN/DP

Status:
Operating Mode: RUN
Status: ✓ OK
Mode selector: RUN

CPU operator panel:

Figure 3-3 Start page after login

① "General"

"General" contains information about the CPU with whose web server you are currently connected.

② "Status"

"Status" contains information about the CPU at the moment of the query.

③ "CPU operator panel"

If you have the necessary access permissions, you can change the operating mode of the CPU ("RUN"/"STOP" fields) or have the LEDs flash ("LED flashes" field) in the section "CPU operator panel".

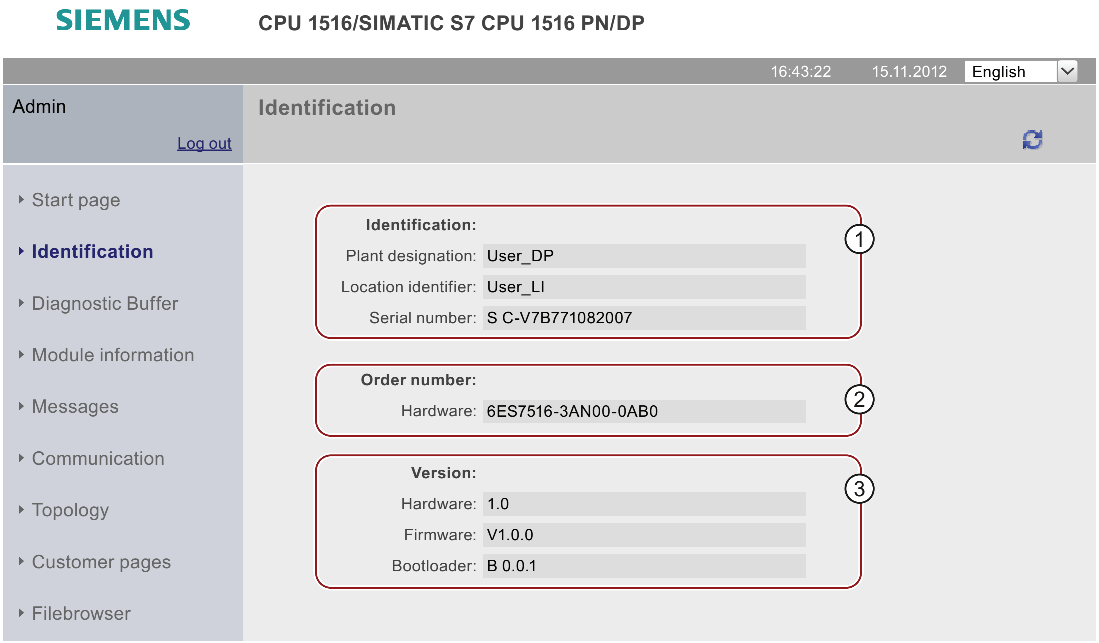
Reference

You can find additional information in the chapter Configuring the Web server (Page 11).

3.2 Identification

Characteristics

The CPU characteristics are available on the "Identification" website.



Identification:
Plant designation: User_DP
Location identifier: User_LI
Serial number: S C-V7B771082007

Order number:
Hardware: 6ES7516-3AN00-0AB0

Version:
Hardware: 1.0
Firmware: V1.0.0
Bootloader: B 0.0.1

Figure 3-4 Identification

① "Identification"

The "Identification" info box contains the plant and location designation and the serial number. Plant and location designations can be configured in STEP 7 in the properties dialog box of the CPU, "General" tab.

② "Order number"

The "Order number" info box contains the order number for the hardware.

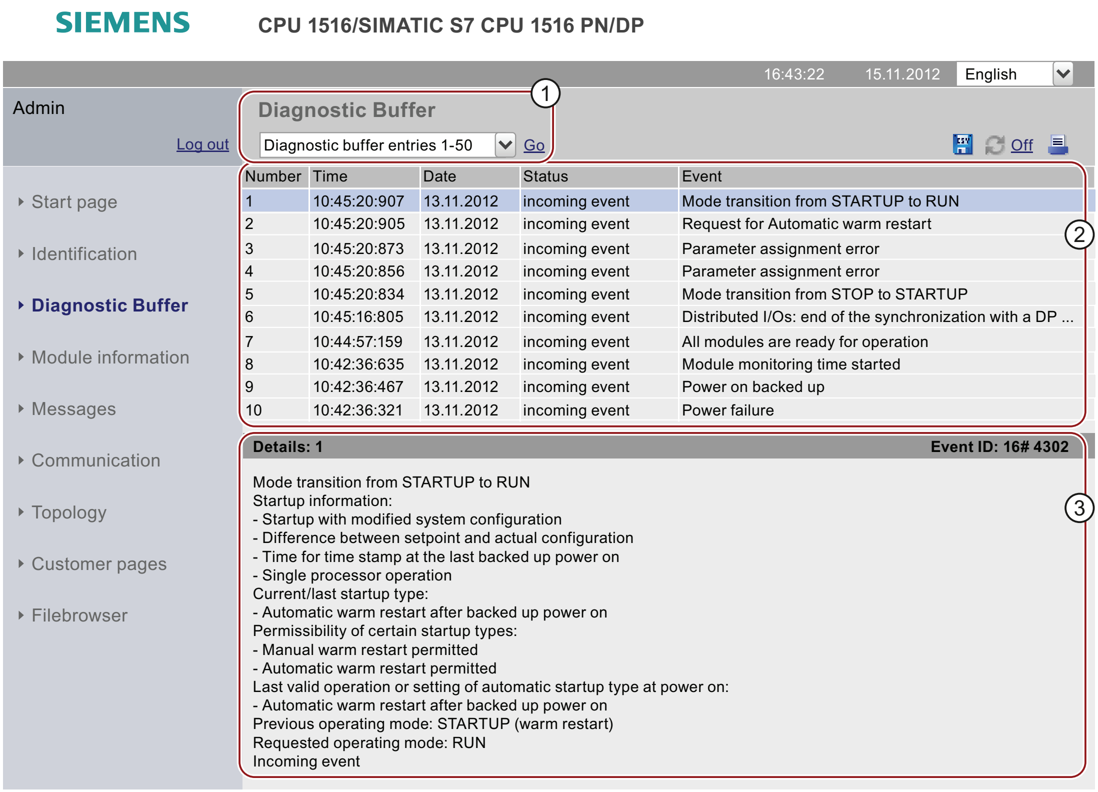
③ "Version"

You can find the hardware, firmware and bootloader versions in the "Version" info box.

3.3 Diagnostic buffer

Diagnostic buffer

The browser displays the content of the diagnostic buffer on the "Diagnostic buffer" website.



Number	Time	Date	Status	Event
1	10:45:20:907	13.11.2012	incoming event	Mode transition from STARTUP to RUN
2	10:45:20:905	13.11.2012	incoming event	Request for Automatic warm restart
3	10:45:20:873	13.11.2012	incoming event	Parameter assignment error
4	10:45:20:856	13.11.2012	incoming event	Parameter assignment error
5	10:45:20:834	13.11.2012	incoming event	Mode transition from STOP to STARTUP
6	10:45:16:805	13.11.2012	incoming event	Distributed I/Os: end of the synchronization with a DP ...
7	10:44:57:159	13.11.2012	incoming event	All modules are ready for operation
8	10:42:36:635	13.11.2012	incoming event	Module monitoring time started
9	10:42:36:467	13.11.2012	incoming event	Power on backed up
10	10:42:36:321	13.11.2012	incoming event	Power failure

Details: 1 **Event ID: 16# 4302**

Mode transition from STARTUP to RUN
Startup information:
- Startup with modified system configuration
- Difference between setpoint and actual configuration
- Time for time stamp at the last backed up power on
- Single processor operation
Current/last startup type:
- Automatic warm restart after backed up power on
Permissibility of certain startup types:
- Manual warm restart permitted
- Automatic warm restart permitted
Last valid operation or setting of automatic startup type at power on:
- Automatic warm restart after backed up power on
Previous operating mode: STARTUP (warm restart)
Requested operating mode: RUN
Incoming event

Figure 3-5 Diagnostic buffer

Requirements

The Web server is activated, languages are set, and the project is compiled and downloaded in STEP 7.

① "Diagnostic buffer entries 1-50"

The diagnostic buffer can save up to 3200 (CPU 1516) or 1000 messages (CPU 1513/1511). Select an interval for the entries from the drop-down list. Each interval comprises 50 entries.

② "Event"

The "Event" info box contains the diagnostic interrupts with date and time.

③ "Details"

This field outputs detailed information about a selected event. Select the corresponding event from the ② "Event" info field.

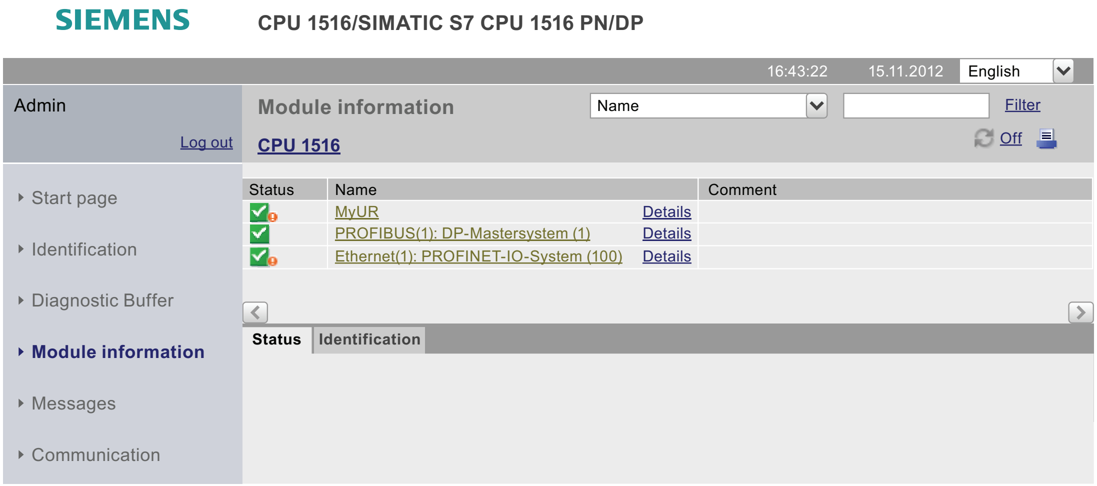
3.4 Module information

Requirements

- You have made the following settings in the project view, "Properties" tab:
  - Web server activated
  - Assigned a project language to the language in the Web server
- You have compiled and downloaded the project with STEP 7.

Module information

The state of a station is indicated by means of symbols and comments on the "Module information" website.



Status	Name		Comment
✓	MyUR	Details	
✓	PROFIBUS(1): DP-Mastersystem (1)	Details	
✓	Ethernet(1): PROFINET-IO-System (100)	Details	

Figure 3-6 Module information

Meaning of the symbols in the "Symbol" column

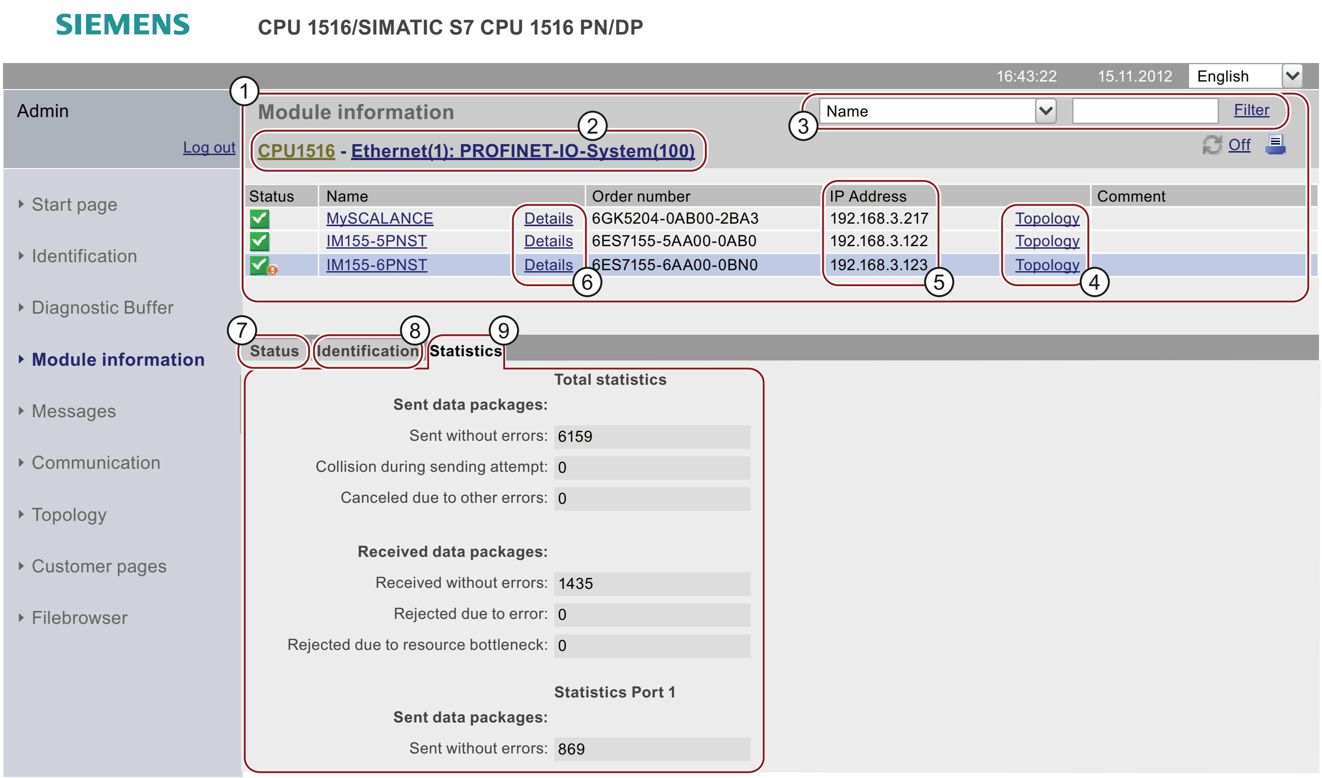
Table 3- 1 Meaning of symbols

Symbol	Symbol color	Meaning
✅	green	Component is OK
✔	gray	Deactivated PROFIBUS slaves or PROFINET devices.
?	black	State cannot be determined • The "Status cannot be determined" is displayed during system diagnostics for all configured I/O modules and I/O systems after restart of the CPU. • However, this status can also be displayed temporarily during operation if a diagnostic interrupt burst occurs for all modules. • It is not possible to determine the status of modules on a subsystem that is connected to a CP.
❌	red	Component failed or is not reachable • "Not reachable" is displayed for e.g. a module that has been removed or a module that has been configured but does not exist.
🔧	green	Maintenance required (Maintenance Required)
🔧	yellow	Maintenance demanded (Maintenance Demanded)
🔧	red	Error - component faulty or not available due to an incorrect type
❗	-	A module in a lower module level does not have the status "Component OK"

Navigation to further module levels

The status of individual components/modules/submodules is displayed when you navigate to the further module levels:

- To the next higher module level using the links in the display of the module levels
- To the next lower module level using the links in the "Name" column



SIEMENS CPU 1516/SIMATIC S7 CPU 1516 PN/DP

16:43:22 15.11.2012 English

Admin

Log out

- Start page
- Identification
- Diagnostic Buffer
- **Module information**
- Messages
- Communication
- Topology
- Customer pages
- Filebrowser

Module information

CPU1516 - Ethernet(1): PROFINET-IO-System(100)

Name Filter

Off

Status	Name		Order number	IP Address		Comment
✓	MySCALANCE	Details	6GK5204-0AB00-2BA3	192.168.3.217	Topology	
✓	IM155-5PNST	Details	6ES7155-5AA00-0AB0	192.168.3.122	Topology	
✓	IM155-6PNST	Details	6ES7155-6AA00-0BN0	192.168.3.123	Topology	

Status | Identification | **Statistics**

Total statistics

Sent data packages:

Sent without errors: 6159

Collision during sending attempt: 0

Canceled due to other errors: 0

Received data packages:

Received without errors: 1435

Rejected due to error: 0

Rejected due to resource bottleneck: 0

Statistics Port 1

Sent data packages:

Sent without errors: 869

Figure 3-7 Navigation to further module levels

① "Module information"

Depending on the selected level, the table contains information about the rack, the DP master system, the PROFINET IO master system, about the devices, the individual modules, as well as the modules or submodules of the station.

② "Display of the module levels"

The links are used to access the "Module information" of the higher module levels.

③ "Filter"

You can search in the table by selecting specific criteria:

1. Select a parameter from the drop-down list:
   - Slot
   - Name
   - Order number
   - I-address
   - O-address
   - Comment
2. Enter the value of the selected parameter.
3. Click "Filter".

Note

The filter criteria are also retained when you update a page.

To deactivate the filter settings, click "Filter" again.

④ "Topology"

The two websites, "Topology" and "Module information", are linked. A click on "Topology" of the selected module automatically takes you to this module in the graphic view of the target topology on the "Topology" website. The module is displayed in the visible area of the "Topology" website. The device header of the selected module flashes for a few seconds.

⑤ "IP address"

If a link is available, you can use it to access this Web server of the configured device you selected.

⑥ "Details"

Further information about the selected module is provided in the "Status" and "Identification" tabs via the "Details" link.

⑦ "Status" tab

The tab contains information about the status of the selected module when a fault or alarm exists.

⑧ "Identification" tab

The tab contains data on the identification of the selected module.

Note

This tab displays only the data configured offline of the module.

⑨ "Statistics" tab

The tab is only displayed for PROFINET IO devices and includes the following information on communication statistics of the selected IO device:

- "Overall statistics - Data packets sent"

  The quality of the data transfer on the send line can be determined from the key data in this info box.

- "Overall statistics - Data packets received"

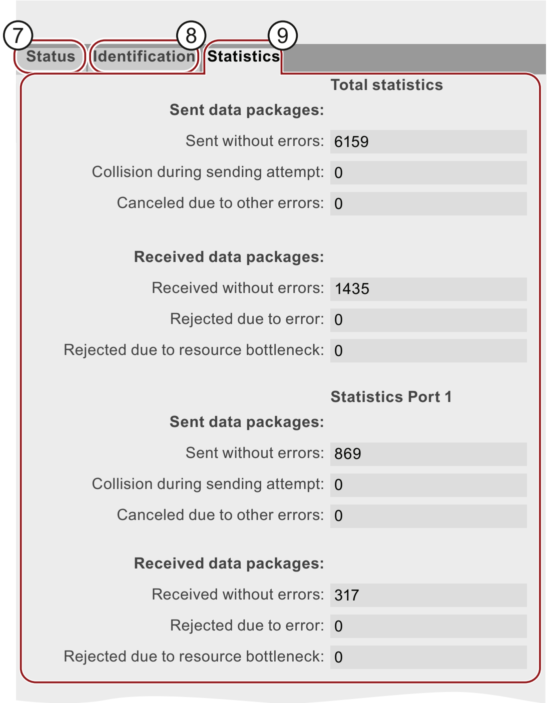
  The quality of the data transfer on the reception line can be determined from the key data in this info box.

- "Statistics port x - Data packets sent"

  The quality of the data transfer on the send line can be determined from the key data in this info box.

- "Statistics port x - Data packets received"

  The quality of the data transfer on the reception line can be determined from the key data in this info box.



Total statistics

Sent data packages:

Sent without errors: 6159
Collision during sending attempt: 0
Canceled due to other errors: 0

Received data packages:

Received without errors: 1435
Rejected due to error: 0
Rejected due to resource bottleneck: 0

Statistics Port 1

Sent data packages:

Sent without errors: 869
Collision during sending attempt: 0
Canceled due to other errors: 0

Received data packages:

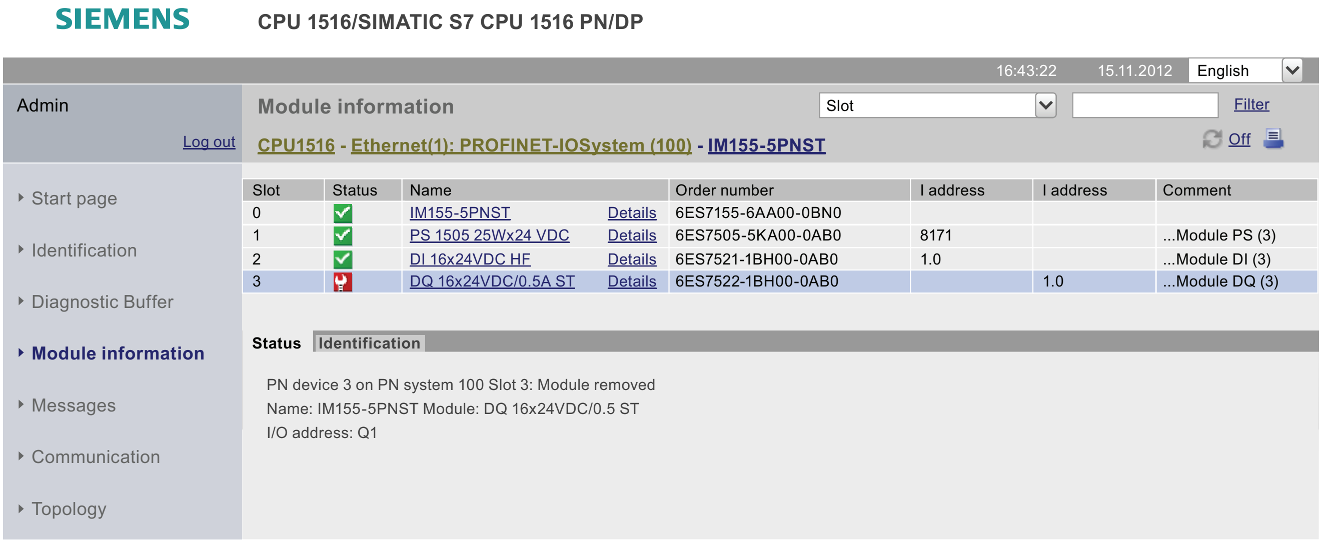
Received without errors: 317
Rejected due to error: 0
Rejected due to resource bottleneck: 0

Figure 3-8 Tabs

Reference

You can find additional information in the "Statistics" tab, in the chapter Communication (Page 32).

Example: Module information - module



Module information

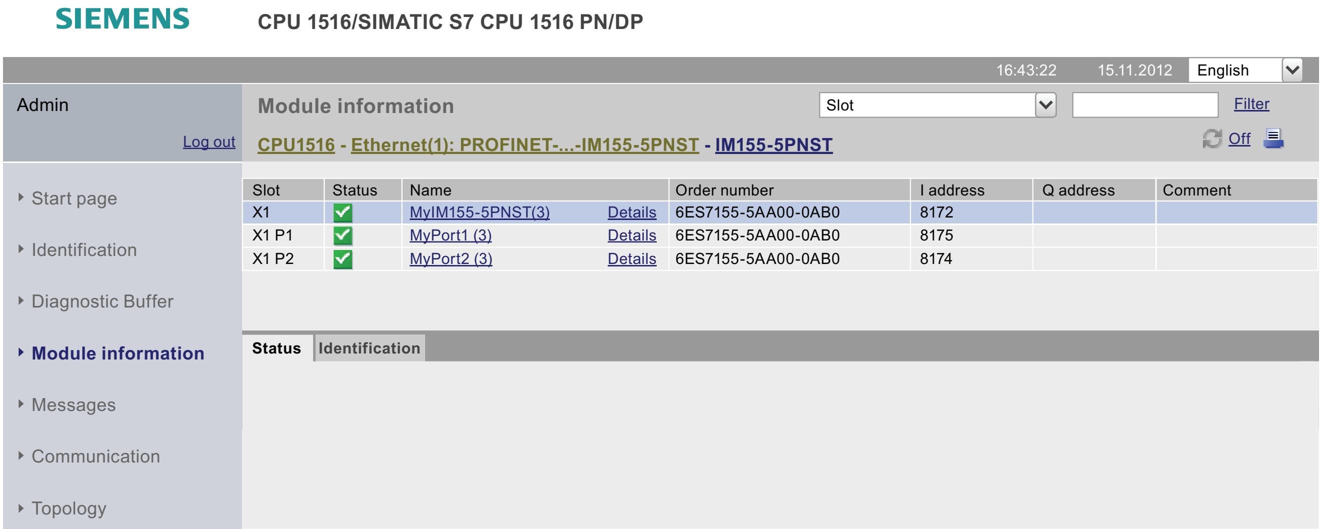
CPU1516 - Ethernet(1): PROFINET-IOSystem (100) - IM155-5PNST

Slot	Status	Name		Order number	I address	I address	Comment
0	✓	IM155-5PNST	Details	6ES7155-6AA00-0BN0			
1	✓	PS 1505 25Wx24 VDC	Details	6ES7505-5KA00-0AB0	8171		...Module PS (3)
2	✓	DI 16x24VDC HF	Details	6ES7521-1BH00-0AB0	1.0		...Module DI (3)
3	🟥	DQ 16x24VDC/0.5A ST	Details	6ES7522-1BH00-0AB0		1.0	...Module DQ (3)

PN device 3 on PN system 100 Slot 3: Module removed
Name: IM155-5PNST Module: DQ 16x24VDC/0.5 ST
I/O address: Q1

Figure 3-9 Example: Module information - module

Example: Module information - submodule



Module information

CPU1516 - Ethernet(1): PROFINET-...-IM155-5PNST - IM155-5PNST

Slot	Status	Name		Order number	I address	Q address	Comment
X1	✓	MyIM155-5PNST(3)	Details	6ES7155-5AA00-0AB0	8172		
X1 P1	✓	MyPort1 (3)	Details	6ES7155-5AA00-0AB0	8175		
X1 P2	✓	MyPort2 (3)	Details	6ES7155-5AA00-0AB0	8174		

Figure 3-10 Example: Module information - submodule

Reference

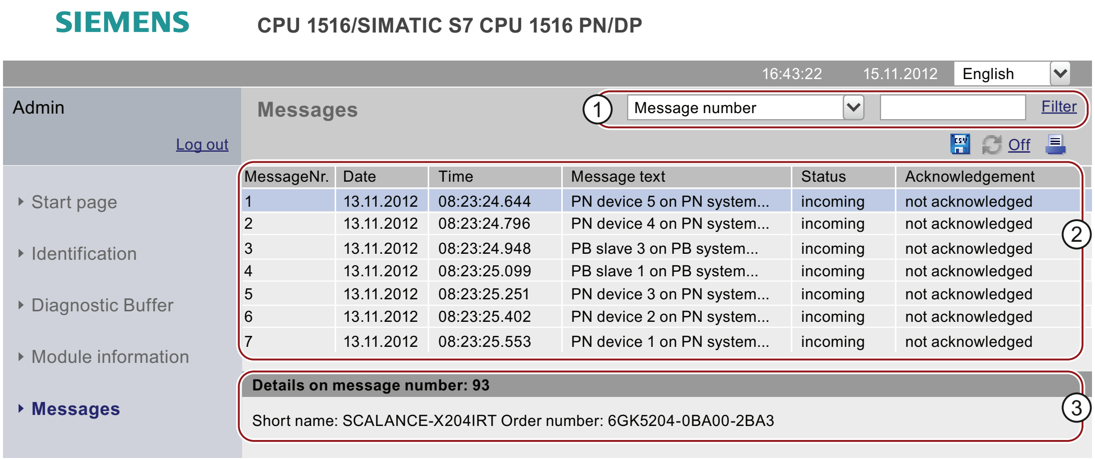
You can find additional information on the "Module information" in the online help for STEP 7, keyword: "Module information".

3.5 Alarms

Alarms

To receive compact information on fault analysis, we recommend that you first read out the content of the alarm buffer. This is the most effective method to get an overview of the pending faults.

The browser displays the content of the alarm buffer on the "Alarms" website.
The alarms cannot be acknowledged on the Web server.



MessageNr.	Date	Time	Message text	Status	Acknowledgement
1	13.11.2012	08:23:24.644	PN device 5 on PN system...	incoming	not acknowledged
2	13.11.2012	08:23:24.796	PN device 4 on PN system...	incoming	not acknowledged
3	13.11.2012	08:23:24.948	PB slave 3 on PB system...	incoming	not acknowledged
4	13.11.2012	08:23:25.099	PB slave 1 on PB system...	incoming	not acknowledged
5	13.11.2012	08:23:25.251	PN device 3 on PN system...	incoming	not acknowledged
6	13.11.2012	08:23:25.402	PN device 2 on PN system...	incoming	not acknowledged
7	13.11.2012	08:23:25.553	PN device 1 on PN system...	incoming	not acknowledged

Details on message number: 93

Short name: SCALANCE-X204IRT Order number: 6GK5204-0BA00-2BA3

Figure 3-11 Alarms

Requirements

The message texts were configured in the user-specific languages. For information about configuring message texts refer to STEP 7 and to the Service&Support pages (http://www.siemens.com/automation/service&support).

① "Filter"

You can filter the table based on specific criteria.

1. Select a parameter from the drop-down list.
2. Enter the value of the selected parameter.
3. Click "Filter".

To deactivate the filter settings, click "Filter" again.

Effects

- The filter settings are also retained when you update a page.
- Filter settings have no effect on the print-out. A printout always contains the entire content of the message buffer.

② "Alarms"

CPU alarms are displayed in chronological order in info box ②, including **date** and **time**.

The **message text** parameter is an entry which contains the message texts configured for the corresponding fault definitions.

Sorting

You can also view the individual parameters in ascending or descending order. For this purpose, click on one of the parameters in the column header:

- Message number
- Date
- Time
- Message text
- Status
- Acknowledgment

The messages are returned in chronological order when you click the "Date" entry. Incoming and outgoing events are output at the **Status** parameter.

③ "Message number details"

You can view detailed alarm information in this info box. Select the corresponding alarm from the info field ②.

3.6 Communication

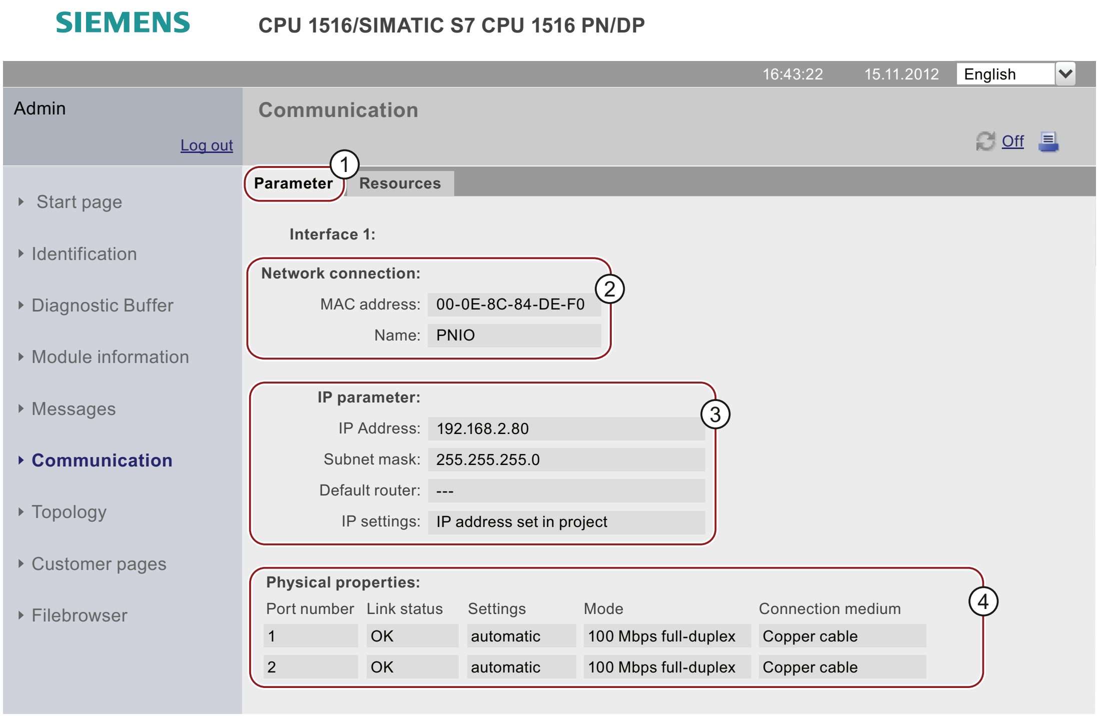
Overview

The "Communication" website provides more information about the following tabs:

- Parameters
- Resources

① "Parameters" tab

A summary of the information on the PROFINET interfaces of the selected CPU is available in the "Parameters" tab.



Interface 1:

Network connection:

MAC address: 00-0E-8C-84-DE-F0

Name: PNIO

IP parameter:

IP Address: 192.168.2.80

Subnet mask: 255.255.255.0

Default router: ---

IP settings: IP address set in project

Physical properties:

Port number	Link status	Settings	Mode	Connection medium
1	OK	automatic	100 Mbps full-duplex	Copper cable
2	OK	automatic	100 Mbps full-duplex	Copper cable

Figure 3-12 Parameters of the integrated PROFINET interface

② "Network connection"

The item "Network connection" includes information for identification of the integrated PROFINET interface of the corresponding CPU. The MAC address is located on the CPU above the respective PROFINET interface.

③ "IP parameters"

This parameter includes information on the configured IP address and number of the subnet in which the corresponding CPU is located.

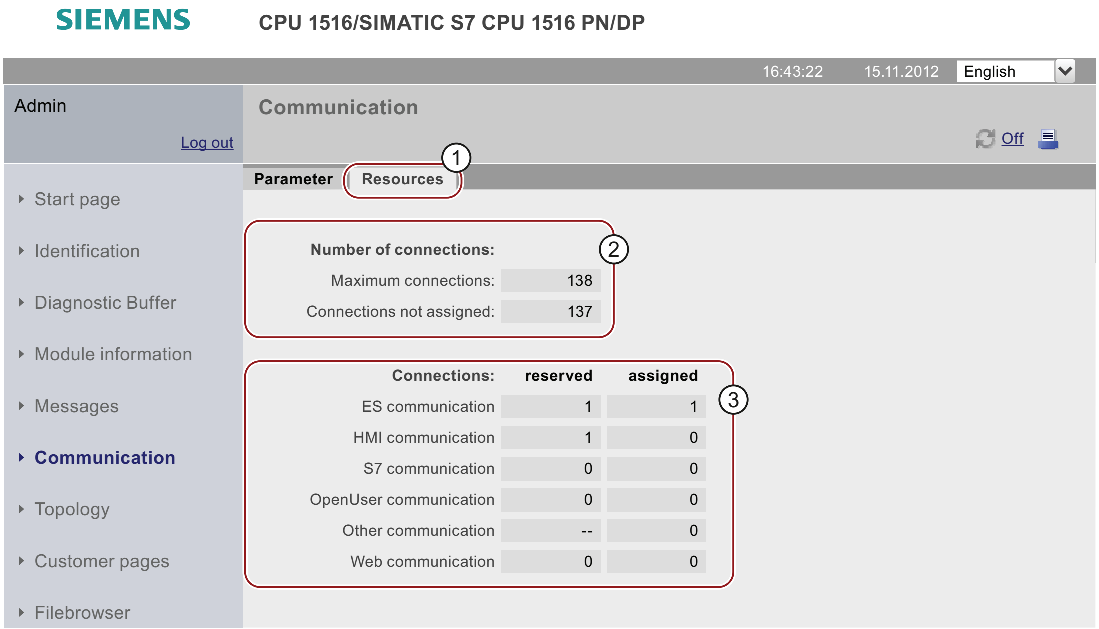
④ "Physical properties"

In a communication window there is additional information on the interface layout in the field "Physical properties".
It includes information on:

- Port number
- Link status
- Settings
- Mode
- Connection medium

① The "Resources" tab

For information about the load of connections on resources, refer to the "Resources" tab.



Number of connections:	
Maximum connections:	138
Connections not assigned:	137

Connections:	reserved	assigned
ES communication	1	1
HMI communication	1	0
S7 communication	0	0
OpenUser communication	0	0
Other communication	--	0
Web communication	0	0

Figure 3-13 "Resources" tab

② Number of connections

The item "Number of connections" includes information on the maximum number of connections and the number of available connections.

③ Connections

The item "Connections" provides information on the number of connections reserved or used for ES, HMI, S7, OpenUser, web communication and other communication functions.

3.7 Topology

3.7.1 Introduction

Topology of the PROFINET devices

The "Topology" website provides information on the topological configuration and status of the PROFINET devices on your PROFINET IO system.

There are three tabs for the following views:

- Graphical view (target and actual topology)
- Tabular view (actual topology only)
- Status overview (excluding topological correlations)

The tabular view and status overview can be printed. Before printing, use the print preview of your browser and, if necessary, correct the format.

Target topology

Display of the PROFINET devices and connections of a PROFINET IO system configured in the topology view of STEP 7. The corresponding status of these devices is displayed.

The display includes neighboring PROFINET devices, provided their topological layout is configured as well. The status is not displayed.

This view identifies the topological assignment of PROFINET devices that have failed, the differences between the target and actual topology, and interchanged ports.

Note

The configured target topology is always displayed by default in the following scenarios:

- When the "Topology" website is called via the navigation bar
- When you change from the overview of PROFINET IO devices on the "Module information" website to the "Topology" website by means of the "Topology" link.

If a target topology was not configured, the actual topology is called by default.

Actual topology

Displays the actual topological layout of the "configured" PROFINET devices of a PROFINET IO system and the directly adjacent, non-configured PROFINET devices (display of the relations, provided these can be determined; but the status of these adjacent PROFINET devices is not displayed).

3.7.2 Graphical view

Requirements

For error-free operation of the topology, the following conditions must be met:

- You completed the Language settings (Page 16).
- In the Topology Editor of STEP 7, you configured the topological interconnection of ports (requirement for display of the target topology and the corresponding topological target connections).
- You compiled the project in STEP 7.
- The download of all project data is completed (configuration and program data).

Target and actual topology - graphical view

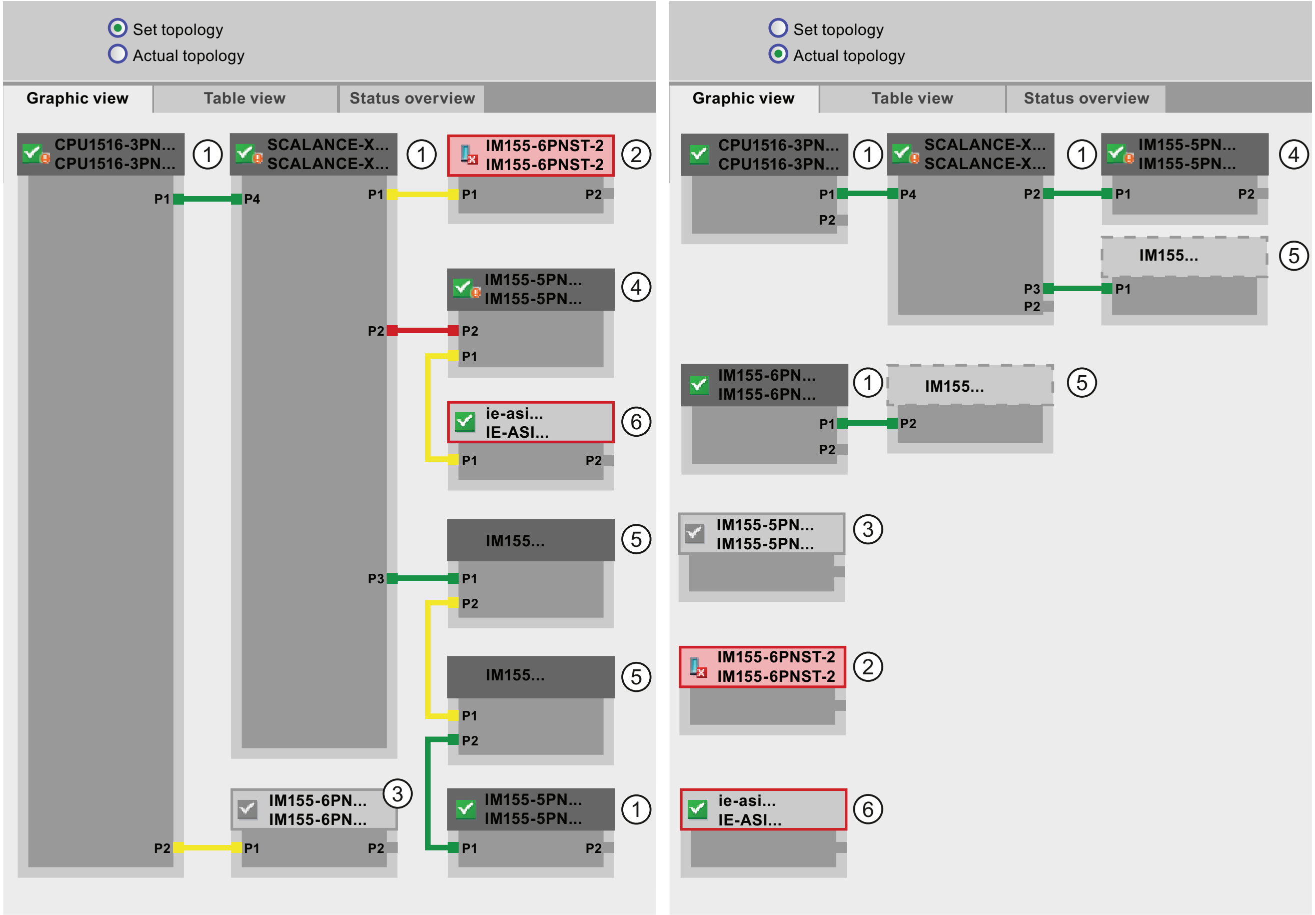


Figure 3-14 Graphical view - Target and actual topology

Meaning of the colored connections in the target/actual topology:

Table 3- 2 Meaning of the colored connections in the target/actual topology

Connection	Meaning	
	Target topology	Actual topology
green	The current actual connection matches the configured target connection.	Connections detected
red	Mismatch between the current actual connection and the configured target connection (e.g., port interchanged).	-
yellow	Connection diagnostics not possible. Causes: • Malfunction of communication with a device (e.g., cable was removed) • Connection to a passive component (e.g., switches or cables) • Connection to devices/PROFINET devices on a different IO controller or IO subsystem.	-

① Configured and accessible PROFINET devices

Configured and accessible PROFINET devices are displayed in dark gray. Connections indicate the ports used to connect the PROFINET devices of a station.

② Configured but inaccessible PROFINET devices

Configured but inaccessible PROFINET devices are indicated in pink color with red frame (e.g., device failure, cable disconnected).

③ Deactivated devices

All deactivated, configured PROFINET devices are displayed in light gray.

④ Interchanged ports

Interchanged ports are highlighted in red color in the target topology view. The actual topology view indicates the actually connected ports, while the target topology view displays the configured target connections.

⑤ PROFINET devices of a different PROFINET IO subsystem

- In the target topology:

  A PROFINET device of a different PROFINET IO subsystem is indicated by means of a green link (or red link for interchanged ports) if it is available on the bus and directly adjacent to an accessible configured PROFINET device ①. If the PROFINET device of a different PROFINET IO is unaccessible, it is identified by means of a yellow connecting line.
  The connection between two PROFINET devices which both belong to a different PROFINET IO subsystem cannot be identified and is always indicated in yellow color.

- In the actual topology:

  The PROFINET device of a different PROFINET IO subsystem is not displayed unless it is directly adjacent to a configured PROFINET device. The PROFINET device is shown in light gray with a dashed line around the device header.

The status of PROFINET devices of a different PROFINET IO subsystem is **not** displayed in the device header.

⑥ Displaying faulty neighbor relationships

Devices from which the relation data could not be read completely or with error are highlighted in light gray with a red frame.

Note

Displaying faulty neighbor relationships

If a device does not have the matching firmware, the relationships cannot be displayed correctly. This means a firmware update of the respective device is required in case a faulty neighbor relationship is displayed.

Views after changes to the configuration

- If a device fails, it remains at the same position in the "Target topology" view. This error state is indicated with a red border around the device header and the icon ☒.
- If a device fails, it is displayed in the "Actual topology" view. This error state is indicated separately in the bottom area with a red border around the device header and the icon ☒.

Link between the "Topology" and "Module information" websites

The two websites, "Topology" and "Module information", are linked. A click on the header of a selected module in the topology view automatically takes you to this module on the "Module information" website.

You can find additional information on this in the chapter Module information (Page 24).

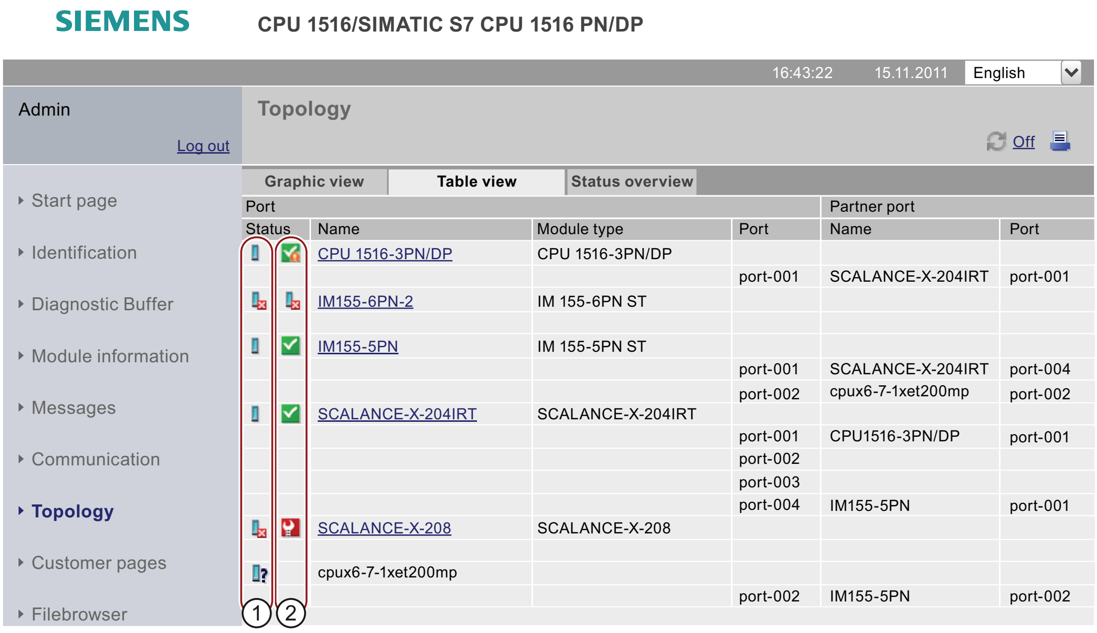
Reference

Additional examples for graphical topology view are available in the chapter Examples for graphical topology views (Page 41).

3.7.3 Tabular view

Topology - tabular view

The "Tabular view" always shows the "Actual topology".



Port					Partner port	
Status		Name	Module type	Port	Name	Port
	✓	CPU 1516-3PN/DP	CPU 1516-3PN/DP			
				port-001	SCALANCE-X-204IRT	port-001
☒	☒	IM155-6PN-2	IM 155-6PN ST			
	✓	IM155-5PN	IM 155-5PN ST			
				port-001	SCALANCE-X-204IRT	port-004
				port-002	cpux6-7-1xet200mp	port-002
	✓	SCALANCE-X-204IRT	SCALANCE-X-204IRT			
				port-001	CPU1516-3PN/DP	port-001
				port-002		
				port-003		
				port-004	IM155-5PN	port-001
☒		SCALANCE-X-208	SCALANCE-X-208			
?		cpux6-7-1xet200mp				
				port-002	IM155-5PN	port-002

Figure 3-15 Topology - tabular view

① Meaning of the symbols relating to the status of the PROFINET devices

Table 3- 3 Meaning of the symbols relating to the status of the PROFINET devices

Symbol	Meaning
▯	Configured and accessible PROFINET devices
▯?	Unconfigured and accessible PROFINET devices
▯☒	Configured but inaccessible PROFINET devices
▯!	Devices for which neighbor relations cannot be determined, or for which the neighbor relationship could not be read out completely, or only with errors

② Meaning of the symbols relating to the module status of the PROFINET devices

Table 3- 4 Meaning of the symbols relating to the module status of the PROFINET devices

Symbol	Color	Meaning
✓	green	Component is OK.
✓	gray	Deactivated PROFIBUS slaves or PROFINET devices
?	black	State cannot be determined • For example, "Status cannot be determined" is always displayed while the CPU is in STOP mode, or during startup evaluation of "Report system error" for all configured I/O modules and I/O systems after a CPU restart. • However, this status can also be displayed temporarily during operation if a diagnostic interrupt burst occurs at all modules. • It is not possible to determine the status of modules on a subsystem that is connected to a CP.
☒	red	Component failed or is not reachable • "Not reachable" is displayed for e.g. a module that has been removed or a module that has been configured but does not exist.
🔧	green	Maintenance required (Maintenance Required)
🔧	yellow	Maintenance demanded (Maintenance Demanded)
🔧	red	Error - component faulty or not available due to an incorrect type.
❗	-	A module in a lower module level does not have the status "Component OK".

Reference

You can find additional information on the "Report system error" in the online help for STEP 7 V12. Keyword: "System diagnostics".

3.7.4 Status overview

Topology - status overview

The "Status overview" provides a clear presentation of all PROFINET IO devices/PROFINET devices (without connection relations) on one page. A quick error diagnostics is possible based on the symbols that show the module statuses.

The overview also provides a link of the modules to the "Module information (Page 24)" website.

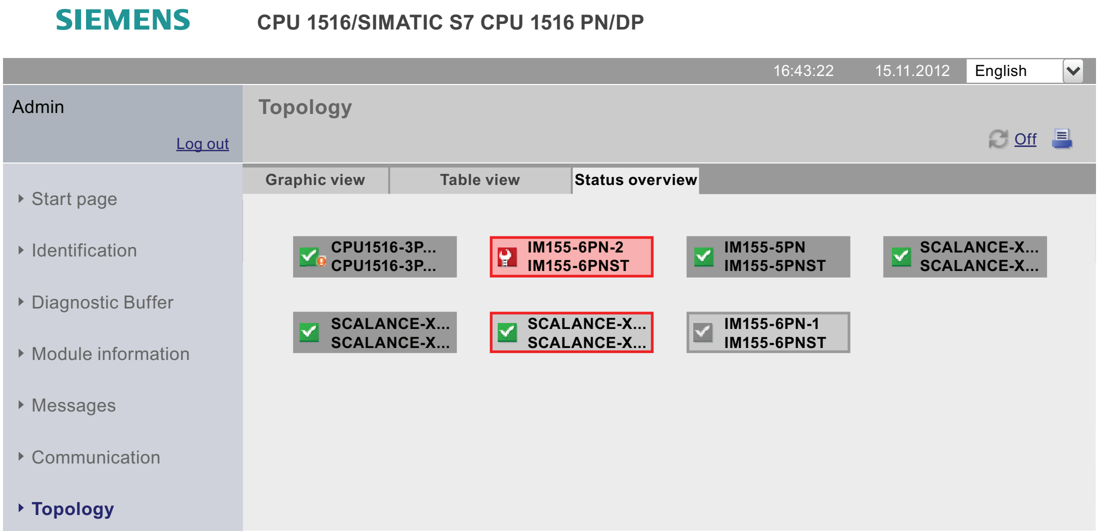


Figure 3-16 Topology - status overview

3.7.5 Examples for graphical topology views

Exemplary information on the different topology views is included in the paragraphs that follow.

"Target topology" is OK

Here you see the connections as they were configured in the topology editor by STEP 7.

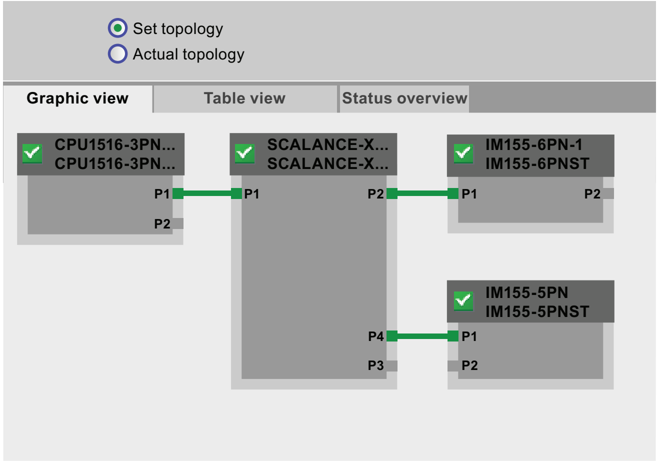

● Set topology
○ Actual topology

Figure 3-17 "Target topology" is OK

"Actual topology" is OK

Shows the actual layout of all configured devices that can be reached topologically.

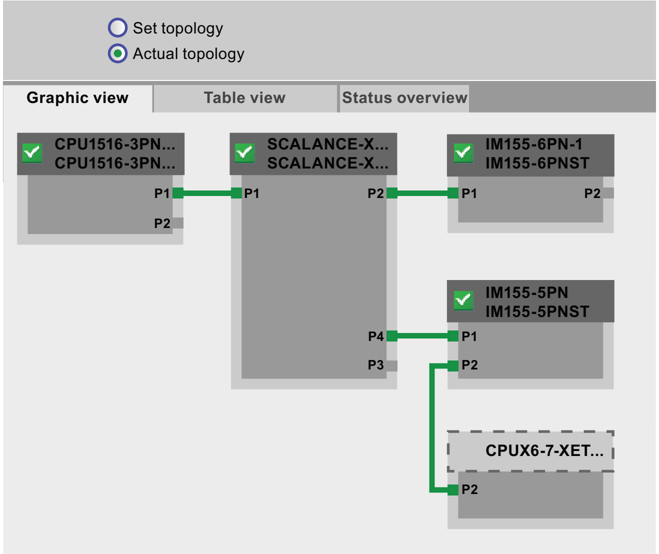
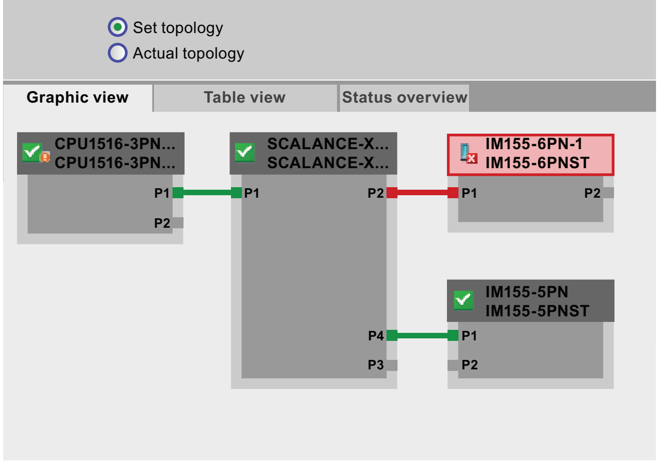

○ Set topology
● Actual topology

Figure 3-18 "Actual topology" is OK

"Target topology" with failed device

If a device has failed in the meantime, this device remains in the same place in the " Target topology" view. The failed device is displayed with a red border around the device header and the icon ☒.

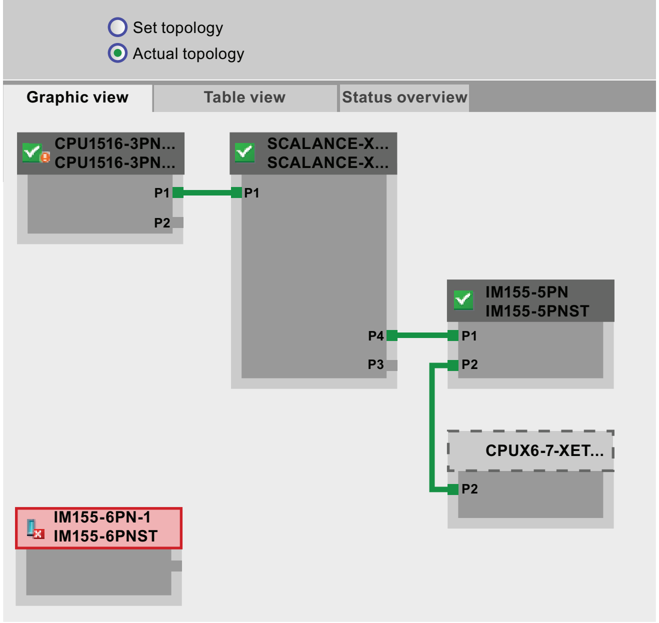


● Set topology
○ Actual topology

Figure 3-19 "Target topology" with failed device

"Actual topology" with failed device

Now switch to "Actual topology". The device that has failed in the meantime is indicated separately in the bottom area of this view. The failed device is displayed with a red border around the device header and the icon ☒.



○ Set topology
● Actual topology

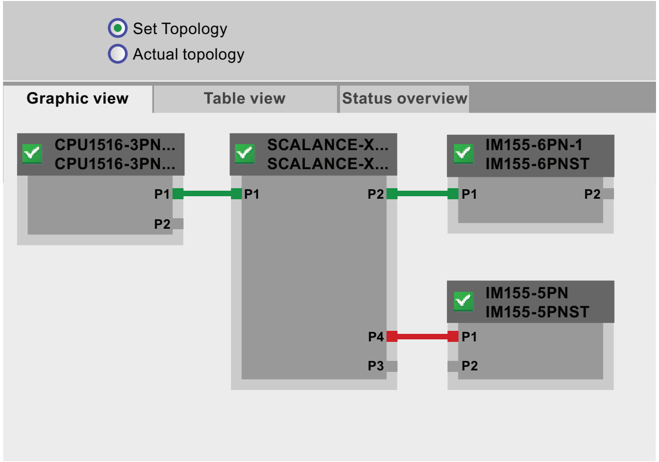
Figure 3-20 "Actual topology" with failed device

"Target topology" with interchanged ports

If a port was interchanged for a configured, directly adjacent PROFINET device, this device remains in the same place in the "Target topology" view. The interchanged connection is indicated by a red line.

The yellow line shows that a connection cannot be clearly determined:

- The connection to the IO devices that are not part of the CPU but can be connected in the target topology.
- Connections to third-party devices.



● Set Topology
○ Actual topology

Figure 3-21 "Target topology" with interchanged ports

3.8 User pages

User pages

In the "User pages" area of the Web server you can upload HTML pages you have created yourself for reading out data of the target system.

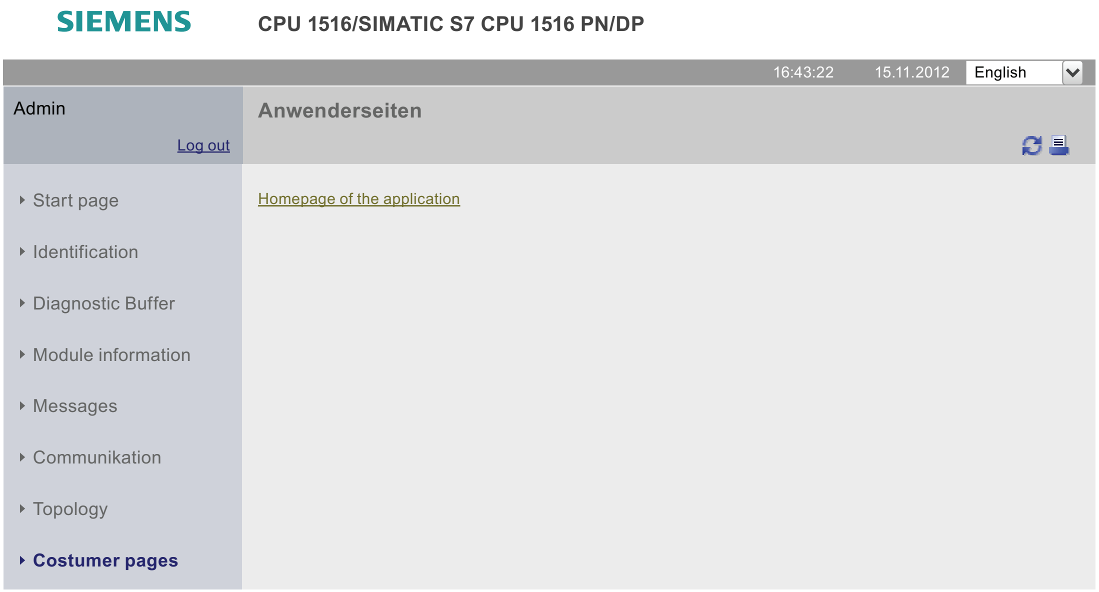


Figure 3-22 User pages

You create the pages with an HTML editor of your choice from which you generate data blocks (Web control DB and Fragment DBs) in STEP 7 and download them to the CPU. The system function SFC 99 "WWW" synchronizes the user program with the Web server on the CPU and initializes the Web server. With the first call of the SFC 99 "WWW", the link to the user page is displayed on the website of the CPU. A click on the link opens the user page in a new window.

Requirements

- You have assigned symbolic names to the tags you want to use on your website in STEP 7 .
- In the Inspector window under "Properties > General > Web server", you have at least
  - activated the Web server.
  - entered a user in the user list.
  - assigned read-only or read and write permissions to this (and additional) users (see chapter Properties of the Web server (Page 9)).
- You completed all necessary communication settings (IP address parameter, subnet mask, etc.).
- You have downloaded the configuration.
- You created your user page in an HTML editor of your choice:
  - Automatic HTML pages, if you want to **disable** control of the page layout by means of the user program (requires at least one call of SFC 99)
  - Manual HTML pages, if you want to **enable** control of the page layout by means of the user program (cyclical call of SFC 99 required)

Creating user pages

You can use any HTML editor to create your own user page(s). Make sure that your HTML code complies with the standards of the W3C (World Wide Web Consortium), because STEP 7 does not check the HTML syntax in any way.

Proceed as follows:

1. Create the HTML file for your user page with an HTML editor.
   To read out data from the S7-1500 with your HTML code, integrate the AWP commands as HTML comments.
2. Store the HTML file and all associated source files (e.g., *.gif, *.jpg, *.js, etc.) on your display device and write down the storage path.
3. Call the "WWW" instruction in STEP 7 and program it (see chapter Programming the WWW instruction (Page 56)).
4. Configure the user page in STEP 7 (see chapter Configuring user pages (Page 55)). This way you compile the contents of your HTML file, among other things, into data blocks in STEP 7.
5. Open your user page with your display device by means of a web browser in the Web server of the CPU.

Note

Comprehensive HTML pages, especially those with a lot of images, take up a lot of memory in the load memory. Make sure you select a SIMATIC memory card with sufficient memory to provide ample load memory.

Updating user pages

User pages are not updated automatically. You can program the HTML code so that the pages are updated automatically.

Pages which read out data from the controller, are always up-to-date due to regular updates.

Note

If the HTML page contains form fields for data input, automatic update can impair the correct data input by the user.

To update the entire page automatically, you can add the following instruction in the <head> area of your HTML page, whereby the number "10" stands for the update interval in seconds:

```
<meta http-equiv="refresh" content="10">
```

Reference

The description of a user page is available in the chapter Example of a user page (Page 58).

Additional help for visualization with user-defined websites is available in the application examples on the Internet:

- S7-1200 and STEP 7 V11 (http://support.automation.siemens.com/WW/view/en/58862931)
- S7-300 / S7-400 and STEP 7 as of V5.5 (http://support.automation.siemens.com/WW/view/en/44212999)

3.8.1 AWP commands

Overview

Automation Web Programming (AWP) commands are a special command syntax for data exchange between CPU and user page (HTML file).

AWP commands are entered as HTML comments and offer the following options for your user pages:

- Reading PLC tags
- Writing PLC tags
- Reading special tags
- Writing special tags
- Define enum types
- Assign enum types to tags
- Define data block fragments
- Import data block fragments

General syntax

All AWP commands, except for the command for reading a PLC tag, have this structure:
`<!-- AWP_< command name and parameter> -->`

Files including AWP commands:

- must be UTF-8 encoded.
  To define UTF-8 as character set of the page, add the following line to your HTML code:
  `<meta http-equiv="content-type" content="text/html; charset utf-8">`

> **Note**
>
> **Saving the HTML page**
>
> Make sure that you save the file in the editor in UTF 8 character encoding as well.

- may not include the following sequence:]]>
- may not include the following sequence outside "Reading tag areas" (:="<Varname>":): :=
- depending on the use, must identify special characters in tag names or data block names with character escape sequences or quotation marks
- are case-sensitive
- should be additionally enclosed by JavaScript comments ("/*...*/") in JavaScript files

Overview of AWP commands

Table 3- 5 AWP commands

Function	Representation
Reading PLC tags	`:=<Varname>:`
Writing PLC tags	`<!-- AWP_In_Variable Name='<Varname1>' -->`
Reading special tags	`<!-- AWP_Out_Variable Name='<Typ>:<Name>' -->`
Writing special tags	`<!-- AWP_In_Variable Name='<Typ>:<Name>' -->`
Define enum types	`<!-- AWP_Enum_Def Name='<Name Enum-Typ>' Values='0: "<Text_1>",1:"<Text_2>",...,x:"<Text_y>"' -->`
Assigning enum types to tags	`<!-- AWP_Enum_Ref Name='<Varname>' Enum='<Name Enum-Typ>' -->`
Define data block fragments	`<!--AWP_Start_Fragment Name='<Name>'[Type=<Typ>] [ID=<Id>] -->`
Import data block fragments	`<!--AWP_Import_Fragment Name='<Name>' -->`

3.8.1.1 PLC tags

PLC tags

User pages can read PLC tags from the CPU and write data to the CPU.

To do so, PLC tags must:

- be enclosed by double quotation marks ("...").
- also be enclosed by single quotation marks ('" ... "') or with quotation marks masked with a backslash ("\" ... "\").
- be specified by a PLC tag name.
- identify, if the PLC tag name includes the characters \ (backslash) or ', these characters with the escape sequence \\ or \' as normal characters of the PLC tag name.
- be enclosed with single quotation marks ('...'), if an absolute address (input, output, bit memory) is used in the AWP command.

Reading PLC tags

These out tags (output direction as seen from the controller) are inserted at any place in the HTML text with the syntax described below.

Syntax

```
:=<Varname>:
```

<Varname> corresponds to the tag to be read from your STEP 7 project and can be a simple shared tag or a complete tag path to a structure element. Make sure that you use the name of the data block and not its number when you use data blocks.

Examples

```
:="Conveying speed": :="My_datablock".bitmemory1:
:=I0.0:
:=MW100:
```

Writing PLC tags

These in-tags (input direction as seen from the controller) are set on the browser page. This can take place in a form on your HTML page, for example, with text input or list selection boxes that correspond to the tags that can be written.

The tags are either set in the HTTP Header (per cookie or POST method) or in the URL (GET method) by the browser in the HTTP request and are then written by the Web server into the respective PLC tag.

Note

Write access during operation

For data to be written from a user page to the CPU, a user must have the required write permissions and be logged on as this user. This applies to all write accesses of websites to the CPU.

Syntax

```
<!-- AWP_In_Variable Name='"<Varname1>"' Name='"<Varname2>"'
Name='"<Varname3>"' -->
```

If the name of the tag that you are using for the web application is not identical with the name of the PLC tag, you can assign it to a PLC tag with the "Use" parameter.

```
<!-- AWP_In_Variable Name='<Varname_Webapp>' Use='<PLC_Varname>' -->
```

Examples with HTML input boxes

```
<!--AWP_In_Variable Name='"Target_Level"' -->
<form method="post">
<p>Input Target Level: <input name='"Target_Level"'
type="text"><input type="submit" value="Write to PLC"> </p>
</form>

<!--AWP_In_Variable Name='"Data_block_1".Braking' -->
<form method="post">
<p>Braking: <input name='"Data_block_1".Braking' type="text"> <input
type="submit" value="Write to PLC"></p>
</form>
```

Examples with HTML drop-down list

```
<!-- AWP_In_Variable Name='"Data_block_1".ManualOverrideEnable' -->
<form method="post">
<select name='"Data_block_1".ManualOverrideEnable'>
<option value=1>Yes</option>
<option value=0>No</option>
</select><input type="submit" value="submit setting"> </form>
```

3.8.1.2 Special tags

Special tags

Special tags are mainly the HTTP tags specified in the definitions of the World Wide Web Consortium (W3C). Special tags are also used for cookies and server tags.

Reading special tags

The Web server can read PLC tags and transfer them to special tags in the HTTP response header. You can, for example, read out a path name from a PLC tag to redirect the URL to another storage location with the special tag "HEADER:Storage location".

Syntax

```
<!--AWP_Out_Varible Name='<Type>:<Name>' Use='<Varname>' -->
```

<Type> corresponds to the type of special tag.
Options are:

- HEADER
- COOKIE_VALUE
- COOKIE_EXPIRES

<Name> corresponds to the name of the HEADER tag or the cookie:

- HEADER tags:
  - Status: HTTP status code (if no other value was set, status code 302 is returned).
  - Location: path for redirection to another site. Status code 302 must be set.
  - Retry-After: time for which the service is most likely not available. Status code 503 must be set.
- COOKIE_VALUE:name: value of the named cookie.
- COOKIE_EXPIRES:name: expiration time of the named cookie in seconds.

Examples

The HTTP HEADER tag is written to the PLC tag of the same name:

```
<!-- AWP_Out_Variable Name='"HEADER:Status"' -->
```

If the name of the special tag is not identical with the name of the PLC tag, you can assign it to a PLC tag with the "Use" parameter:

```
<!-- AWP_Out_Variable Name='"HEADER:Status"' Use='"Status"' -->
```

Writing special tags

The Web server gives you the option to write values of special tags written in the HTTP header in the CPU. You can, for example, save information on the cookie of a user page or on the user that accesses a site in STEP 7.

Syntax

```
<!-- AWP_In_Variable Name='<Type>:<Name>' Use='Varname' -->
```

<Type> corresponds to the type of special tag.
Options are:

- HEADER
- SERVER
- COOKIE_VALUE

<Name> corresponds to the name of the HEADER tag or the cookie:

- HEADER tags:
  - Accept-Language: accepted or preferred language
  - User-Agent: information on the browser
  - Authorization: credentials for a requested resource
- SERVER tags:
  - current_user_id: shows if a user is logged on (current_user_id=0: no user logged on)
  - current_user_name: user name of the logged-on user
- COOKIE_VALUE:name: value of the named cookie.

Examples

The HTTP-SERVER tag is written to the PLC tag of the same name:

```
<!--AWP_In_Variable Name='"SERVER:current_user_id"' -->
```

The HTTP-SERVER tag is written to the PLC tag "My_userID":

```
<!--AWP_In_Variable Name='"SERVER:current_user_id"'
Use='"My_userID"'-->
```

3.8.1.3 Enum types

Enumeration types (enum types)

Enum types convert numerical values from the PLC program into texts or vice versa. The numerical values may also be assigned for use with several languages.

Define enum types

You can define enum types in your user pages and assign the values in an AWP command.

Syntax

```
<!-- AWP_Enum_Def_Name='<Name Enum-Typ>' Values='0:"<Text_1>",
1:"<Text_2>",...,x:"<Text_y>"' -->
```

Examples

To store German values as HTML file in the "de" folder of the HTML directory:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"Fault"' -
->
```

To store English values as HTML file in the "en" folder of the HTML directory:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -
->
```

Assigning enum types to tags

The assignment of tags from the user program to the individual enum types takes place by means of a separate AWP command. The used tag can be used at a different location of the user pages in a read operation or in a write operation.

For a read operation, the Web server replaces the value read from the CPU with the enum text value defined correspondingly. For a write operation, the Web server replaces the defined enum text value with the corresponding integer value of the enumeration before the value is written to the CPU.

Syntax

```
<!-- AWP_Enum_Ref_Name='<Varname>' Enum="<Enum-Type>" -->
```

<Varname> is the symbolic tag name from the user program; <Enum-Type> the previously specified name of the enum type.

Example for a declaration

```
<!-- AWP_Enum_Ref_Name='"Alarm"' Enum="AlarmEnum" -->
```

Example for use when reading a tag

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms",
1:"Container is full", 2:"Container is empty"' -->
<!-- AWP_Enum_Ref Name='"Alarm"' Enum="AlarmEnum" -->
...
<p> The current value of "Alarm" is :="Alarm": </p>
```

If the value of "Alarm" in the CPU is 2, the HTML page shows 'The current value of "Alarm" is container is empty' because the definition of the enum type assigns the numerical value 2 to the character sequence "Container is empty".

Example for use when writing a tag

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms",
1:"Container is full", 2:"Container is empty"' -->
<!-- AWP_In_Variable_Name='"Alarm"' -->
<!-- AWP_Enum_Ref Name='"Alarm"' Enum="AlarmEnum" -->
...
<form method="post">
<p><input type="hidden" name ='"Alarm"' value='Container is full'
/></p>
<p><input type="submit" value='Set container is full' /></p>
</form>
```

The value 1 is written to the PLC tag "Alarm" because the definition of the enum type assigns the numerical value 1 the text "Container is full".

Keep in mind that the name specified in "AWP_In_Variable" must be exactly the same in "AWP_Enum_Ref".

3.8.1.4 Fragments

Fragments

Fragments are "short paragraphs" of a website that the CPU processes separately.

Fragments are usually entire pages, but they can be individual elements, such as files (e.g. images) or documents.

Defining fragments

A fragment extends to the beginning of the next fragment or to the end of the file.

Syntax

```
<!-- AWP_Start_Fragment_Name='<Name>' [Type="<Typ>"] [ID="<Id>"] -->
```

This command specifies the start of a fragment.

- <Name> Specifies the name of the fragment. The name must start with a letter [a-zA-Z] or an underscore (_). This first character can be followed by letters, underscores or numbers [0-9].
- <Type> Specifies the type of the fragment.
  - "manual": The user program is notified of the request for a fragment and can react accordingly. The function of the fragment must be controlled with STEP 7 and the tags of the control DB.
  - "automatic": The page is automatically processed (default)

- <Id> You can specify a numerical ID for the fragment. If no ID is assigned, the fragment is automatically assigned an ID. For manual pages (<Type>=manual), the fragment can be addressed with this ID in the user program of the CPU.

  Note

  ID assignment

  Set the ID as low as possible because the highest ID influences the size of the Web control DB.

- <Modus>
  - "visible": The contents of the fragment are displayed on the user page (default).
  - "hidden": The contents of the fragment are **not** displayed on the user page.

Importing fragments

You can specify a fragment in an HTML page and import this fragment into other websites.

Example

A company logo is to be displayed on all websites of a web application.

The HTML code for the fragment that displays the company logo exists only once. You can import the fragment as often and into as many HTML files as necessary.

Syntax

```
<!-- AWP_Import_Fragment_Name='<Name>' -->
```

<Name> corresponds to the name of the fragment to be imported.

Example

HTML code within a website that creates a fragment for displaying an image:

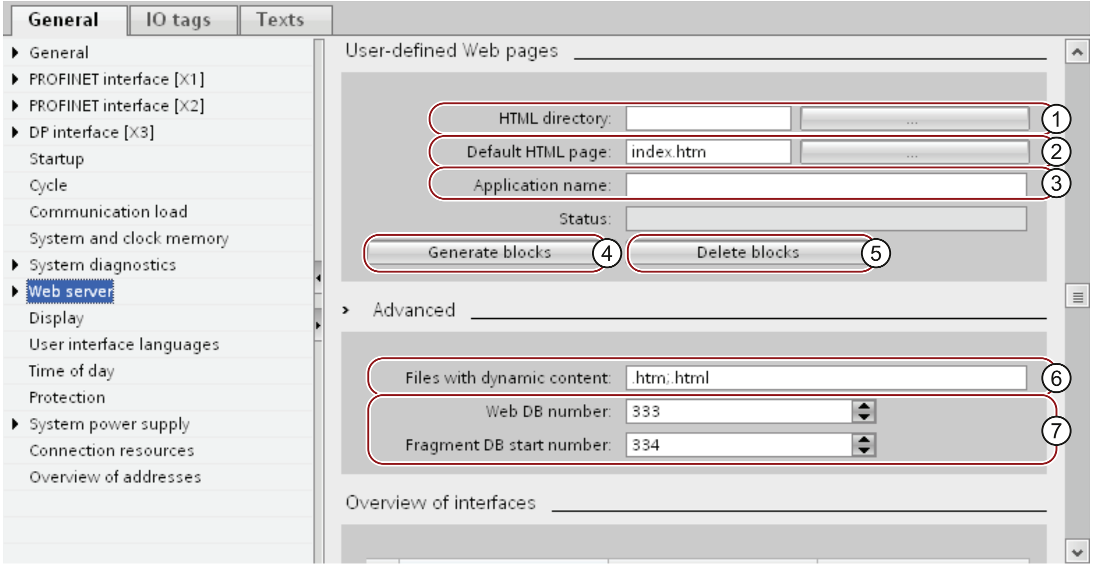
```
<!-- AWP_Start_Fragment Name='My_company_logo' -->
<p><img src="Company logo.jpg"></p>
```

HTML code that imports the created fragment into another website:

```
<!-- AWP_Import_Fragment Name='My_company_logo' -->
```

3.8.2 Configuring user pages

Configuring user pages



HTML directory:

Default HTML page: index.htm

Application name:

Status:

Files with dynamic content: .htm;.html

Web DB number: 333

Fragment DB start number: 334

Figure 3-23 Configuring user pages in STEP 7

Proceed as follows to configure the user pages in STEP 7:

- Select the CPU in the device configuration.
- Open the settings in the Inspector window of the CPU under "Properties > General > Web server".
- In the area "User pages" under ① "HTML directory", select the folder on your display device in which you have saved your HTML page.
- Enter the name of the HTML page under ② "Start HTML page" that is to open when you start the application.
- You can also specify a name for your application under ③ "Application name". This name is used to further divide or group the websites. If an application name already exists, the URL is displayed in the following format: http://a.b.c.d/awp/<application name>/<page name>.html
- In the "Extended" area, enter the file extensions that have to be checked for AWP commands in input box ⑥ "Files with dynamic contents". By default, STEP 7 analyses files with the extensions ".js", ".htm" and ".html". Here you can enter other file extensions that you have used when creating your user page.
- You can accept the number for the Web DB ⑦ and the fragment DB start number ⑦ or you can assign a new number of your choice that is not assigned.
- Click the button ④ "Create blocks" to create data blocks from the source files. The created data blocks are stored in the STEP 7 project tree in the folder "System blocks > Web server". These data blocks consist of a control data block (Web control DB) that controls the display of the websites and one or several data block fragments (fragment DBs) with the compiled websites.

- In the network view, select the CPU you want to download and select the command "Download to device" in the "Online" menu to download the blocks. The compilation of the blocks is implicitly triggered before the download. If errors are signaled during this process, they must be remedied before you can download the configuration.

Deleting data blocks

Click the "Delete block" button ⑤ to delete previously generated data blocks. STEP 7 Deletes the Web-Control DB and all fragment DBs from the project containing your user pages.

3.8.3 Programming the WWW instruction

The WWW instruction

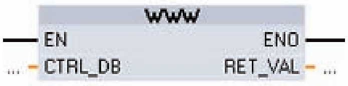
The WWW instruction initializes the Web server of the CPU or synchronizes the user pages with the user program in the CPU. The Web control DB is the input parameter for the WWW instruction and specifies the content of the pages as they are displayed in the fragment DBs as well as the status and control information. STEP 7 creates the Web control DB when you click the button "Create blocks".

Programming the WWW instruction

STEP 7 must execute the WWW instruction so that the user pages can be accessed in the Web server.

Table 3- 6 WWW instruction

LAD/FBD	SCL	Description

	`ret_val`
	`:=WWW(ctrl_db:=uint_in_)`
	`;`
		Access to user pages by means of the Web server

Parameters

The table below shows the parameters of the WWW instruction.

Table 3- 7 Parameters

Parameters	Declaration	Data type	Description
CTRL_DB	Input	DB_WWW	Data block that describes the user pages (Web control DB)
RET_VAL	Output	INT	Error information

RET_VAL parameter

Table 3- 8 RET_VAL

Error code (W#16#...)	Explanation
0000	No error has occurred. There are no pending website requests that must be released by the user program.

00xy	x: indicates if an error occurred during initialization of the Web control DB (CTRL_DB):
	x=0: no errors.
	x=1: errors occurred. The error is encoded in the "CTRL_DB.last_error" byte of the Web control DB.
	y: Number of the pending request. Several requests are possible (e.g., request "0" and "1" are pending: y="3").
	y="1": Request "0"
	y="2": Request "1"
	y="4": Request "2"
	y="8": Request "3"

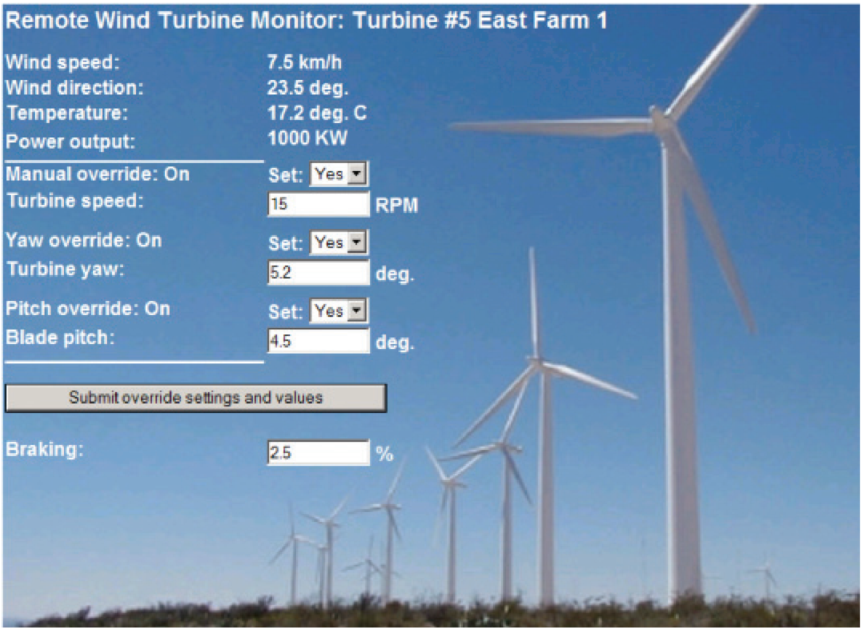
803A	The specified Web Control DB does not exist on the CPU.
8081	Incorrect version or incorrect format of the Web Control DB.
80C1	There are no resources available to initialize the Web application.

3.8.4 Example of a user page

3.8.4.1 Website for monitoring and controlling a wind turbine

Example of a user page

Here you see a user page for monitoring and controlling a wind turbine:



Remote Wind Turbine Monitor: Turbine #5 East Farm 1

Wind speed: 7.5 km/h
Wind direction: 23.5 deg.
Temperature: 17.2 deg. C
Power output: 1000 KW

Manual override: On — Set: Yes
Turbine speed: 15 RPM

Yaw override: On — Set: Yes
Turbine yaw: 5.2 deg.

Pitch override: On — Set: Yes
Blade pitch: 4.5 deg.

Braking: 2.5 %

Figure 3-24 Overview of user page wind turbine

The user page was created in English in this example, but you can select any language you wish when you create your own user page.

In this application, each wind turbine of the wind farm has a data block in STEP 7 with specific data for the respective location and the turbine.

The user page gives you the option to access the turbine remotely with a display device. A user can open the standard websites for a CPU of a specific wind turbine and go to the "Remote Wind Turbine Monitor" user page to view the turbine data. A user with the corresponding access permissions can also set the turbine into the manually controlled mode and thus control the tags for speed, orientation and angle of attack of the turbine by means of the website. The user can also specify a brake value regardless of manual or automatic control of the turbine.

STEP 7 checks the Boolean values for override of the automatic control and, if set, uses the values for speed, orientation and angle of attack of the turbine as defined by the user.

Files used

Three files are used in the application example:

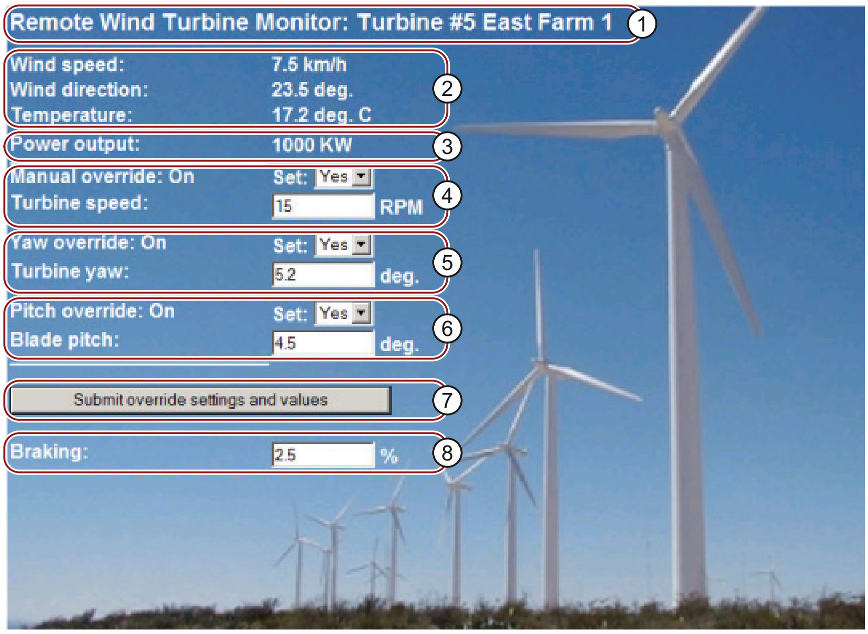
- Wind_turbine.html: The user page in the figure shown above. The control data is accessed by AWP commands.
- Wind_turbine.css: The Cascading Style Sheet which includes the formatting specifications of the user page. The use is optional but can simplify the design of the user page.
- Wind_turbine.jpg: The background image displayed on the user page. The use of images is optional, user pages with lots of images require a lot more memory in the load memory.

These files are not part of your installation but they are described as an example below.

Implementation

The user page uses AWP commands to read out values from the CPU as well as writing values to it. The user page also uses AWP commands for the definition of enum types, such as the assignment of tags to enum types for handling the ON/OFF settings.

The user page is structured as follows:



Remote Wind Turbine Monitor: Turbine #5 East Farm 1

Wind speed: 7.5 km/h
Wind direction: 23.5 deg.
Temperature: 17.2 deg. C

Power output: 1000 KW

Manual override: On Set: Yes
Turbine speed: 15 RPM

Yaw override: On Set: Yes
Turbine yaw: 5.2 deg.

Pitch override: On Set: Yes
Blade pitch: 4.5 deg.

Braking: 2.5 %

① Header of the website with number and location of the wind turbine.

② Atmospheric conditions at the turbine, wind speed, wind direction and current temperature are displayed.

③ Read-out power output.

④ Manual override: Activates manual override of the turbine. To make manual settings for speed, orientation and angle of attack, the STEP 7 user program requires that manual override has been activated.

⑤ Override of the orientation: Activates manual override of the turbine orientation.

⑥ Override of the angle of attack: Activates manual override of the angle of attack of the rotor blades.

⑦ By clicking this button, you transfer the override settings to the CPU.

⑧ Manual setting of a percentage value for braking. The setting "Manual override" is not required to enter a brake value.

Figure 3-25 Overview of user page wind turbine

The user page also uses an AWP command which writes the special tag into the tag table; it includes the user ID of the user who currently accesses the page.

3.8.4.2 Reading and displaying data from the CPU

Example HTML code for reading out and displaying data from the CPU

This part of the HTML code is used for displaying the power output on the user side.

The text "Power Output:" is displayed on the left-hand side; on the right-hand side, the value of the tags for the power output including the unit ("KW") is displayed.

The AWP command `:="Data_block_1".PowerOutput` executes the reading operation. The data block is referenced by its symbolic name here and not by its number ("Data_block_1" instead of "DB1").

The code used in the example is:

```
<tr style="height:2%;">
<td>
<p>Power output:</p>
</td>
<td>
<p style="margin-bottom:5px;"> :="Data_block_1".PowerOutput: KW</p>
</td>
</tr>
```

3.8.4.3 Using enum types

Definition of enum types

The described user page uses enum types in three locations. "On" or "Off" is displayed for a Boolean value at these locations.

The enum type for "On" results in a value of 1; the enum type for "Off" results in a value of 0. The following excerpts from the HTML code of the user page show the declaration of an enum type with the name "OverrideStatus" and the values "0" and "1" for "Off" or "On" as well as the specification of an enum type reference of "OverrideStatus" for the tag "ManualOverrideEnable" in the data block "Data_block_1".

Note

Assignment of enum types

If the user page writes into a tag by using an enum type, there has to be a declaration "AWP_In_Variable" for each "AWP_Enum_Ref" declaration.

The code used in the example is:

```
<!-- AWP_In_Variable_Name='"Data_block_1".ManualOverrideEnable' -->
<!-- AWP_Enum_Def_Name="OverrideStatus" Values='0: "Off",1:"On"' -->
<!-- AWP_Enum_Ref_Name='"Data_block_1".ManualOverrideEnable'
Enum="OverrideStatus" -->
```

The following code describes a display box for displaying the current status of "ManualOverrideEnable". A normal read command for tags is used but because of the declared and referenced enum type, the website displays the values "On" and "Off" instead of "1" and "0".

```
<td style="width:24%; border-top-style: Solid; border-top-width:
2px; border-top-color: #ffffff;">
<p>Manual override: :="Data_block_1".ManualOverrideEnable:</p>
</td>
```

The following code describes a drop-down list for changing "ManualOverrideEnable" by the user. The drop-down list consists of the "Yes" and "No" options that are assigned to the "On" or "Off" values by means of the enum type reference. If you make no selection, the status remains the same.

```
<select name='"Data_block_1"ManualOverrideEnable'>
<option value=':"Data_block_1".ManualOverrideEnable:'> </option>
<option value="On">Yes</option>
<option selected value="Off">No</option>
</select>
```

The drop-down list is included in the form on the website. The form is uploaded, when the user clicks on the "Send" button. If the user has selected "Yes", the value "1" is written in the tag "ManualOverrideEnable" in the "Data_block_1" data block; if the user has selected "No", the value "0" is written.

3.8.4.4 Writing user inputs into the controller

Setting options

The user page "Remote Wind Turbine Monitor" includes different AWP commands for writing data into the controller. A user with the corresponding access permissions can control the wind turbine manually, activate the override for the turbine speed and the turbine orientation as well as the angle of attack of the rotor blades with the declaration of different "AWP_In_Variable" write commands. The user can also specify floating-point numbers for turbine speed, orientation angle of attack and percentage of braking. The user page uses an HTTP command in the format "POST" to write the tags into the controller.

The code used in the example for setting the brake value is:

```
<!-- AWP_In_Variable Name='"Data_block_1"' -->
...
<tr sytle="vertical-align: top; height: 2%;">
<td style="width: 22%;"><p>Braking:</p></td>
<td>
<form method="POST">
<p><input name='"Data_block_1".Braking' size="10" type ="text">
%</p>
</form>
</td>
</tr>
```

This excerpt from the HTML code first defines a "AWP_In_Variable" for the "Data_block_1" data block which enables the user page to write any number of tags into the data block. The text "Breaking:" is displayed on the left-hand side; a box is available on the right-hand side in which the user can make inputs for the tag "Braking" in the data block.

The user page reads out the actual braking value from the controller and displays it in the text box. A user with the corresponding access permissions can then write a brake value that controls the braking process into the data block of the CPU.

Note

Declaration of data blocks

If you declare an entire data block by means of a "AWP_In_Variable", each tag in the data block can be written by means of the user page. If you want only specific tags in the data block to be writable, you have to declare it specifically using `<!-- AWP_In_Variable Name='"Data_block_1".Breaking'>`, for example.

3.8.4.5 Writing special tags

Using special tags

The user page "Remote Wind Turbine Monitor" writes the special tag "Server:current_user_id" into a tag of the CPU. The tag value contains the value "1" if a user is logged on and "0" if a user is not logged on. In this example, a user is logged on, so the tag value is set to "1". The special tag is written into the CPU by the user page and does not need a user interface.

The code used in the example is:

```
<!-- AWP_in_variable Name="SERVER:current_user_id" Use="User_ID" -->
```

3.8.4.6 HTML code of the user page "Remote Wind Turbine Monitor"

The complete HTML code of the example user page "Remote Wind Turbine Monitor" as well as the used Cascading Style Sheet (CSS) is listed below.

Wind_turbine.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd"><!--
This test program simulates a website for monitoring and operating a
wind turbine.
Required PLC tags and data block tags in STEP 7:
PLC tag:
User_ID: Int
Data blocks:
Data_block_1
Tags in Data_Block_1:
Location: StringTurbineNumber: IntWindSpeed: RealWindDirection:
RealTemperature: RealPowerOutput: RealManualOverrideEnable:
BoolTurbineSpeed: RealYawOverride: BoolYaw: RealPitchOverride:
BoolPitch: RealBraking: Real
The user-defined website shows the current values for the PLC data
and offers a drop-down list to specify the three Boolean values with
the assigned enumeration type. The selected Boolean values as well
as the data text boxes for speed, orientation and angle of attack of
the turbine are uploaded with the "Send" button. The brake value can
be specified without the "Send" button.

No actual STEP 7 program is required for using this page. The STEP 7
program would theoretically only respond to the values for speed,
orientation and angle of attack of the turbine, if the assigned
Boolean values were specified. The only requirement for STEP 7 is
that the WWW instruction is called with the DB number of the
generated data blocks for this page.
-->
<!-- AWP_In_Variable Name='"Data_block_1"' -->
<!-- AWP_In_Variable Name='"Data_block_1".ManualOverrideEnable' -->
<!-- AWP_In_Variable Name='"Data_block_1".PitchOverride' -->
<!-- AWP_In_Variable Name='"Data_block_1".YawOverride' -->
<!-- AWP_In_Variable Name="SERVER:current_user_id" Use="User_ID"-->
<!-- AWP_Enum_Def Name="OverrideStatus" Values='0:"Off",1:"On"' -->
<!-- AWP_Enum_Ref Name='"Data_block_1".ManualOverrideEnable'
Enum="OverrideStatus" --><!-- AWP_Enum_Ref
Name='"Data_block_1".PitchOverride' Enum="OverrideStatus" --><!--
AWP_Enum_Ref Name='"Data_block_1".YawOverride' Enum="OverrideStatus"
-->
<html>
 <head>
 <meta http-equiv="content-type" content="text/html; charset=utf-8">
    <link rel="stylesheet" href="Wind_turbine.css" type="text/css">
    <title>Remote monitoring of wind turbines
    </title>
 </head>
```

```
<body>
    <form method="post" action="">
      <table style=
      "background-image: url('./Wind_turbine.jpg'); width: 904px; height: 534px;"
      cellpadding="0" cellspacing="2">
        <tr style="height: 2%;">
          <td colspan="2">
            <h2>
              Remote Wind Turbine Monitor: Turbine
              #:="Data_block_1".TurbineNumber:
              :="Data_block_1".Location:
            </h2>
          </td>
        </tr>
        <tr style="height: 2%;">
          <td style="width: 24%;">
              Wind speed:
          </td>
          <td>
              :="Data_block_1".WindSpeed: km/h
          </td>
        </tr>
        <tr style="height: 2%;">
          <td style="width: 24%;">
              Wind direction:
          </td>
          <td>
              :="Data_block_1".WindDirection: deg.
          </td>
        </tr>
        <tr style="height: 2%;">
          <td style="width: 24%;">
              Temperature:
          </td>
          <td>
              :="Data_block_1".Temperature: deg. C
          </td>
        </tr>
        <tr style="height: 2%;">
          <td style="width: 24%;">
              Power output:
          </td>
          <td>
            <p style="margin-bottom:5px;">
              :="Data_block_1".PowerOutput: KW
          </td>
        </tr>
        <tr style="height: 2%;">
          <td style=
          "width=24%; border-top-style: Solid; border-top-width: 2px; border-top-
color: #ffffff;">
          <p>
              Manual override:
              :="Data_block_1".ManualOverrideEnable:
          </td>
          <td class="Text">
            Set: <select name=
            '"Data_block_1".ManualOverrideEnable'>
              <option value="On">Yes</option>
              <option value="Off">No</option>
            </select>
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width: 24%;">
              Turbine speed:
          </td>
          <td>
            <p style="margin-bottom:5px;">
              <input name='"Data_block_1".TurbineSpeed' size="10"
              value=':="Data_block_1".TurbineSpeed:' type="text">
              RPM
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width: 24%;">
              Yaw override: :="Data_block_1".YawOverride:
          </td>
          <td class="Text">
            Set: <select name='"Data_block_1".YawOverride'>
              <option value="On">Yes</option>
              <option value="Off">No</option>
            </select>
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width: 24%;">
              Turbine yaw:
          </td>
          <td>
            <p style="margin-bottom:5px;">
              <input name='"Data_block_1".Yaw' size="10" value=
              ':="Data_block_1".Yaw:' type="text"> deg.
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width: 24%;">
              Pitch override: :="Data_block_1".PitchOverride:
          </td>
          <td class="Text">
            Set: <select name='"Data_block_1".PitchOverride'>
              <option value="On">Yes</option>
              <option value="Off">No</option>
            </select>
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width=24%; border-bottom-style: Solid; border-bottomwidth:
2px; border-bottom-color: #ffffff;">
          <p>
```

```
              Blade pitch:
          </td>
          <td>
            <p style="margin-bottom:5px;">
              <input name='"Data_block_1".Pitch' size="10" value=
              ':="Data_block_1".Pitch:' type="text"> deg.
          </td>
        </tr>
        <tr style="height: 2%;">
          <td colspan="2">
            <br>
            <input type="submit" value=
            "Submit override settings and values">
          </td>
        </tr>
        <tr style="vertical-align: top; height: 2%;">
          <td style="width: 24%;">
              Braking:
          </td>
          <td>
              <input name='"Data_block_1".Braking' size="10" value=
              ':="Data_block_1".Braking:' type="text"> %
          </td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

Wind_turbine.css

```
              H2 {
                  font-family: Arial;
                  font-weight: bold;
                  font-size: 14.0pt;
                  color: #FFFFFF;
                  margin-top:0px;
                  margin-bottom:10px;
              }
              P {
                  font-family: Arial;
                  font-weight: bold;
                  color: #FFFFFF;
                  font-size: 12.0pt;
                  margin-top:0px;
                  margin-bottom:0px;
              }
              TD.Text {
                  font-family: Arial;
                  font-weight: bold;
                  color: #FFFFFF;
                  font-size: 12.0pt;
                  margin-top:0px;
                  margin-bottom:0px;
              }
```

3.9 File browser

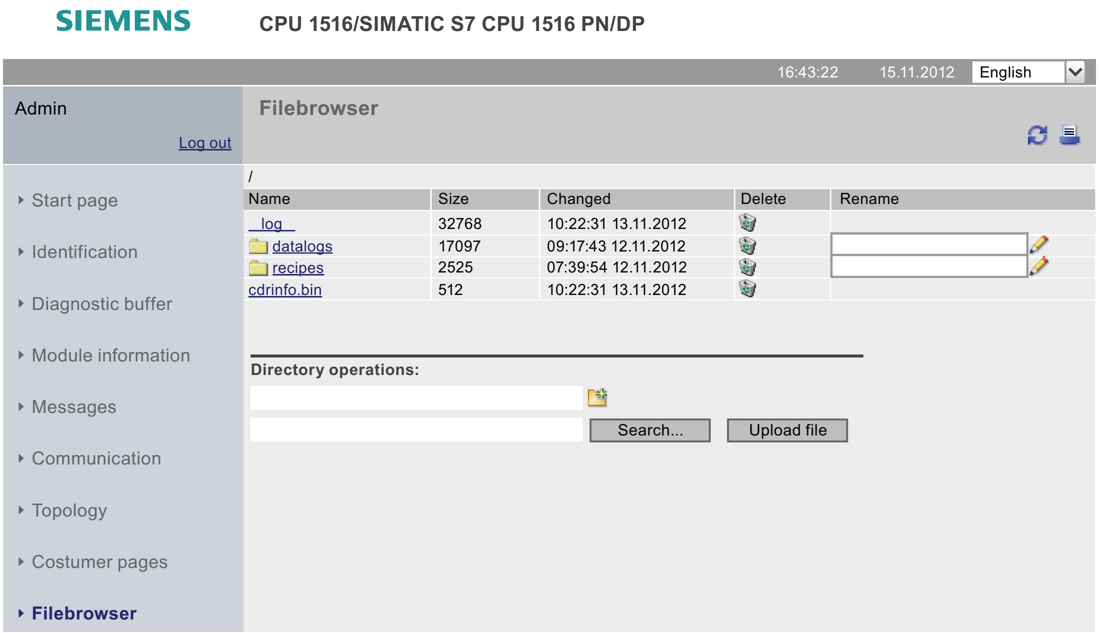
Requirements

For error-free operation, the following conditions must be met:

- The SIMATIC memory card is inserted into the CPU.
- Execute permissions must be assigned for the user in the user administration.

File browser

The browser displays the content of the SIMATIC memory card on the "File browser" website. This means, for example, that you can read and edit the log files generated by the CPU without having to run STEP 7.



Name	Size	Changed	Delete	Rename
log	32768	10:22:31 13.11.2012		
datalogs	17097	09:17:43 12.11.2012		
recipes	2525	07:39:54 12.11.2012		
cdrinfo.bin	512	10:22:31 13.11.2012		

Figure 3-26 File browser view

The file browser lists all existing files and directories on the SIMATIC memory card. Depending on the file type, the data can be downloaded, deleted, renamed or uploaded. The directories can only be created, deleted or renamed.

Exception system files

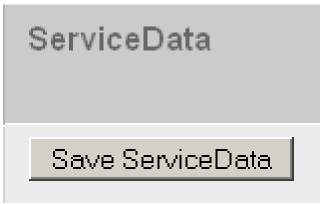
The system files include the job file and all special directories including their contents to which the job file refers. System files are not displayed, which means they cannot be changed or deleted.

3.10 Reading out service data

The Web server gives you the option to save service data. In addition to the content of the diagnostic buffer, they include additional information on the internal status of the CPU. If you should encounter a problem with the CPU that cannot be resolved otherwise, you therefore have the option to submit the service data to the Service&Support team.

Procedure

1. Enter the following address in the address bar of your web browser:
   "http://<CPU IP address>/save_service_data.html", e.g.,
   "http://192.168.3.141/save_service_data.html"
2. Your screen displays the service data page with a button for saving the service data.



3. Save the service data locally on your display device by clicking "Save ServiceData".

Result

The data is saved into a .dmp file with the following naming convention: "<MLFB><serial number><time stamp>.dmp". The user can change the file name at a later time.

Glossary

Automation system

An automation system is a programmable logic controller that consists of at least one CPU, various input and output modules, as well as operating and monitoring devices.

AWP

Automation Web Programming

AWP commands

Special command syntax for data exchange between CPU and HTML file.

Configuration

Systematic arrangement of individual modules (design).

CSS

A CSS (Cascading Style Sheet) specifies how an area or content marked up in HTML is displayed.

Diagnostics

The detection, localization, classification, visualization and further evaluation of errors, malfunction and alarms.

Diagnostics provides monitoring functions that run automatically during plant operation. This increases the availability of plants by reducing commissioning times and downtimes.

Firewall

A firewall is used to restrict the network access based on sender or target address of the used services. The firewall decides based on specified rules which of the network packets it handles are forwarded and which are not. This way the firewall tries to prevent unauthorized network access.

It is not the function of a firewall to detect attacks. It only implements rules for network communication.

HTTP

Hypertext Transfer Protocol (HTTP). Protocol for data transmission across a network.

HTTPS

Hypertext Transfer Protocol Secure (HTTPS). Protocol for tap-proof transmission of sensitive data across a network.

Identification data

Identification data is stored on a module, and contains information which supports the user in

- Checking the system configuration
- Locating hardware changes in a system
- Correcting errors in a system

Modules can be clearly identified online using the identification data.

Master

The master in possession of the token is an active device. This master has the option to receive data from other devices and to send data to other devices.

PROFIBUS

PROcess **FIeld BUS**, process and field bus standard specified in standard IEC 61784-1:2002 Ed1 CP 3/1. It specifies functional, electrical, and mechanical properties for a bit-serial field bus system.

PROFIBUS is available with the protocols DP (= Distributed I/O), FMS (= Fieldbus Message Specification), PA (= Process Automation), or TF (= Technological Functions).

PROFINET

Within the framework of Totally Integrated Automation (TIA), PROFINET represents the consistent continuation of:

- PROFIBUS DP, the established field bus
- Industrial Ethernet, the communication bus for the cell level

Experience gained from both systems was and is being integrated into PROFINET.

PROFINET as an Ethernet-based automation standard from PROFIBUS International (previously PROFIBUS User Organization) defines a vendor-independent communication, automation, and engineering model.

PROFINET component

A PROFINET component includes the entire data of the hardware configuration, the parameters of the modules, and the corresponding user program. The PROFINET component is made up as follows:

- Technological Function

  The (optional) technological (software) function includes the interface to other PROFINET components in the form of interconnectable inputs and outputs.

- Device

  The device is the representation of the physical programmable controller or field device including the I/O, sensors and actuators, mechanical parts, and the device firmware.

PROFINET IO

As part of PROFINET, PROFINET IO is a communication concept that is used to implement modular, distributed applications.

PROFINET IO allows you to create automation solutions which are familiar to you from PROFIBUS.

PROFINET IO is implemented by the PROFINET standard for automation devices on the one hand, and on the other hand by the engineering tool STEP 7.

That is, you have the same application view in STEP 7, regardless of whether you configure PROFINET or PROFIBUS devices. Programming your user program is essentially the same for PROFINET IO and PROFIBUS DP if you use the extended blocks and system status lists for PROFINET IO.

PROFINET IO controller

Device used to address connected I/O devices. This means that the IO controller exchanges input and output signals with assigned field devices. The IO controller is often the controller on which the automation program runs.

PROFINET IO device

A distributed field device that is assigned to one of the IO controllers (e.g., remote IO, valve terminals, frequency converters, switches).

UTF-8

Abbreviation for 8-bit UCS (Universal Character Set) transformation format. Most popular coding of unicode characters.

Each unicode character is assigned a specially coded byte string of variable length in this format. UTF-8 supports up to four bytes on which all unicode characters can be mapped.

Web browser

Web browsers are visualization programs for websites and can communicate with Web servers.

Typical web browsers are, for example:

- Microsoft Internet Explorer
- Mozilla Firefox

Index