

Installing and starting up the AXC 1050 PN STARTERKIT

Quick start guide

Quick start guide

Installing and starting up the AXC 1050 PN STARTERKIT

2015-03-31

Designation: UM QS EN AXC 1050 PN STARTERKIT

Revision: 00

Order No.: —

This user manual is valid for:

| | | |
|------------------------|---------|-----------|
| Designation | Version | Order No. |
| AXC 1050 PN STARTERKIT | 1.0 | 2400361 |

Please observe the following notes

User group of this manual

The use of products described in this manual is oriented exclusively to:

- Qualified electricians or persons instructed by them, who are familiar with applicable standards and other regulations regarding electrical engineering and, in particular, the relevant safety concepts.
- Qualified application programmers and software engineers, who are familiar with the safety concepts of automation technology and applicable standards.

Explanation of symbols used and signal words



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety measures that follow this symbol to avoid possible injury or death.

There are three different categories of personal injury that are indicated with a signal word.

DANGER This indicates a hazardous situation which, if not avoided, will result in death or serious injury.

WARNING This indicates a hazardous situation which, if not avoided, could result in death or serious injury.

CAUTION This indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.



This symbol together with the signal word **NOTE** and the accompanying text alert the reader to a situation which may cause damage or malfunction to the device, hardware/software, or surrounding property.



This symbol and the accompanying text provide the reader with additional information or refer to detailed sources of information.

How to contact us

Internet

Up-to-date information on Phoenix Contact products and our Terms and Conditions can be found on the Internet at:

phoenixcontact.com

Make sure you always use the latest documentation.

It can be downloaded at:

phoenixcontact.net/products

Subsidiaries

If there are any problems that cannot be solved using the documentation, please contact your Phoenix Contact subsidiary.

Subsidiary contact information is available at phoenixcontact.com.

Published by

PHOENIX CONTACT GmbH & Co. KG
Flachsmarktstraße 8
32825 Blomberg
GERMANY

Should you have any suggestions or recommendations for improvement of the contents and layout of our manuals, please send your comments to:

tecdoc@phoenixcontact.com

Please observe the following notes

General terms and conditions of use for technical documentation

Phoenix Contact reserves the right to alter, correct, and/or improve the technical documentation and the products described in the technical documentation at its own discretion and without giving prior notice, insofar as this is reasonable for the user. The same applies to any technical changes that serve the purpose of technical progress.

The receipt of technical documentation (in particular user documentation) does not constitute any further duty on the part of Phoenix Contact to furnish information on modifications to products and/or technical documentation. You are responsible to verify the suitability and intended use of the products in your specific application, in particular with regard to observing the applicable standards and regulations. All information made available in the technical data is supplied without any accompanying guarantee, whether expressly mentioned, implied or tacitly assumed.

In general, the provisions of the current standard Terms and Conditions of Phoenix Contact apply exclusively, in particular as concerns any warranty liability.

This manual, including all illustrations contained herein, is copyright protected. Any changes to the contents or the publication of extracts of this document is prohibited.

Phoenix Contact reserves the right to register its own intellectual property rights for the product identifications of Phoenix Contact products that are used here. Registration of such intellectual property rights by third parties is prohibited.

Other product identifications may be afforded legal protection, even where they may not be indicated as such.

Table of contents

| | | |
|-------|---|----|
| 1 | General | 9 |
| 1.1 | Introduction..... | 9 |
| 1.2 | Information about this document..... | 9 |
| 1.3 | Components of the AXC 1050 PN STARTERKIT | 10 |
| 1.4 | System requirements..... | 11 |
| 1.4.1 | Operating systems supported | 11 |
| 1.4.2 | Hardware requirements | 11 |
| 1.4.3 | Required programming software | 11 |
| 1.4.4 | Required firmware versions | 11 |
| 2 | Installing the hardware | 13 |
| 3 | Installing the software | 17 |
| 3.1 | Starting the installation program of the AUTOMATIONWORX Software Suite | 17 |
| 3.2 | Starting PC Worx | 18 |
| 3.3 | Enabling the PC Worx license | 19 |
| 4 | Helpful information about PC Worx | 21 |
| 4.1 | Online help | 21 |
| 4.2 | Selecting the language | 21 |
| 4.3 | The PC Worx user interface..... | 21 |
| 4.4 | Toolbars | 22 |
| 4.5 | Workspaces..... | 23 |
| 4.6 | “Bus Structure” window..... | 26 |
| 4.6.1 | Icons in the “Bus Structure” window | 26 |
| 4.6.2 | Display in the “Bus Structure” window | 28 |
| 4.7 | Compiling and sending a project and performing a cold restart | 30 |
| 4.7.1 | Compiling a project | 30 |
| 4.7.2 | Downloading a project | 30 |
| 4.7.3 | Performing a cold restart | 32 |
| 4.7.4 | Compiling (additional information) | 33 |
| 5 | Example project | 35 |
| 5.1 | Overview | 35 |
| 5.2 | Creating a new project..... | 35 |
| 5.3 | Adapting the project information | 38 |
| 5.4 | Preparing the PC for communication | 39 |
| 5.5 | Checking/modifying IP settings for the PROFINET controller | 40 |
| 5.5.1 | Assigning the IP address for the PROFINET controller | 41 |
| 5.6 | Activating the PROFINET controller function | 46 |
| 5.7 | Reading in PROFINET devices | 47 |

| | | |
|----------|---|-----------|
| 5.8 | Checking/modifying the PROFINET settings for PROFINET devices | 51 |
| 5.9 | Transferring PROFINET device names and IP settings to PROFINET devices ... | 54 |
| 5.10 | Reading in Axioline devices | 55 |
| 5.11 | Compiling and sending a project, and performing a cold restart after reading the bus topology | 56 |
| 5.12 | Creating the program | 57 |
| 5.12.1 | “Moving_Light” program description | 57 |
| 5.12.2 | Extracting the “Moving_Light” function block library | 58 |
| 5.12.3 | Importing and compiling the “Moving_Light” function block library in PC Worx | 58 |
| 5.12.4 | Programming | 60 |
| 5.12.5 | Compiling after creating the program | 63 |
| 5.13 | Assigning process data | 63 |
| 5.14 | Compiling and sending a project (including program), and performing a cold restart..... | 65 |
| 5.15 | Operation and debug mode – Viewing values online | 66 |
| 5.16 | Example project on CD..... | 68 |
| 6 | Additional functions of PC Worx | 71 |
| 6.1 | Changing the PROFINET device name (“DNS Name”)..... | 71 |
| 6.2 | Renumbering devices..... | 74 |
| 6.3 | Diagnostics with Diag+ | 75 |
| 6.4 | Importing Ethernet devices into the project..... | 81 |
| 6.5 | Device description files..... | 84 |
| 6.5.1 | Phoenix Contact device description files (FDCML files) | 84 |
| 6.5.2 | Device description files from other manufacturers (GSD files) | 85 |
| 7 | Visualization with WebVisit | 87 |
| 7.1 | General..... | 87 |
| 7.2 | Saving/extracting the visualization..... | 88 |
| 7.3 | WebVisit: path to the PC Worx project..... | 89 |
| 7.4 | WebVisit: downloading the visualization to the controller..... | 90 |
| 7.5 | Starting the visualization..... | 92 |
| A | Status information and system variables..... | 95 |
| A 1 | Status of the PROFINET controller | 95 |
| A 2 | Status of a PROFINET device | 96 |
| A 3 | PNIO_FORCE_FAILSAFE system variable..... | 97 |

| | | |
|-----|-----------------------|----|
| B | Appendixes..... | 99 |
| B 1 | List of figures | 99 |

1 General

1.1 Introduction

The AXC 1050 PN STARTERKIT is a combination of hardware and software. It contains all the components you need in order to create and start up a basic PROFINET system.

1.2 Information about this document

Using an example project, this document helps you to install and parameterize a bus configuration and to program the application program (according to IEC 61131).

It is assumed the user has knowledge and experience in the operation of PCs, Windows® operating systems, and knowledge of IEC 61131.



For more detailed information about the hardware components, please refer to the documentation for the components. The documentation is also included on the CD supplied with the AXC 1050 PN STARTERKIT. In addition, the documentation can be downloaded at phoenixcontact.net/products.

More detailed information about the individual functions of PC Worx can be found in the online help for the program. The entire help function can be called via "Help" in the menu bar. Help for specific functions can be called via F1.



This quick start guide describes the quickest way to start up. Since the devices are read, a complete physical bus configuration is required.

No functions or commands that require communication with the control system can be executed without a physical bus configuration. However, complete parameterization is possible in the "offline" state. The application program can also be created and compiled. Should you wish to proceed in this way, please refer to the quick start guide for PC Worx.

1.3 Components of the AXC 1050 PN STARTERKIT

Table 1-1 Components of the AXC 1050 PN STARTERKIT

| Description | Type | Order No. | Pcs. / Pkt. |
|---|----------------------------|-----------|-------------|
| Software | | | |
| Demo DVD of the PC Worx automation software | PC WORX DEMO | 2985725 | 1 |
| License | PC WORX EDUCATION LIC | 2700205 | 1 |
| CD with example project/program, "Moving_Light" function block library, and "Installing and starting up the AXC 1050 PN STARTERKIT" quick start guide | CD AXC 1050 PN STARTERKIT | – | 1 |
| Hardware (module) | | | |
| Axioline F for the direct control of Axioline I/Os. With two Ethernet interfaces and programming options according to IEC 61131-3. Complete with connector plug and labeling field. | AXC 1050 | 2700988 | 1 |
| Axioline F bus coupler for PROFINET (including bus base module and connector) | AXL F BK PN | 2701815 | 1 |
| Axioline F digital input module, 16 inputs, 24 V DC, 2, 3, 4-wire connection technology (including bus base module and connectors) | AXL F DI16/4 2F | 2688022 | 1 |
| Axioline F digital output module, 16 outputs, 24 V DC, 500 mA, 2, 3-wire connection technology (including bus base module and connectors) | AXL F DO16/3 2F | 2688048 | 1 |
| Hardware (accessories, included in the starter kit) | | | |
| 24 V DC/1.75 A DIN rail power supply unit, primary-switched, single-phase | STEP-PS/ 1AC/24DC/1.75 | 2868648 | 1 |
| Patch cable, CAT5, pre-assembled, 1.0 m | FL CAT5 PATCH 1,0 | 2832276 | 1 |
| Connecting cable, for connecting the controller to a PC for PC Worx, USB A to micro USB B, 2 m in length | CAB-USB A/MICRO USB B/2,0M | 2701626 | 1 |

1.4 System requirements

1.4.1 Operating systems supported

- Windows® XP SP3
- Windows® Vista Business SP2
- Windows® 7 (32/64 bits) SP1
- Windows® 8 Professional/Ultimate (32/64 bits)

1.4.2 Hardware requirements

Table 1-2 Hardware requirements

| Hardware requirements for PC Worx Express | |
|---|---|
| CPU | Pentium 4, 1 GHz (2 GHz recommended) |
| RAM | min. 1 Gbyte (minimum), (2 Gbytes recommended) |
| Hard disk space | 2 Gbytes free memory space |
| DVD drive | Yes |
| Interfaces | 1 x Ethernet (TCP/IP) |
| Monitor | min. XGA, resolution of 1024 x 768 pixels (minimum), SXGA, resolution of 1280 x 1024 (recommended) |
| Operator panels | Keyboard, mouse |
| Web browser | Java Standard Edition SE 6 (or later) with at least Java Runtime Environment JRE 6 (Version 1.6.x or later) |

1.4.3 Required programming software

Table 1-3 Required programming software

| Software | Software version |
|----------|------------------|
| PC Worx | ≥ 6.30 |

1.4.4 Required firmware versions

Table 1-4 Required firmware version

| Device | Firmware version |
|----------|------------------|
| AXC 1050 | ≥ 2.10 |

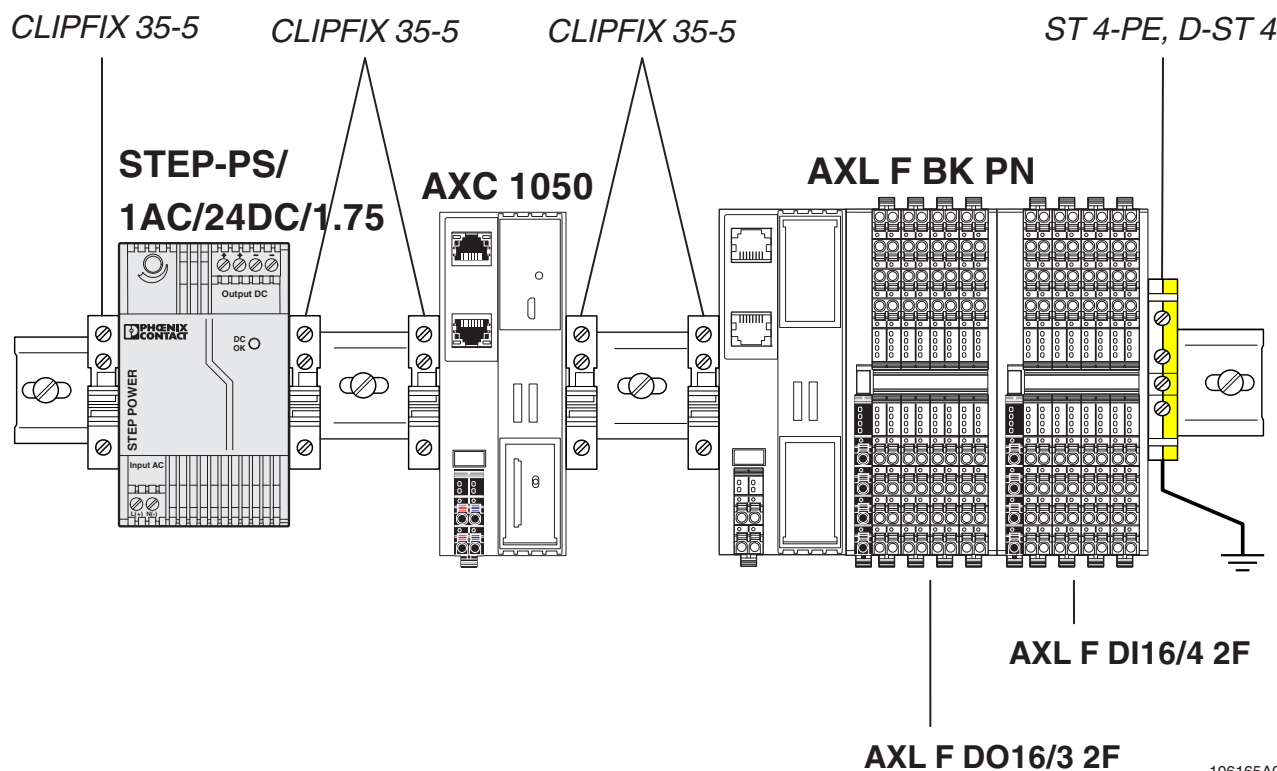


If the firmware version on your device is earlier than that specified in Table 1-4, please update your firmware. The latest firmware version can be downloaded at phoenixcontact.net/products.

2 Installing the hardware

One possible installation method is shown in Figure 2-1.

- Please mount all PROFINET components and the corresponding accessories on DIN rails. To do so, proceed as described in the corresponding package slips.



106165A001

Figure 2-1 Possible installation method for the AXC 1050 PN STARTERKIT

Key:

| | |
|---------------|--|
| Bold | Included in the AXC 1050 PN STARTERKIT |
| <i>Italic</i> | Order as accessories |

- Connect the supply voltage cables.

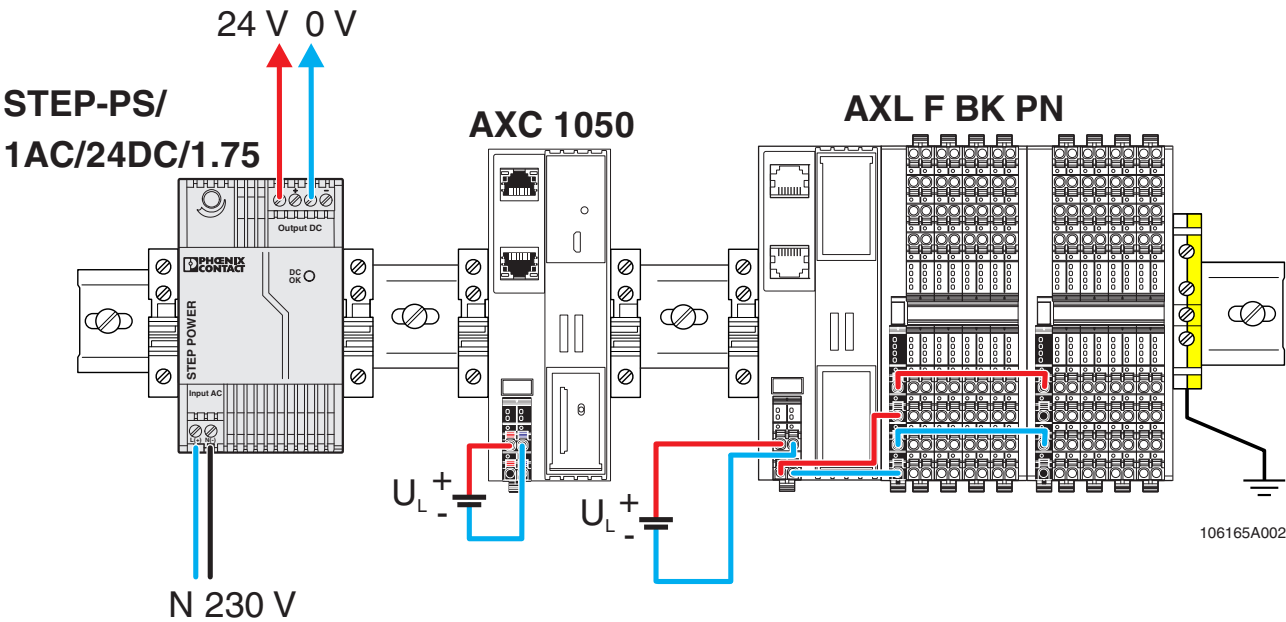


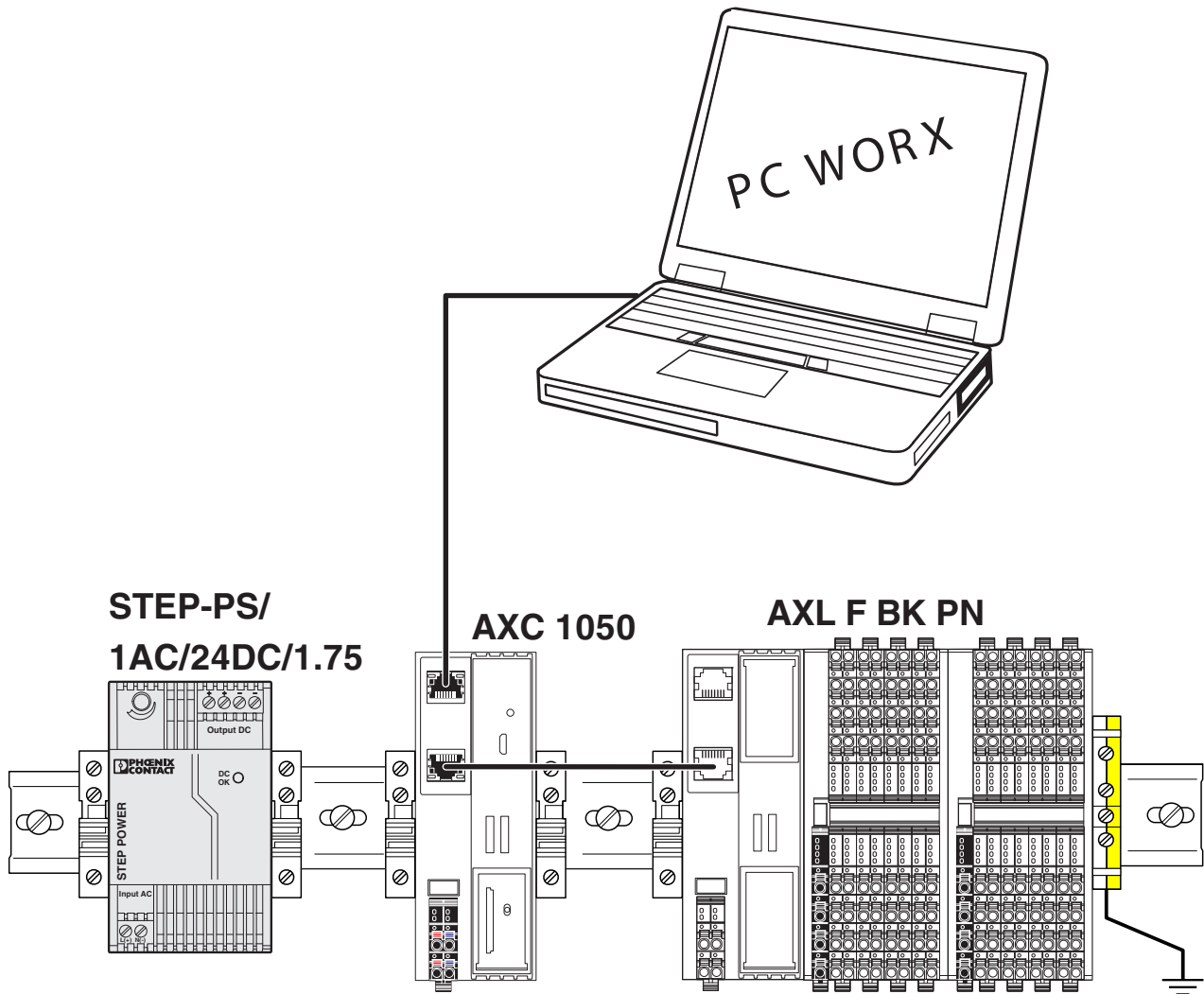
Figure 2-2 Possible installation of the AXC 1050 PN STARTERKIT – Illustration with supply voltage wiring

For the example program in this documentation, the following outputs are used:

Table 2-1 Outputs used in the example program

| Module | Input/output | Signal at | Variable |
|-----------------|-----------------|-----------|-----------------|
| AXL F DO16/3 2F | Outputs 1 ... 8 | 00 ... 07 | LED_0 ... LED_7 |

- Connect the Ethernet cables.



106165A003

Figure 2-3 Possible installation of the AXC 1050 PN STARTERKIT – Illustration with Ethernet cabling

- Connect the supply voltages.

3 Installing the software



- Prior to installation, close all open Windows® applications.

This PC Worx version can be installed parallel to an existing earlier version.

- Carry out all the installation steps described in this section one after the other. Observe the installation order specified here.

3.1 Starting the installation program of the AUTOMATIONWORX Software Suite

Once you have inserted the PC WORX DEMO DVD into the drive, the start page of the installation program opens automatically.



If the start page of the installation program does not open automatically, open the installation program via the "Index.html" file ("Computer, DVD Drive, Index.html").

- Select the English language version by clicking the American flag.
You are taken to the main page.
- Select "Programs".
- Click the "AutomationWorx Software Suite 1.82" link and download the displayed zip file.
- Extract the zip file to a folder.
- Start the unpacked "Setup.exe" file.
- Follow the instructions in the installation program.



During installation, the installation program asks which components from the AUTOMATIONWORX Software Suite should be installed.

- Install all components of the AUTOMATIONWORX Software Suite.

The installation program creates all directories that are necessary for operation and copies the files according to your selection in the installation program.

USB driver

The "Phoenix Contact USB to UART Interface" USB driver must be selected explicitly when installing the AUTOMATIONWORX Software Suite/PC Worx (see Figure 3-1). This is the only way to ensure that the AXC 1050 will be detected as a USB device under Windows®.

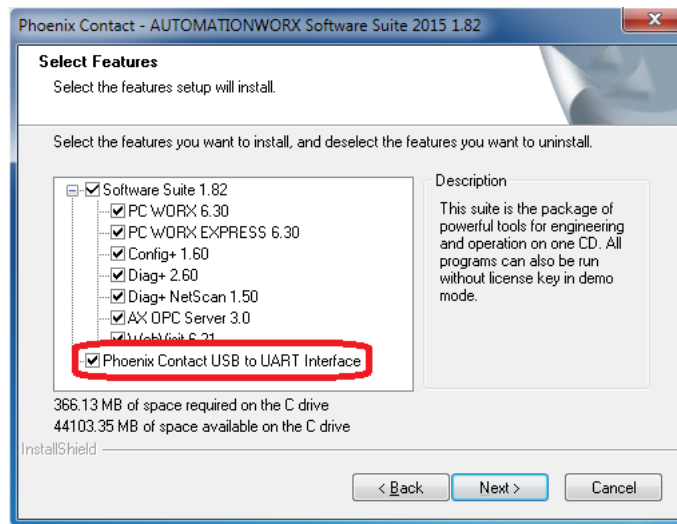


Figure 3-1 Selecting the USB driver

- You must restart your PC in order for the configuration file changes to come into effect. To do so, click on “Finish” at the end of the installation process.
- If, after installation, the AXC 1050 is not automatically detected as a USB device under Windows®, repeat the installation process and make sure that the “Phoenix Contact USB to UART Interface” USB driver is selected.

3.2 Starting PC Worx

For installation using the default settings, start PC Worx via “Start, Programs, Phoenix Contact, AUTOMATIONWORX Software Suite <Version>, PC Worx 6.xx”.

3.3 Enabling the PC Worx license

After starting PC Worx, proceed as follows to enable your license:

- Select the “Register...” command in the “?” menu.

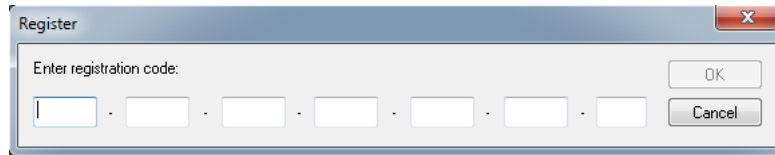


Figure 3-2 Dialog box for entering the registration code

- Enter the registration code.
- Confirm your entry with “OK”.



Alternatively, you can also license PC Worx via the License Manager in the AUTOMATIONWORX Software Suite.

Registration comes into effect the next time PC Worx is started.

4 Helpful information about PC Worx

4.1 Online help

More detailed information about the individual functions of PC Worx can be found in the on-line help for the program. The entire help function can be called via “Help” in the menu bar. Help for specific functions can be called via F1.

4.2 Selecting the language

When installing the software, the language in which PC Worx should be started can be selected. The program language can be changed at any time.

- Select the “Extras, Options...” menu.
- Select the “General” tab.
- Select the language.
- Confirm your selection with “Apply” and “OK”.

The selected language is activated the next time PC Worx is started.

4.3 The PC Worx user interface

The user interface consists of the following main components: menu bar, toolbars, main window, and status bar. The contents of the main window depend on the selected work-space.

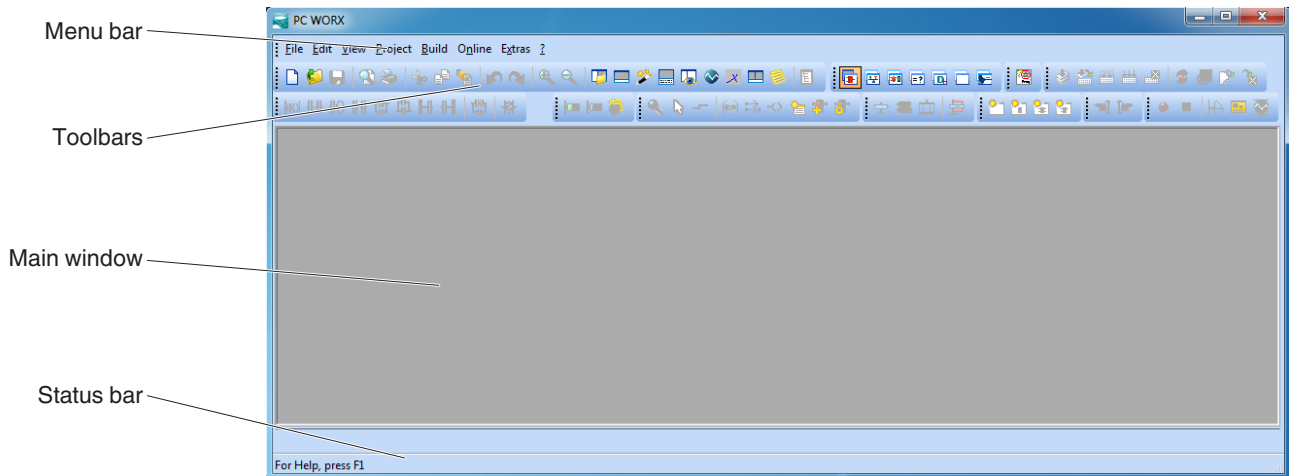


Figure 4-1 User interface

4.4 Toolbars

The program contains several toolbars with different icons, which enable frequently used operations to be executed quickly. Alternatively, these operating steps can be called via menu items or predefined shortcuts.

By default, all the toolbars are shown. To display or hide a specific toolbar, use the “Extras, Options” dialog box.

When the mouse pointer is placed over an icon (without clicking on it), a tool tip appears. The tool tip displays the name of the current icon. In addition, a short function description appears in the status bar. If tool tips are not displayed, this feature can be activated in the “Extras, Options, Toolbars” dialog box.

Icons for selecting the workspace

The workspace can be changed via the icons in the toolbar:



Open the “IEC Programming” workspace



Open the “Bus Configuration” workspace



Open the “Process Data Assignment” workspace



Open the “Project Comparison Result” workspace



Open the “FDT” (Field Device Tool) workspace

Frequently used icons

The following icons are frequently required for compiling and debugging:



Online modifications



Make (compile project; corresponds to “Build, Make” in the menu bar)



Rebuild project (corresponds to “Build, Rebuild Project” in the menu bar)



Switch debug mode on/off



Display project control dialog box

4.5 Workspaces

PC Worx is divided into the following workspaces:

- IEC programming
- Bus configuration
- Process data assignment
- Project comparison
- FDT (Field Device Tool)

The “View” menu or the corresponding icon in the toolbar can be used to switch between the workspaces. Following initial installation, the “IEC Programming” workspace is the default setting.

Figure 4-2 to Figure 4-6 show the default workspaces. The windows that you wish to display can be defined at any time for each workspace.



Which windows will actually be displayed depends on which windows have been toggled on. Select the windows that are to be toggled on or off via the “View” menu item.

The last setting for each workspace is saved when the program is closed and restored when it is started again.

“IEC Programming” workspace

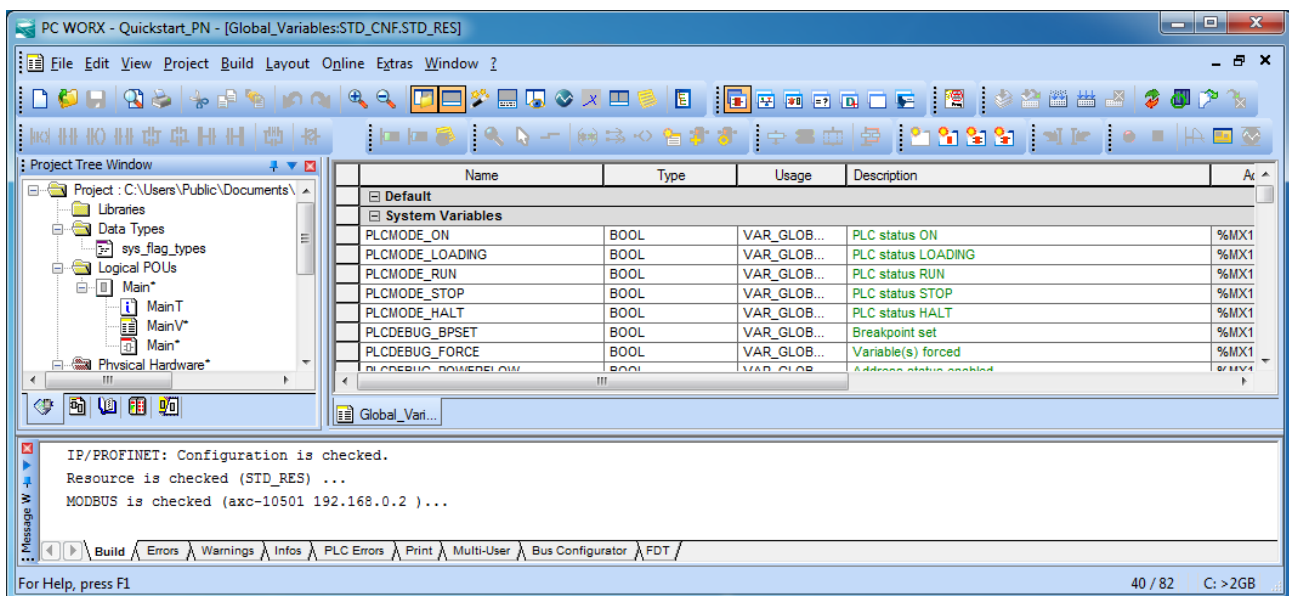


Figure 4-2 “IEC Programming” workspace

“Bus Configuration” workspace

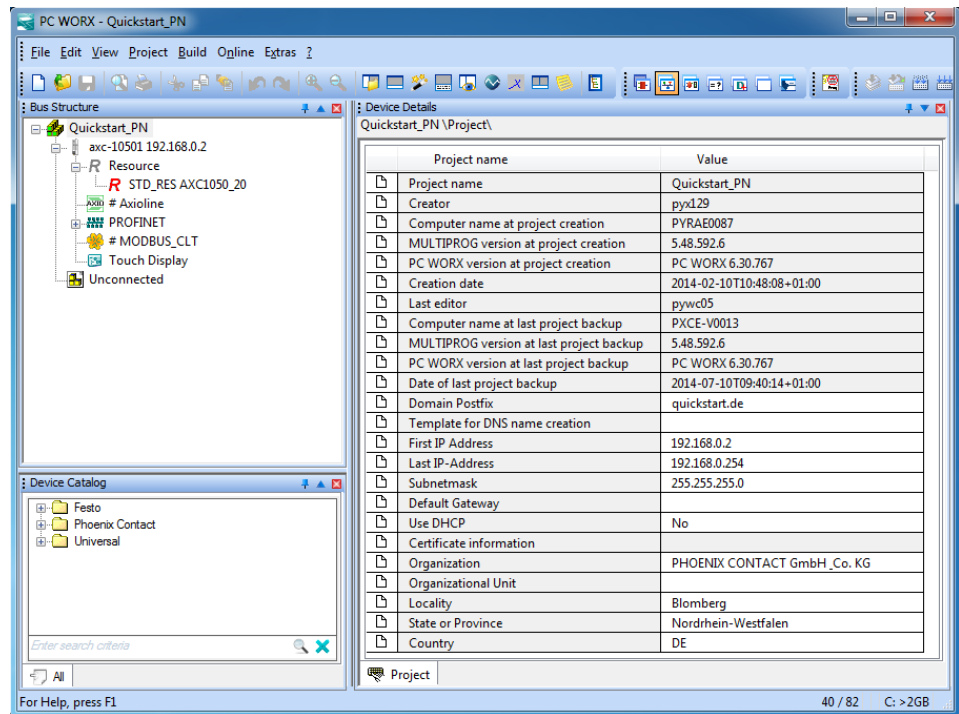


Figure 4-3 “Bus Configuration” workspace

“Process Data Assignment” workspace

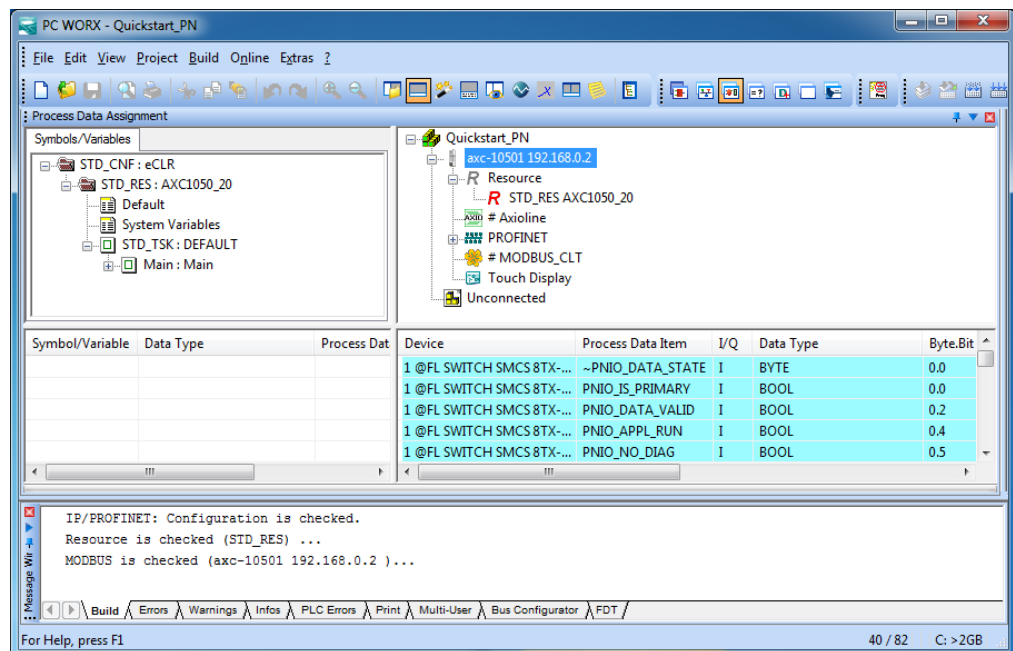


Figure 4-4 “Process Data Assignment” workspace

“Project Comparison Result” workspace

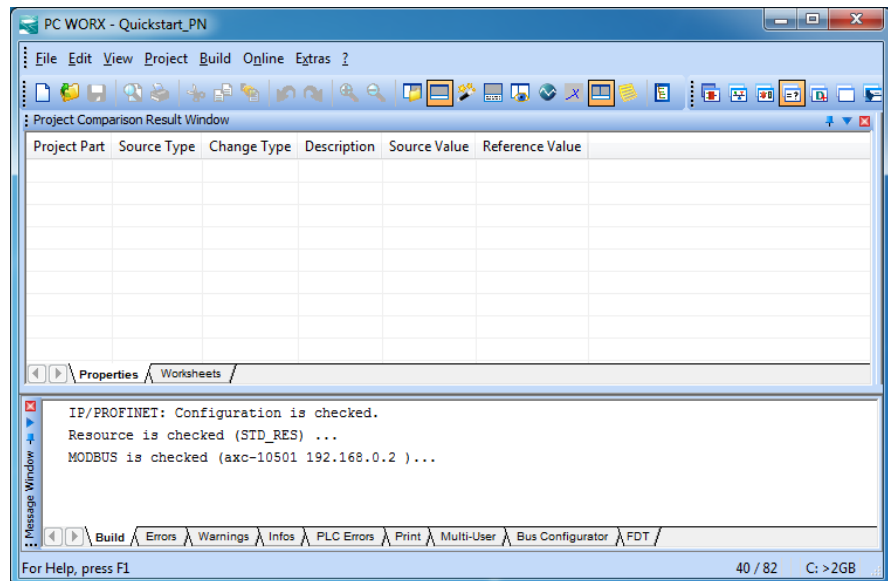


Figure 4-5 “Project Comparison Result” workspace

“FDT” workspace

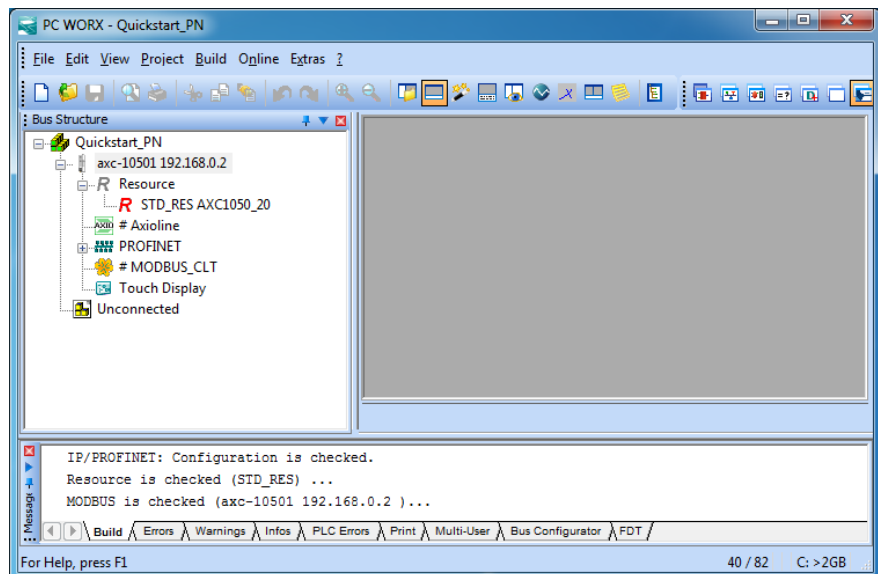


Figure 4-6 “FDT” workspace

Toggle windows on/off

Each window can be toggled on/off via the “View” menu by selecting the corresponding menu item.

Docking/undocking windows

For each window, you can specify whether or not it is to be docked in the other windows. There are various options for docking/undocking windows:

1. Permanent docking/undocking:
In the title bar or the gray frame of the relevant window, right-click with the mouse and activate/deactivate the “Allow Docking” menu item.
Move an undocked window to the position where it is to be inserted in the desktop.
2. Temporary undocking:
Double-click on the title bar or the gray frame of the window. The window is then displayed as a “normal” window. Its size can be modified and it can be moved to any position on the screen. In order to re-dock the window, i.e., to reinsert it in the desktop, double-click on the title bar.

4.6 “Bus Structure” window

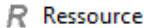
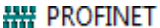
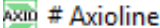
The “Bus Structure” window is used to display and edit the bus topology of the project.

4.6.1 Icons in the “Bus Structure” window

4.6.1.1 Logical device functions

The individual device functions are identified in the “Bus Structure” window with logical icons. The logical icons in the example bus configuration have the following meaning:





Table 4-1 Icons for logical device functions

| Icon | Meaning |
|---|---|
|  | Controller resource When creating the project using a template, the controller resource is inserted below this icon. When creating the project using the wizard, you insert the controller resource below this icon. |
|  | PROFINET controller Insert all PROFINET devices below this icon in the lower level. |
|  | Axioline In the example project, the Axioline icon is displayed below the PROFINET icon after the bus coupler has been inserted. Below this icon, insert the Axioline F modules that are connected to the bus coupler. |

4.6.1.2 Validity of actions

When editing the bus configuration with the mouse, the mouse pointer indicates the validity of your actions.





Table 4-2 Icons when editing the bus configuration

| Icon | Meaning |
|---|--|
|  | Insert at the same level The device can only be inserted in/moved to the same level as existing devices. |
|  | Insert in the lower level The device can only be inserted in/moved to a lower level than existing devices. |
|  | Replace The device under the mouse pointer can be replaced by holding down the <Ctrl> key and placing the mouse pointer on the existing device. |
|  | Not permitted This icon indicates a mouse pointer position for which actions are not permitted. |

4.6.1.3 Display of status information

In the “Bus Structure” window, some icons, which superimpose other graphics, are used to display important status information.

Table 4-3 Icons for displaying status information

| Icon | Meaning |
|---|---|
|  | The device is hidden or the bus is deactivated. |
|  | Errors have occurred for the device. |
|  | Warnings have occurred for the device. |
|  | The pin indicates a module that is a fixed component of a device. This element cannot be deleted or replaced. |

4.6.2 Display in the “Bus Structure” window

The display in the “Bus Structure” window can be adapted to your requirements.



- Switch to the bus configuration workspace.
- In the context menu of a device, open the “Edit Device Representation...” menu item.

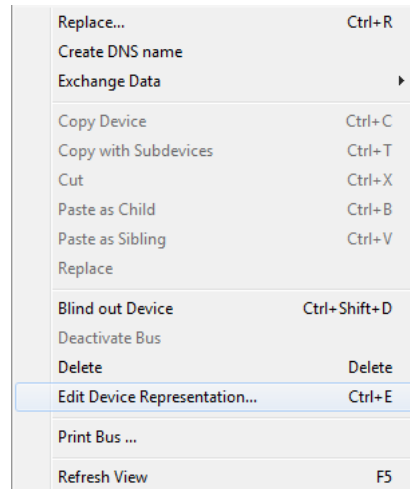


Figure 4-7 “Edit Device Representation...” menu item

- Select the criteria that you would like to see in the display.

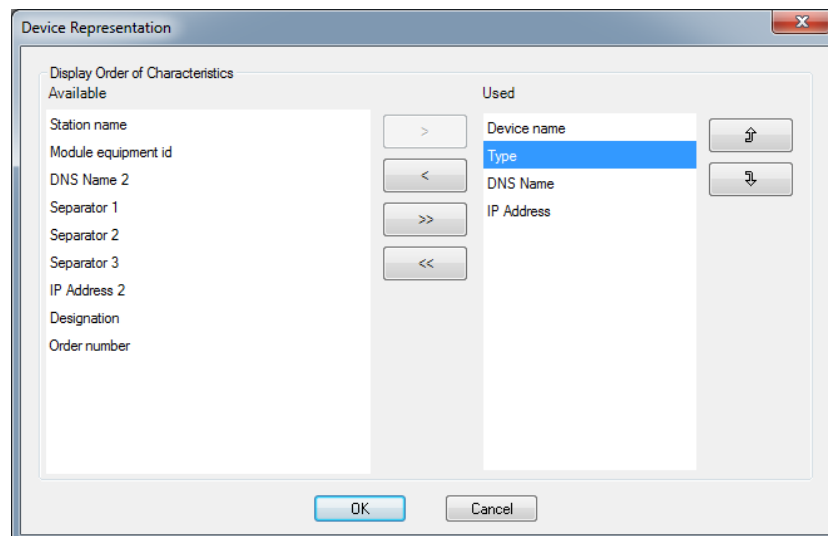


Figure 4-8 Device representation

Examples of different device representations:

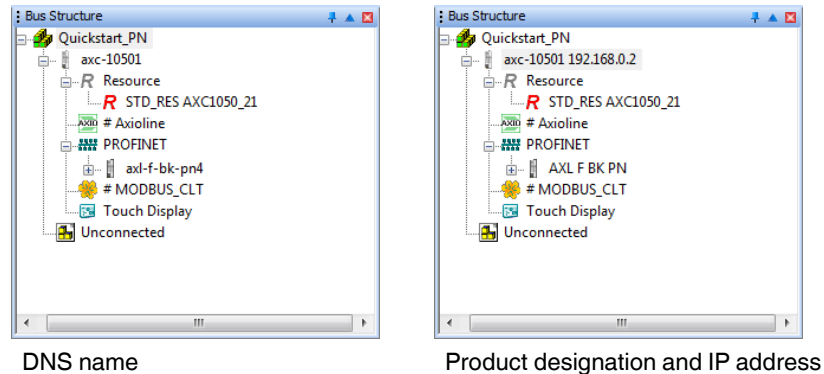


Figure 4-9 Examples of different device representations

The selected representation only affects the representation of a specific device group.

These groups are for example:

- Controllers
- PROFINET devices
- Axioline modules
- Modbus/TCP devices

For example, the set device representation for the AXC 1050 controller does not have any influence on the representation of the PROFINET devices (see Figure 4-10).

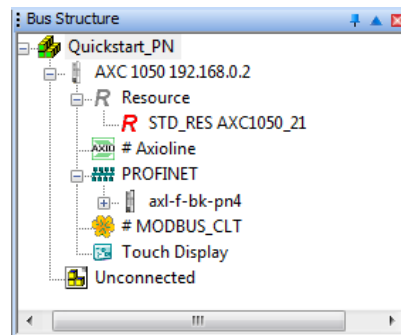


Figure 4-10 Example of different device representations of the individual device groups

4.7 Compiling and sending a project and performing a cold restart

4.7.1 Compiling a project

- When compiling a project for the first time, select the “Rebuild Project” command from the “Build” menu. For further compiling processes you can also use the “Make” command (see also “Compiling (additional information)” on page 33).
- If errors occur when compiling, remove the errors and repeat the compiling process until it is completed successfully. Error messages must be removed. Warning messages do not have to be removed.



If you are compiling the project but have not yet programmed anything, you will receive the “Empty worksheet” warning message. This warning message does not affect the next step. The warning message can be ignored.

The results of the compiling process are displayed in the “Message Window” together with details of the number of errors and warnings.

4.7.2 Downloading a project

When downloading the project to the controller, the desired PROFINET configuration is transmitted and activated.



- Open the “Project Control Dialog” dialog box.

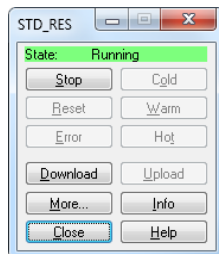


Figure 4-11 “Project Control Dialog” dialog box

Before downloading the project to the controller, the download options can be set.

- Click on “More...”.

Download options

The “Download Options” dialog box opens.

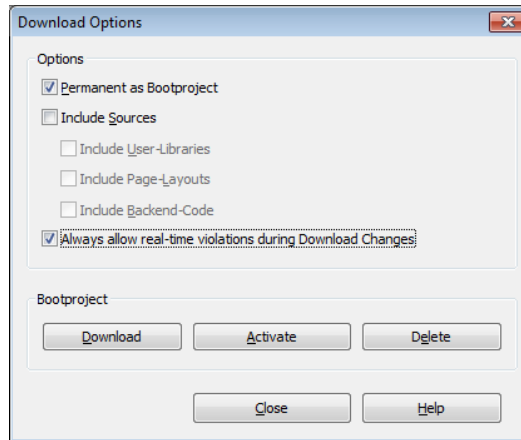


Figure 4-12 “Download Options” dialog box

In the dialog box in Figure 4-12, the following download options can be selected for the project, which is to be transmitted to the controller. In addition, the boot project of the current project can be sent separately or activated/deleted on the controller. The settings selected in this dialog box are used for all transmissions to this specific controller (AXC 1050 in the example).

“Options” area

- | | |
|-----------------------------|---|
| – Permanent as Boot-project | This option can be deactivated to exclude the boot project when downloading to the controller. |
| – Include Sources | This option enables additional options that can be used to select sources, which are included when downloading a project. |
| – Include User-Libraries | All included user libraries are included in the packed program source. |
| – Include Page-Layouts | All page layouts that are used are included in the packed program source. |
| – Include Backend-Code | All project source data including the compilation is included in the packed program source. |

- Always allow real-time violations during Download Changes

This checkbox refers to the “Download Changes” function.

When it is selected, the following standard procedure applies: next time “Download Changes” is executed, the system does not attempt to observe the realtime conditions (i.e., execution of the user task at the exact time) on the controller. Instead the system stops program execution on the controller, downloads the project, copies all instance-related data, and then resumes execution of the modified program. The system does not issue a warning message prior to the associated violation of realtime conditions.

Use of this option is recommended if “Download Changes” cannot be executed while observing the realtime conditions because there is too much POU instance data (modified project parts) to be copied.

Once you have selected the checkbox, this setting is applied to all future “Download Changes” actions.



The box is also selected if the checkbox of the same name is selected in the message dialog box (“Download changes could not be executed in realtime. Try again?”).



NOTE: If realtime violations are permitted, first ensure that your project is running within safe limits. Please note that each realtime violation may result in unexpected consequences in the automation system, as under certain circumstances user tasks are not processed for a short time. Therefore, check the level of risk before enabling this option.

“Bootproject” area

- “Download” button
- “Activate” button
- “Delete” button

The boot project is downloaded separately to the controller without the complete project.

The boot project saved on the controller is activated. Following activation, the boot project can be executed.

The boot project saved on the controller is deleted.

- Close the “Download Options” dialog box once you have made the necessary settings for your project.
- Click “Download” in the “Project Control Dialog” dialog box.

The project is now in the RAM of the controller.

4.7.3 Performing a cold restart

In order to activate the project, it is necessary to perform a cold restart.

- To do this, click “Cold” in the “Project Control Dialog” dialog box.

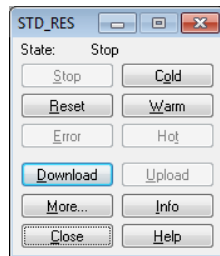


Figure 4-13 “Project Control Dialog” dialog box

The AXC 1050 has been started up successfully if the green RUN LED is on.

4.7.4 Compiling (additional information)

There are two options for compiling:

- 1 “Build, Make”
- 2 “Build, Rebuild Project”

Rebuilding the project

Use this command to compile an **entire** project for the first time or after modifying a user library.

“Rebuild Project” compiles and links all worksheets. Errors and warnings that are discovered by the compiler are logged in the “Message Window”. After the syntax has been checked successfully, the system automatically generates the IEC code and the special PLC code. Finally, the project can be downloaded to the controller.



The “Rebuild project” command should only be used if errors occurred when compiling with “Make” or if your project was unpacked without frontend.

Make

The “Make” command is the standard mode for compiling. Use this command after editing and completing your project.

When the “Make” menu item is executed, all **modified** worksheets are compiled/linked and the modified PLC code is generated.

This command can be executed from the menu bar via “Build, Make”, with the “Make” icon in the toolbar or via F9.

Modified worksheets in the project tree are identified by an asterisk that appears behind the worksheet name.

Both

After the compiler has been started, the “Message Window” appears automatically if it was closed before. This window shows the steps the compiler is currently performing. In addition, errors, warnings, and additional information about the process are logged here.

Once the compiling process has been completed successfully, (i.e., no errors have been reported), the modified project can be downloaded to the controller.

5 Example project

In this section, it is assumed that the following steps have already been completed:

| Step | Section |
|------------------------------|--|
| – The hardware is installed. | See Section 2 “Installing the hardware” |
| – The software is installed. | See Section 3, “Installing the software” |

5.1 Overview

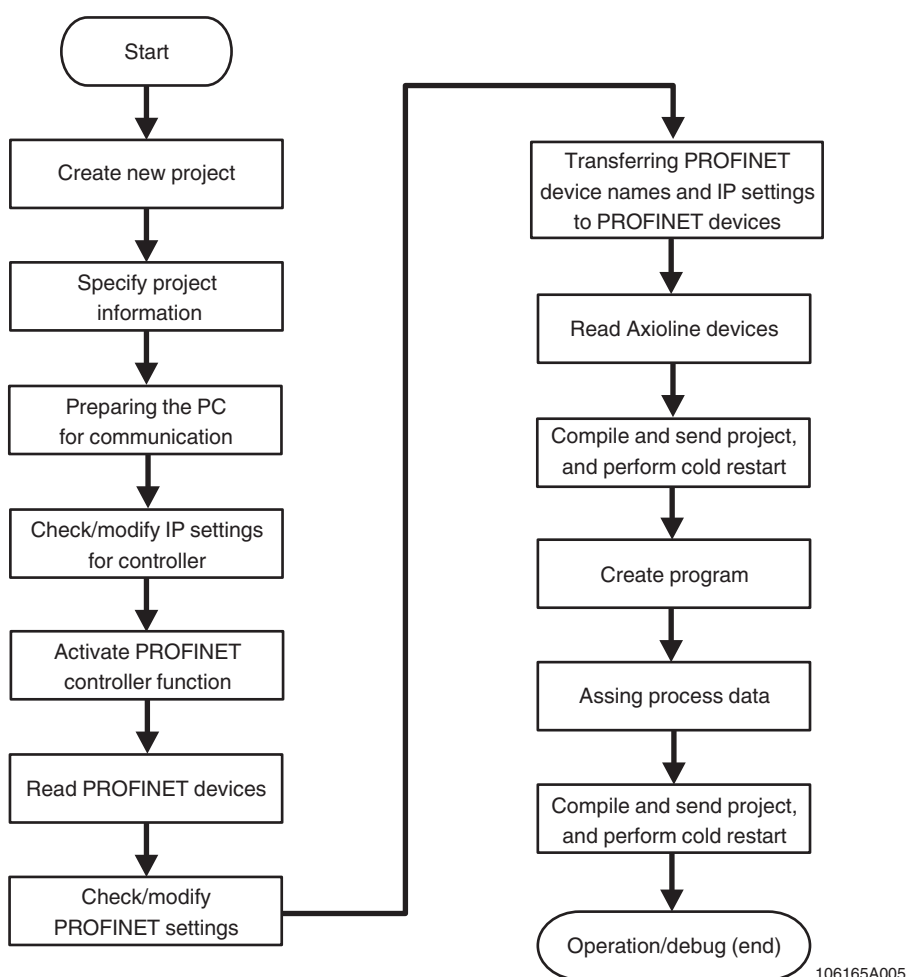


Figure 5-1 Sequence for creating a project

5.2 Creating a new project

In the following sections, an example project is developed.



Please note that the example project is available in compressed format as the “Quickstart_PN.zwt” file on the CD AXC 1050 PN STARTERKIT.

- If you want to try out the following steps yourself, please use the same identifiers and names as in this user manual in order to achieve the best possible result.
- Read the detailed description in Section “Example project on CD” on page 68 if you want to use the example project directly from the CD in PC Worx.

- Select the “New Project...” command from the “File” menu to create a new project using a template.
- Select the “AXC 1xxx” tab because the AXC 1050 controller is used.

The tree structure and the selection of the controller are now prepared.

- Select the “AXC 1050 Rev. > 01/2.10” controller and confirm your selection with “OK”.

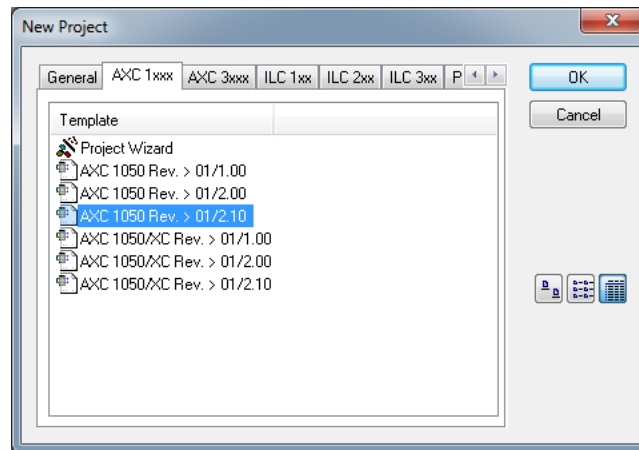


Figure 5-2 “New Project” dialog box: project templates

- Select the “File, Save Project As / Zip Project As...” command.
- Enter a project name (here: Quickstart_PN) and save the project.

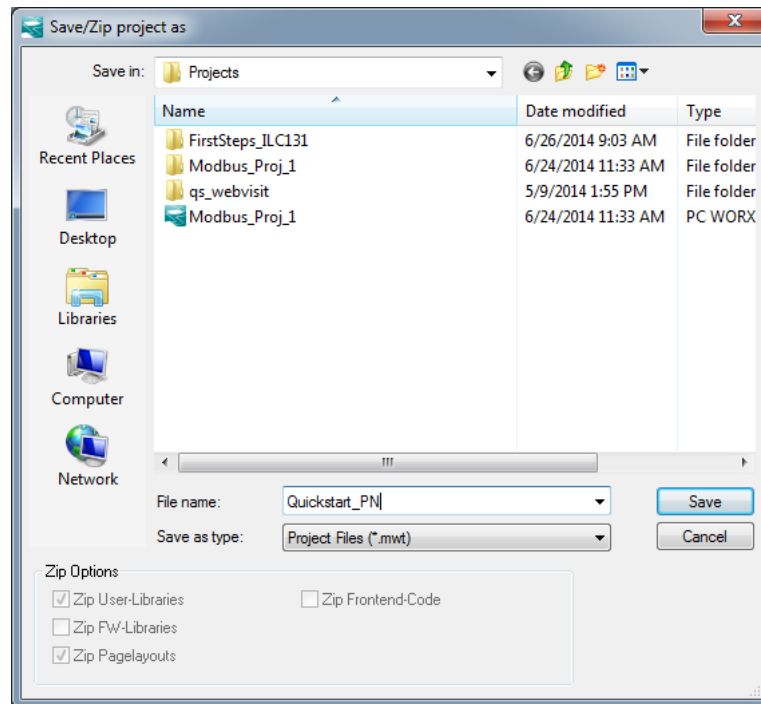


Figure 5-3 “Save/Zip project as” window

5.3 Adapting the project information



- Switch to the bus configuration workspace.

After creating a new project, the project information is displayed in the bus configuration workspace.

- Adapt the project information to your project.

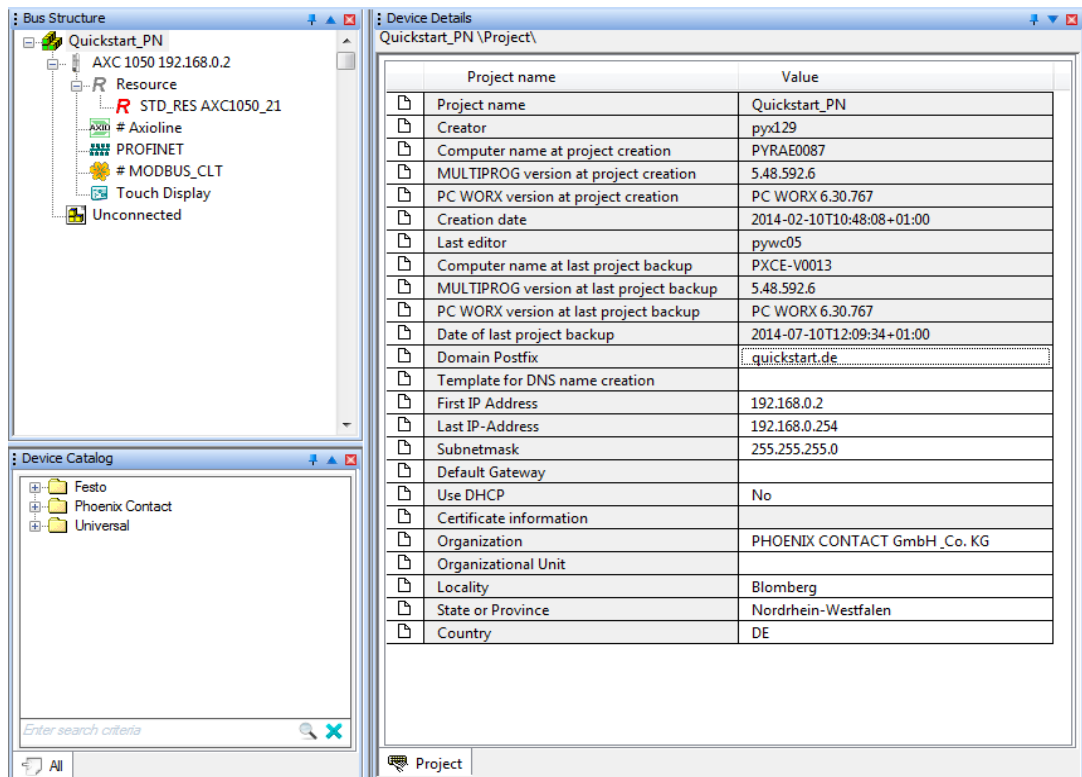


Figure 5-4 Project information after creating a new project

Domain postfix

- Assign the name suffix for the PROFINET device name (here: quickstart.de) in the “Device Details” window under “Domain Postfix”.

The following characters are permitted for the “Domain Postfix”:

Lowercase letters without umlauts a to z

As separator: dot .



Upper case letters without umlauts are permitted, however, they will be converted to lower case letters internally. The entry is therefore not case-sensitive.

If the “Domain Postfix” is entered at the start of the project, it will be used automatically for all subsequent PROFINET devices that are inserted in the PROFINET device name (see also “DNS name” on page 52).



If the PROFINET devices are read in, the PROFINET device name for the devices will also be read in. In this case, the “Domain Postfix” may differ. Adapt it accordingly.

**First and last
IP address,
subnet mask**

During project creation, PC Worx automatically assigns an IP address area for a local network (area from 192.168.0.2 to 192.168.0.254). If you would like to use a different address area (e.g., a global network), adapt the start and end address on the project node (here: Quickstart_PN) to your application.

If the IP parameters are modified in the project settings, the IP addresses of the PROFINET controller may also have to be modified manually (see Section “Checking/modifying IP settings for the PROFINET controller” on page 40).

When inserting or reading PROFINET devices, PC Worx automatically assigns IP parameters that are within the specified area.



If you are using the addresses of a local network in your project, also assign the corresponding address (e.g., 192.168.0.225) to the PC on which PC Worx is installed. Otherwise the devices in the local network cannot be accessed.

For information on the IP addresses to be used within your system, please contact your system administrator.

If the start address of the new address area to be entered is higher than the previous end address, modify the end address first.

If you manually assign IP parameters that are outside the defined area, they will not be accepted.

Default gateway

If you are using a default gateway:

- Enter the address for the default gateway under “Default Gateway” in the “Device Details” window.

This address must be within the specified IP address area. If the default gateway is entered at the start of the project, it will be used automatically for all subsequent Phoenix Contact devices that are inserted (see also Section 5.8).

5.4 Preparing the PC for communication

In the “Extras, PROFINET Configuration...” menu in PC Worx, select the network card for your computer that is to be used for communication.

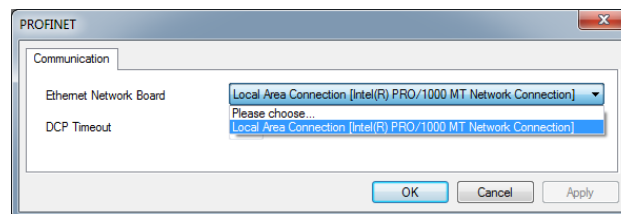


Figure 5-5 Selecting the network card

5.5 Checking/modifying IP settings for the PROFINET controller

The IP settings for the controller are made when the project is created.



Changes to project information are not applied automatically

If any modifications are made to the project information (see Figure 5-4) that affect the IP settings for the controller, a warning is displayed. However, the modification is not implemented automatically.

When a new project is created, the default settings are specified under “IP Settings”.

Adapt the IP settings, if required.



- Switch to the bus configuration workspace.
- Select the controller node.
- In the “Device Details” window, switch to the “IP Settings” tab.
- Check the IP settings and modify them, if necessary. Make sure that the controller IP address is within the project IP address area.

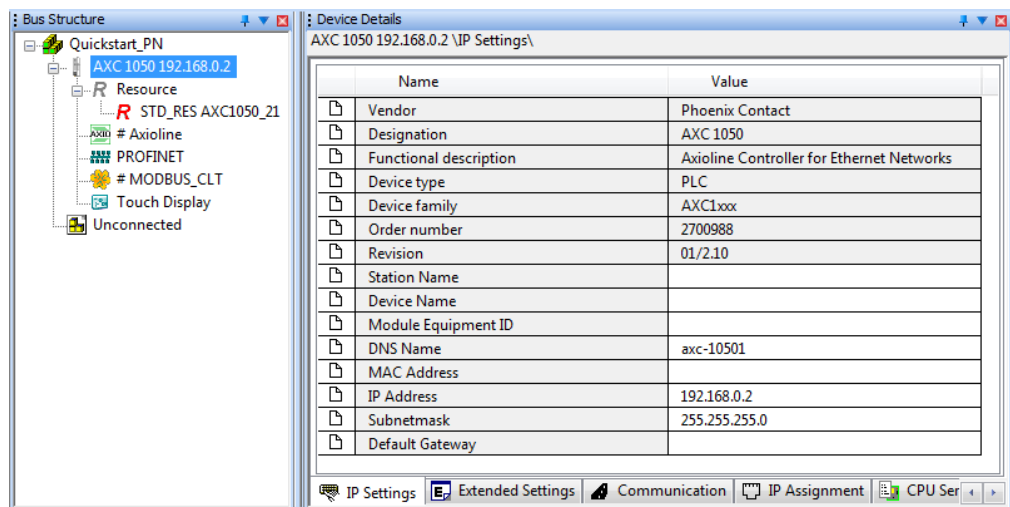


Figure 5-6 IP settings



The IP parameters that are assigned here for the controller are also implemented as the IP parameters for the communication path via TCP/IP.

5.5.1 Assigning the IP address for the PROFINET controller

By default for the AXC 1050 PN STARTERKIT, the AXC 1050 has no preset IP address. Initial setting of the controller's IP address can be carried out with the PC Worx software manually via the DCP protocol or manually via the USB interface. The IP address can be changed later using the PC Worx software via the USB connection, Ethernet, or the DCP protocol.



Phoenix Contact recommends that you assign the IP address via the DCP protocol.

5.5.1.1 Dynamic Configuration Protocol (DCP)



The AXC 1050 controller from firmware version 2.10 or later and from PC Worx version 6.30.767 or later (part of the AUTOMATIONWORX Software Suite 1.81 incl. AddON V1) supports the DCP protocol.

The DCP protocol is used to assign an IP address and a name to individual network devices. The PC Worx software sends a broadcast request with a DCP telegram to the devices within the network. The devices respond with their MAC address and device type. By default upon delivery, the AXC 1050 controller has no IP address and therefore responds with 0.0.0.0 and its specific MAC address. In the next step, the IP address is set in PC Worx; then the IP address is transmitted to the AXC 1050 by means of a DCP telegram.

PC/network adapter

To determine whether your network permits the IP settings used in the example project (see Figure 5-4 on page 38), proceed as follows:

- In the Windows® Control Panel, check the settings for your PC network adapter.
- If necessary, adjust these settings so that the AXC 1050 can be accessed in your network via the IP address used in the example project.

If your network does not permit the use of the IP address used in the example project, adjust the settings in the project information accordingly (see Figure 5-4 on page 38).



Changes to project information are not applied automatically

If any modifications are made to the project information (see Figure 5-4) that affect the IP settings for the controller, a warning is displayed. However, the modification is not implemented automatically.

When a new project is created, the default settings are specified under "IP Settings".

The IP address is assigned via the DCP protocol in the PC Worx software via the "Device Details" window:

- Select the controller in the "Bus Structure" window.
- Select the "IP Settings" tab in the "Device Details" window.
- Enter the IP address of the controller (here: 192.168.0.2).

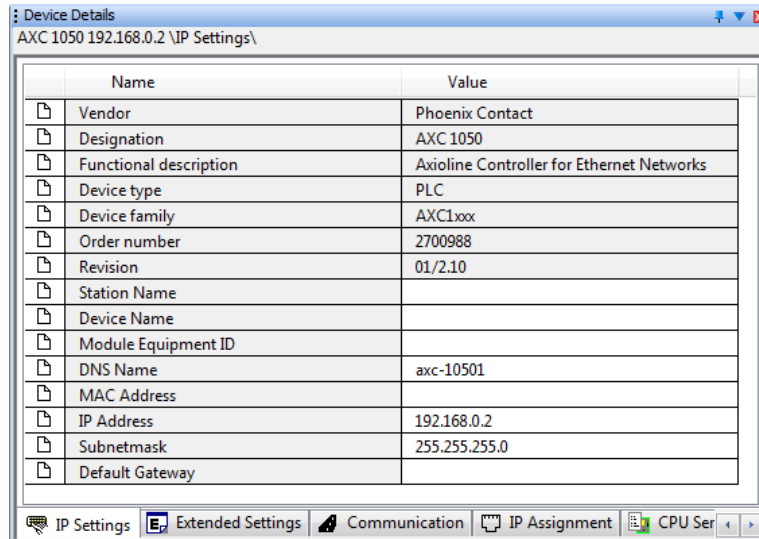


Figure 5-7 DCP: setting the IP address

The “IP Assignment” tab is used for actual IP address assignment with DCP.

- Select the “IP Assignment” tab.

After selecting the “IP Assignment” tab, the PROFINET network is searched for DCP devices.

- Deactivate the checkbox in the “Filter” area.
- Select the “[Unknown Type]: AXC 1050” controller.
- Click on “Assign IP” to start IP address assignment with DCP.

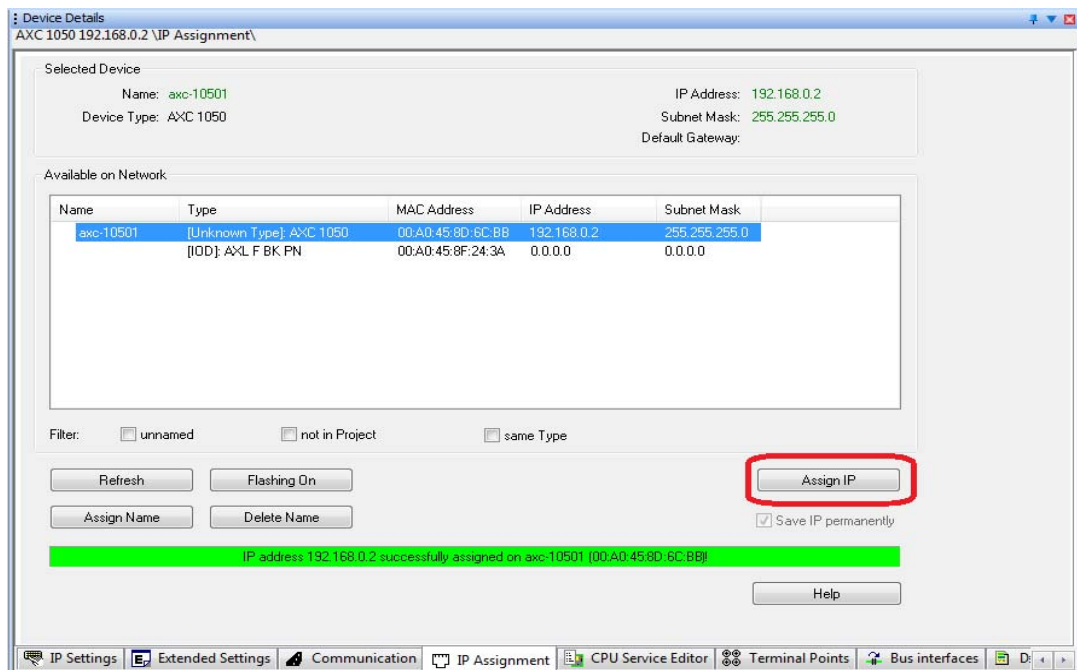


Figure 5-8 Starting IP address assignment via DCP

A green status indicator indicates successful IP address assignment.

- Assign the PROFINET device name in the same way.

Then check the communication between PC Worx and the AXC 1050:

- In the “Device Details” window select the “Communication” tab.
- Click on the “Test” button.

A green status indicator indicates successful communication.

If communication between PC Worx and the AXC 1050 could not be established:

- Check again whether your network permits the IP settings used in the example project (see Section 5.3 on page 38).
- Adjust the settings, if required, then check again communication between PC Worx and the AXC 1050.

5.5.1.2 USB interface



The CAB-USB A/MICRO USB B/2,0M USB connecting cable (Order No. 2701626) is available as an optional accessory for connecting the controller to a PC via the USB interface.



In order to use the USB interface of the AXC 1050, the “Phoenix Contact USB to UART Interface” driver must be installed (see Section 3.1 on page 17).

- Establish a USB connection between the PC on which the PC Worx software is running and the AXC 1050.

If you selected to install the USB driver when installing the AUTOMATIONWORX Software Suite/PC Worx (see Section 3.1) and if your PC supports automatic detection of USB devices, the PC will immediately detect the AXC 1050 as a USB device. No other settings are required.

Manual assignment of IP address settings



- Switch to the bus configuration workspace.
- Select the controller node (in the example: “AXC 1050”).
- Select the “Extended Settings” tab in the “Device Details” window.
- Select the “Ethernet” setting below the “Extended Settings” folder.
- In the “Network Settings” area, select the “Manual definition of the TCP/IP settings” item.
- Enter the required IP address (in the example in Figure 5-9: IP address: 192.168.0.02; subnet mask: 255.255.255.0).

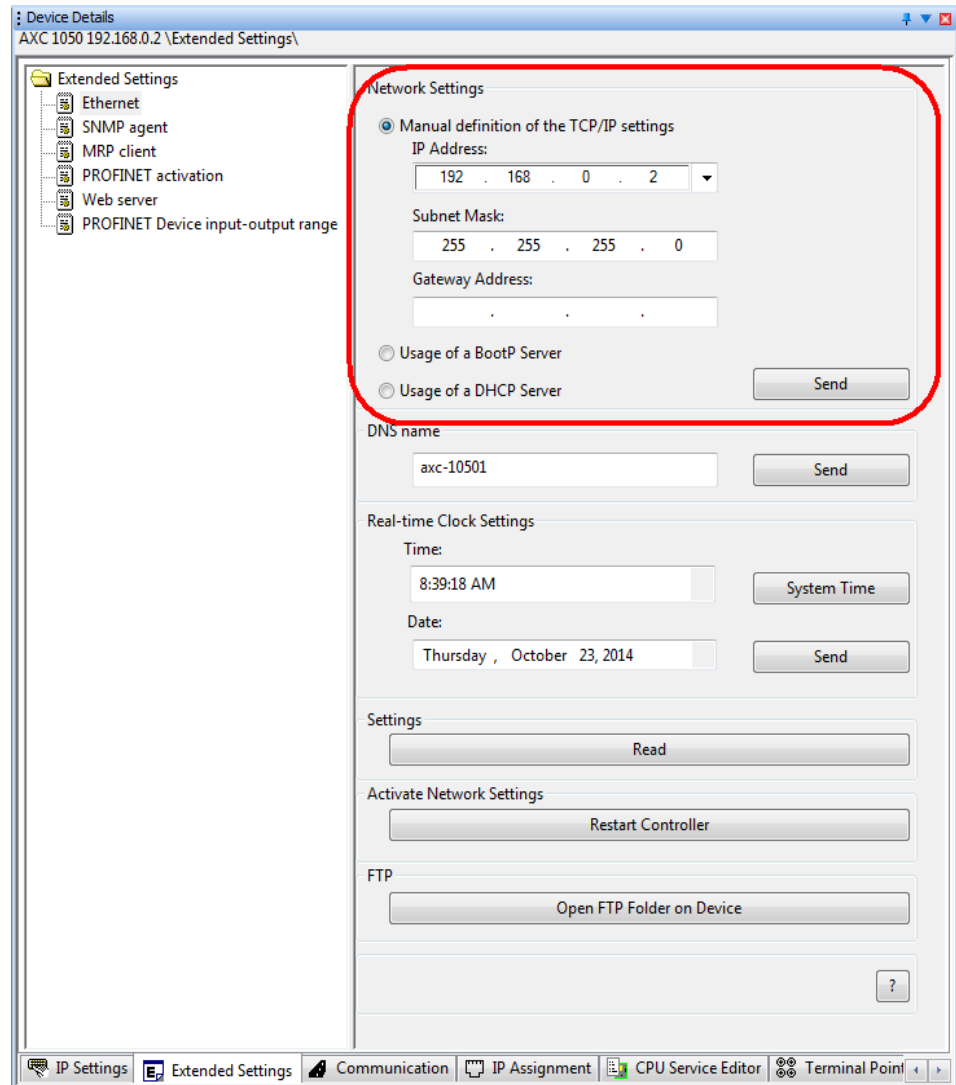


Figure 5-9 Network settings: manual assignment of the IP address

- Click on “Send” in the “Network Settings” area.
- In the “Settings Communication Path” dialog box, select the USB interface and confirm with “OK”.

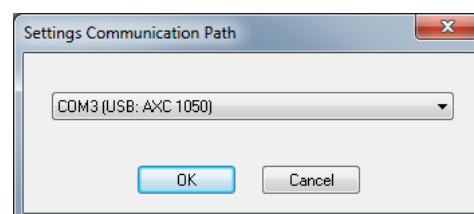


Figure 5-10 “Settings Communication Path” dialog box

Successful download of the IP settings to the controller is displayed in the status window:

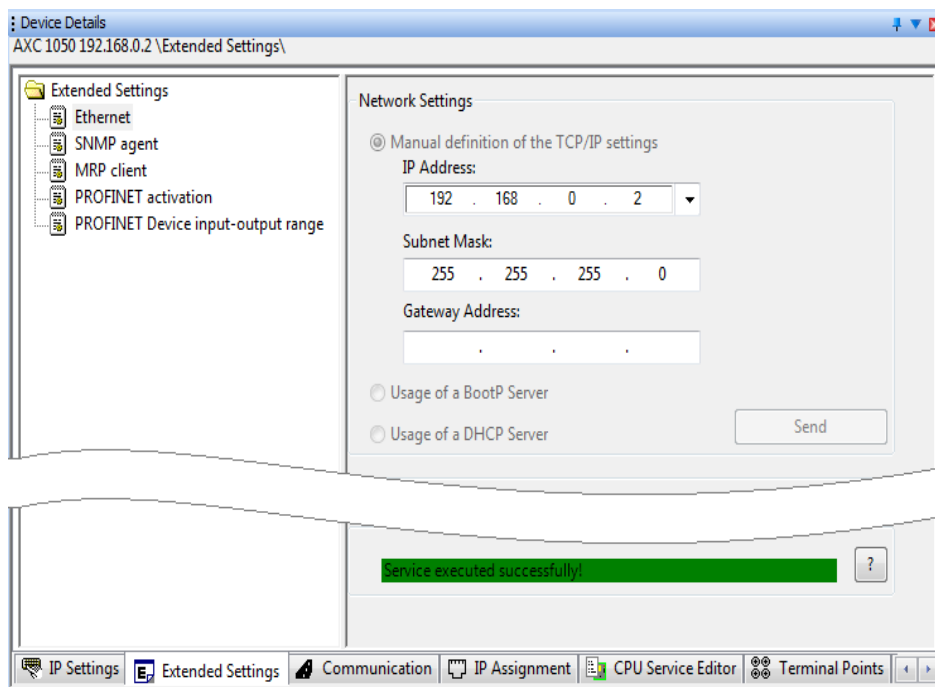


Figure 5-11 Successful download of the IP settings

To accept the IP settings, the AXC 1050 must be restarted.

- Select the “Ethernet” setting below the “Extended Settings” folder.
- In the “Activate Network Settings” area, click on the “Restart Controller” button.

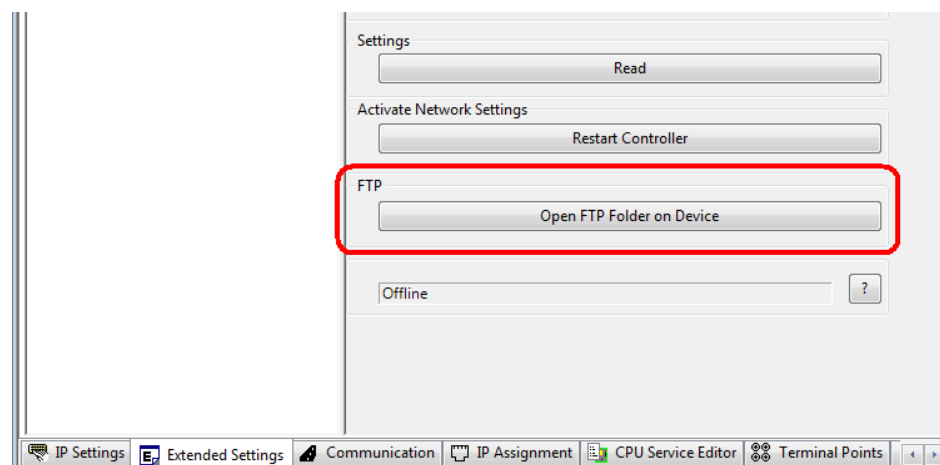


Figure 5-12 Activating the network settings: restarting the controller

- If a prompt appears asking whether you really want to restart the controller, confirm the dialog box with “Yes”.
- In the “Settings Communication Path” dialog box, select the USB interface and confirm with “OK” (see Figure 5-10).

Successful execution of the service will be displayed in the status window.

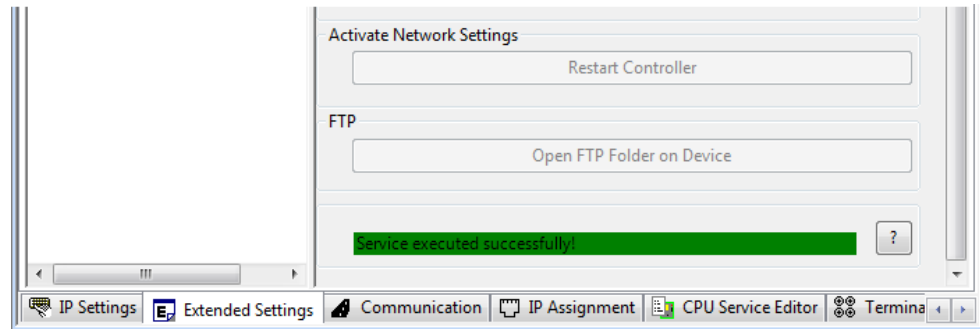


Figure 5-13 Activating the network settings: service executed successfully

The network settings assigned to the AXC 1050 are active.

You can test whether the settings are working correctly on the “Communication” tab in the “Device Details” window.

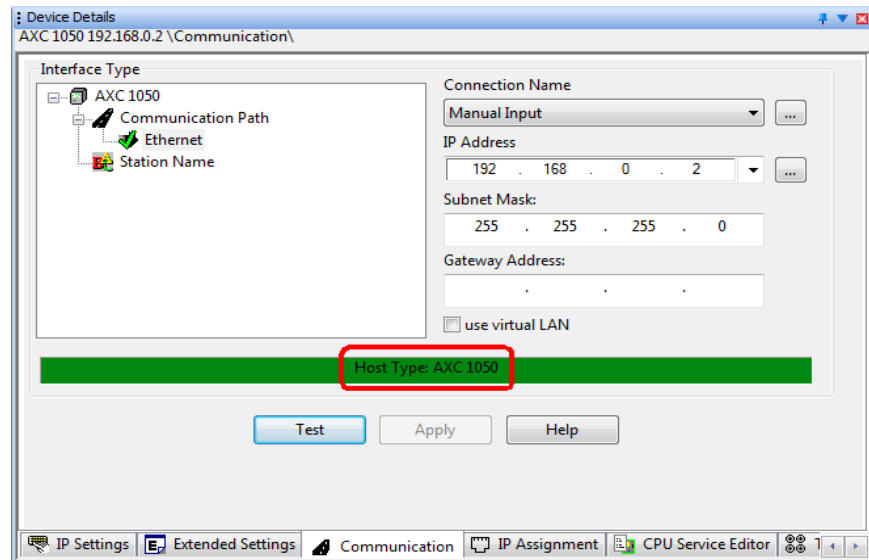


Figure 5-14 Successful communication path test

5.6 Activating the PROFINET controller function

Once you have assigned an IP address for the AXC 1050, the PROFINET controller function must be activated.

- Select the “Extended Settings” tab in the “Device Details” window.
- Select the “PROFINET activation” setting below the “Extended Settings” folder.
- In the “Settings” area, select “PROFINET controller activated” from the drop-down list.

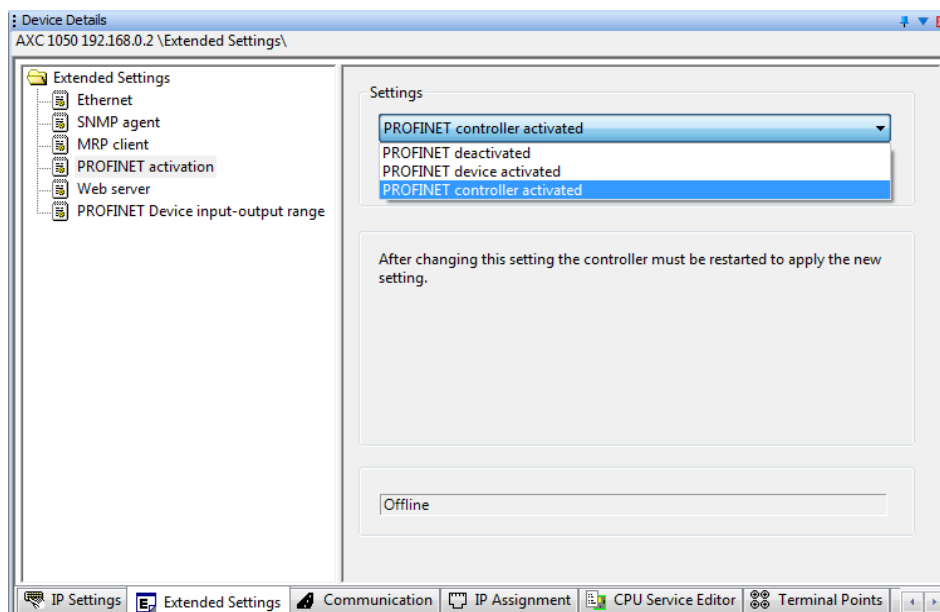


Figure 5-15 Activating the PROFINET controller function

- Click on “Send”.
- In the “Settings Communication Path” dialog box, confirm the suggested communication path or a communication path suitable for your application with “OK”.

Successful execution of the service will be displayed in the status window.

To activate the enabled PROFINET controller function on the AXC 1050, the AXC 1050 must be restarted.

- In the “Activate Network Settings” area, click on the “Restart Controller” button (see Figure 5-12 on page 45).
- If a prompt appears asking whether you really want to restart the controller, confirm the dialog box with “Yes”.
- In the “Settings Communication Path” dialog box, confirm the suggested communication path or a communication path suitable for your application with “OK”.

The BF LED flashes following successful execution of the service. The AXC 1050 can now be read in as a PROFINET device.

5.7 Reading in PROFINET devices



- Make sure you are in the bus configuration workspace.
- In the “Bus Structure” window, select the PROFINET node of the controller.
- Right-click to open the context menu and select “Read PROFINET”.

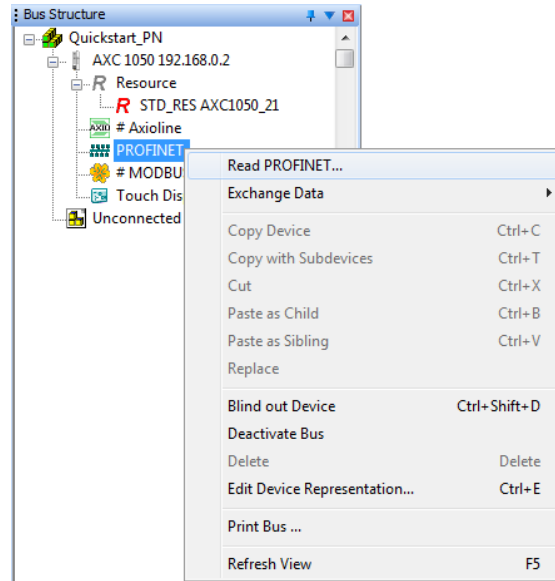


Figure 5-16 Reading in PROFINET

All the connected PROFINET devices are displayed. When reading in, the device name and the IP parameters are imported for each device - if present.

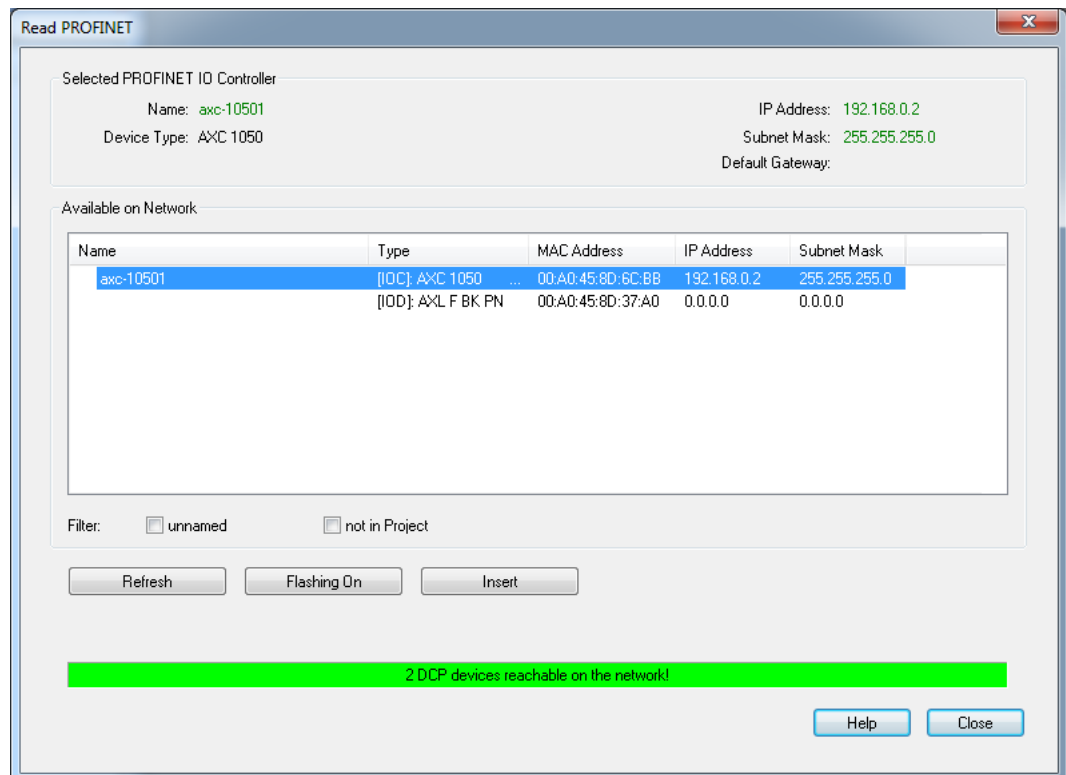


Figure 5-17 Connected PROFINET devices

- Select the devices that are to be included in your project.

The AXC 1050 controller is already included in the project. It is therefore only necessary to insert the AXL F BK PN in the project.

Importing PROFINET devices into the project

- Click on the “Insert” button to import the selected devices into the project.

Selecting the PROFINET device description

- If the “Select PROFINET device description” dialog box opens, select the device description (here: “AXL F BK PN”, version “01/1.01”).

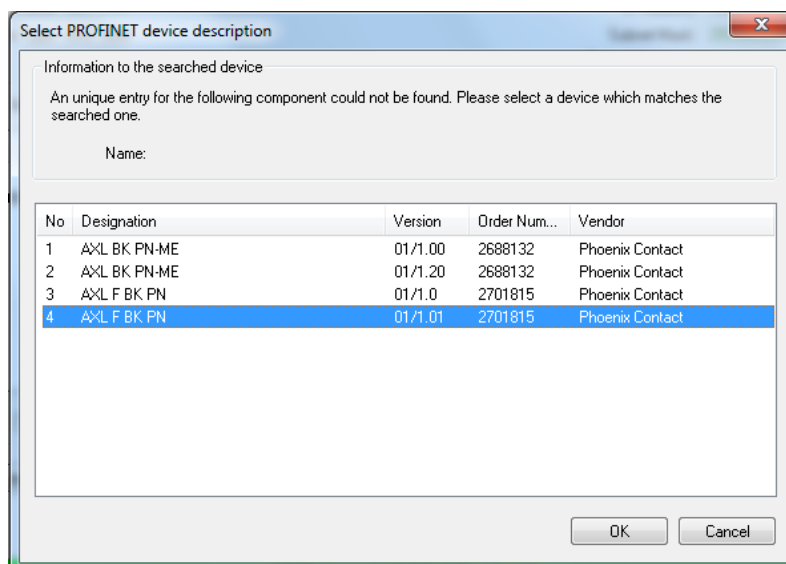


Figure 5-18 Selecting the PROFINET device description

- Confirm your selection with “OK”.

Naming the PROFINET device

If a PROFINET device name does not yet exist for a device (“DNS Name”), a name should be assigned at this point in the “Insert PROFINET Device” dialog box that opens.

The PROFINET device name (“DNS Name”) is the unique identification for the PROFINET device in the network. The PROFINET device name must be known to the PROFINET device before the PROFINET device can be used in the network.

PC Worx automatically ensures that each device is assigned a PROFINET device name (“DNS Name”) that is unique throughout the entire project. The “Domain Postfix” that was set on the project node is attached to this name (here: quickstart.de).

The PROFINET device name can be freely modified.

The following characters are permitted for the PROFINET device name:

| | |
|-----------------------------------|--------|
| Lowercase letters without umlauts | a to z |
| As separator: dot | . |
| Hyphen | - |

If the naming conventions have not been observed, an error message is output in the “Message Window” and the name is rejected.



Upper case letters without umlauts are permitted, however, they will be converted to lower case letters internally. The entry is therefore not case-sensitive.

- Check the displayed PROFINET device name and modify it, if necessary.
- Activate the “Name Device” checkbox.
- Then click “OK”.

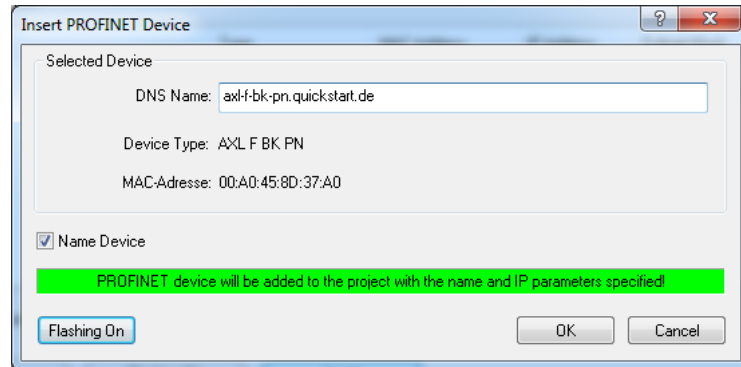


Figure 5-19 Naming the PROFINET device

- Close the “Read PROFINET” window.

The PROFINET topology is displayed in the “Bus Structure” window. The current PROFINET device name was assigned and the IP address was specified for each device.

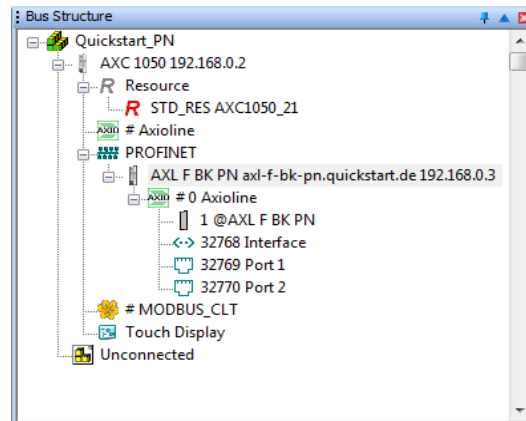


Figure 5-20 Bus configuration with read in PROFINET devices

The current settings can also be checked in the “Read PROFINET” window. All PROFINET devices should now have a PROFINET device name. The IP parameters are only assigned after project download and controller startup.

The updated list contains the PROFINET devices together with their PROFINET device names.

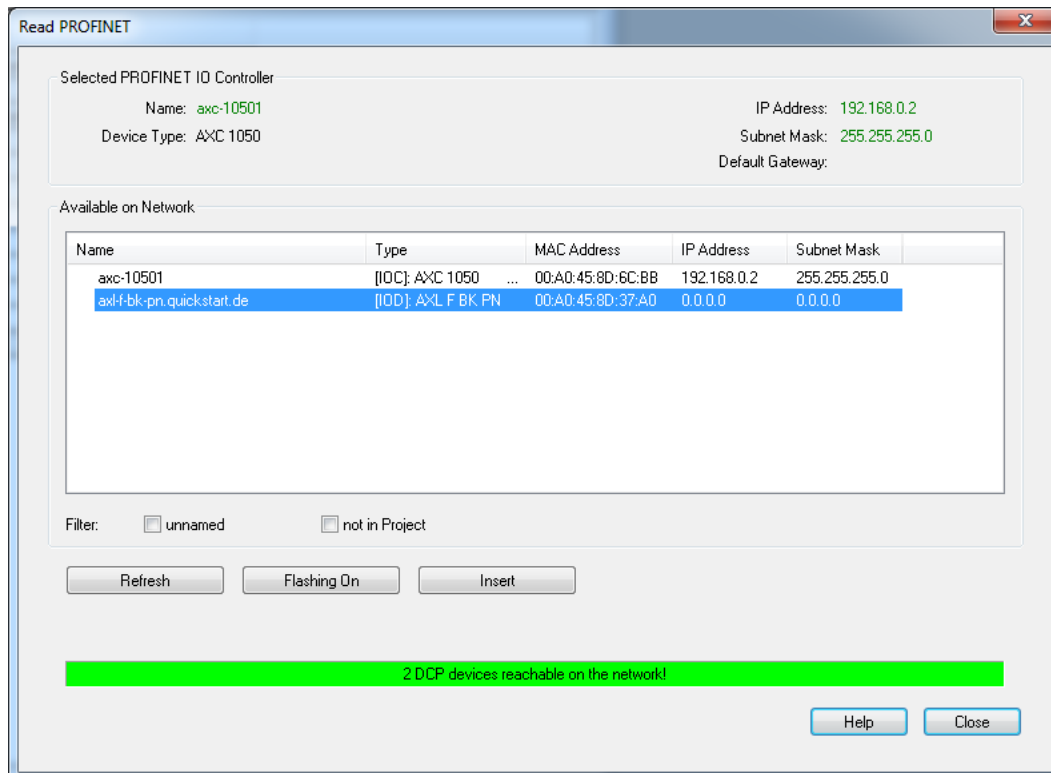


Figure 5-21 All devices with PROFINET device names

5.8 Checking/modifying the PROFINET settings for PROFINET devices

When reading in, the PROFINET settings of the relevant PROFINET devices are imported with default values. Check these settings and modify them, if necessary.



- Make sure you are in the bus configuration workspace.
- In the “Bus Structure” window, select the PROFINET device (here: AX-L F BK PN bus coupler).
- Select the “PROFINET Settings” tab in the “Device Details” window.
- Check the PROFINET settings and modify them, if necessary.

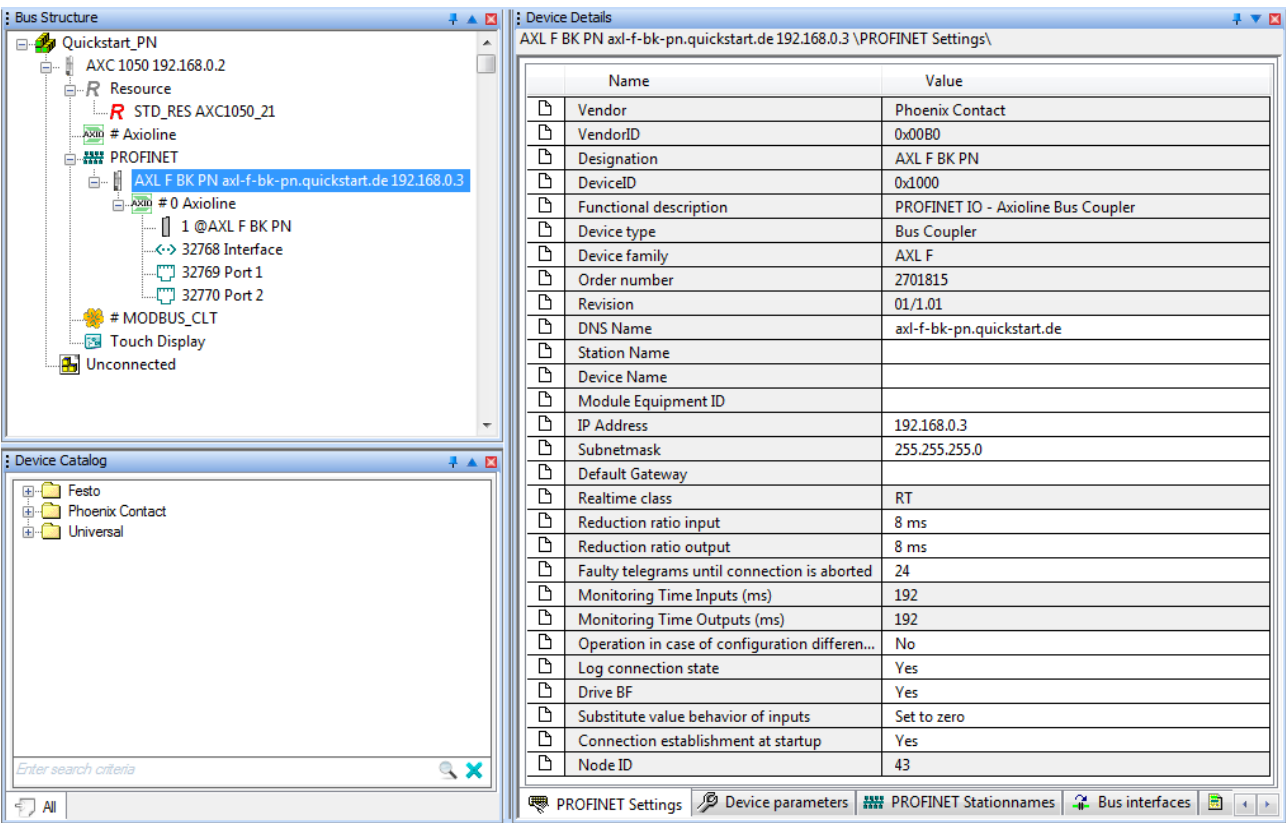


Figure 5-22 PROFINET settings of the “AXL F BK PN” bus coupler

The PROFINET settings comprise:

DNS name

This name is the unique identification for the PROFINET device in the network. It must be known to the PROFINET device before it can be used in the network (see also page 49).



If you modify the “DNS Name” PROFINET device name later, you will also have to adapt the DNS name of the PROFINET device accordingly.

IP address

During operation, the device can be accessed via the IP address. PC Worx selects the address from the area that is set on the project node.



If the area for the IP addresses is later modified in the project node, you will also have to adapt the addresses of the PROFINET devices accordingly.

Subnet mask

The subnet mask that was specified on the project node is assigned to each PROFINET device as the default subnet mask. It can be modified specifically for each individual device.

Default gateway

If a default gateway is used, enter its IP address here. The default gateway specified on the project node is automatically assigned to each PROFINET device.



“IP Address”, “Subnetmask”, and “Default Gateway” represent the **IP parameters** for a device.

**Reduction ratio input/
Reduction ratio output**

The time in which the inputs/outputs of the PROFINET device are updated. It can be set individually for each PROFINET device for both data directions.

The reduction ratio (update time) directly affects the number of PROFINET devices that can be operated via a PROFINET controller. The default setting displayed in PC Worx is the value that was stored in the FDCML file for the corresponding PROFINET device. If this setting exceeds the performance level of the PROFINET controller, a corresponding error message appears when translating the project. In this case, select a longer update time for certain devices.

**Monitoring time inputs
(ms)**

Period during which the PROFINET controller waits for data from the PROFINET device. If the PROFINET device does not send data to the PROFINET controller within this period, the connection between the PROFINET controller and the PROFINET device is terminated.

**Monitoring time outputs
(ms)**

Setting this time decides how long the PROFINET devices cannot receive valid PROFINET telegrams before substitute values are written to the outputs. This setting can be used to influence the error insensitivity of the network.

**NOTE: Errors within the process**

Please note that excessively high settings can result in errors in the process as the PROFINET device “freezes” the last output value until the error is detected.

**Operation in case of con-
figuration differences**

This setting affects the startup behavior of the controller for devices that return a “Module-DiffBlock” information to the controller in the PROFINET parameterization sequence. The following settings can be made:

- | | |
|-----|--|
| No | Default setting; the connection to the device is disconnected again in the event of configuration differences. <ul style="list-style-type: none"> • Start parameterization again via a download to set the device to the operational state. |
| Yes | The connection is maintained even in the event of configuration differences. |

Log connection state

This setting affects connection logging in the controller for each device. The following settings can be made:

- | | |
|-----|--|
| Yes | Default setting; a message is written to the diagnostic archive of the controller upon every connection abortion and connection reestablishment. |
| No | Logging is prohibited because, for example, switching off a device is an application-relevant situation. |

Drive BF

This setting defines whether the BF LED and the PNIO_SYSTEM_BF system variable on the controller are to be controlled in the event of a missing connection to the selected device. The following settings can be made:

- | | |
|-----|---|
| Yes | Default setting; each aborted connection is indicated by the BF LED and by setting the system variable to TRUE. |
| No | Indication is prohibited because, for example, switching off a device is an application-relevant situation. |

5.9 Transferring PROFINET device names and IP settings to PROFINET devices

Before a PROFINET device can be operated in a PROFINET network, the PROFINET device name (“DNS Name”) and IP address configured in PC Worx must also be made known to the device itself.

- Make sure that the PROFINET controller can establish communication to the PROFINET devices.
- Compile the project, send it to the controller, and perform a cold restart. Proceed as described in “Compiling and sending a project and performing a cold restart” on page 30.

During startup, the AXC 1050 automatically assigns the IP settings and device parameterizations specified in the project to the PROFINET devices.

- To check the currently assigned device names and IP settings, select the “PROFINET Stationnames” tab in the “Device Details” window.

The updated list contains the PROFINET devices together with their PROFINET device names and the IP parameters.

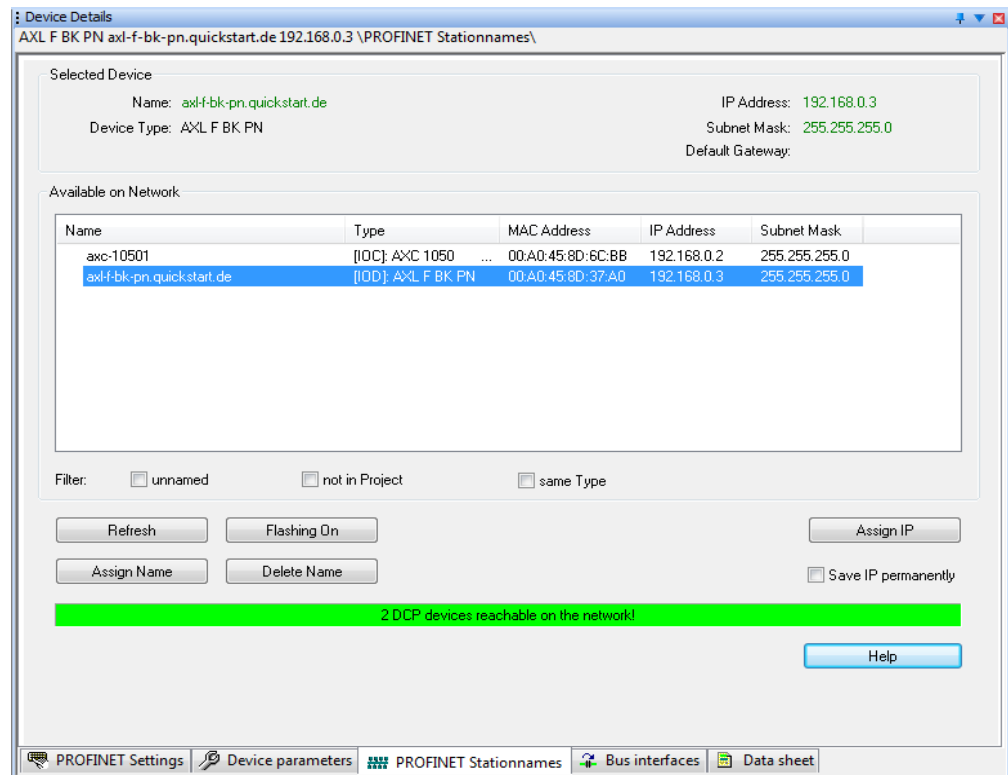


Figure 5-23 All PROFINET devices with PROFINET device name and IP address

When the “PROFINET Stationnames” tab is selected, the list is updated. If you wish to update at a later point in time, click on the “Refresh” button.

The BF LED is now off on all PROFINET devices.

The PROFINET network is running.

5.10 Reading in Axioline devices

The process for reading in Axioline F modules that are connected to a PROFINET controller or a PROFINET device is the same.

The Axioline F modules connected to a **PROFINET controller** can be read once the communication path to the PROFINET controller has been established.



In the example configuration (see Figure 2-3) no further modules are connected to the AXC 1050.

Axioline F modules that are connected to PROFINET devices can be read once the device names and IP addresses have been specified for the PROFINET devices. In the example project, Axioline F modules are connected to the AXL F BK PN bus coupler.

How to proceed

- In the “Bus Structure” window, select the “Axioline” node **below** the bus coupler.
- Right-click to open the context menu and select the “Read Axioline” menu item.

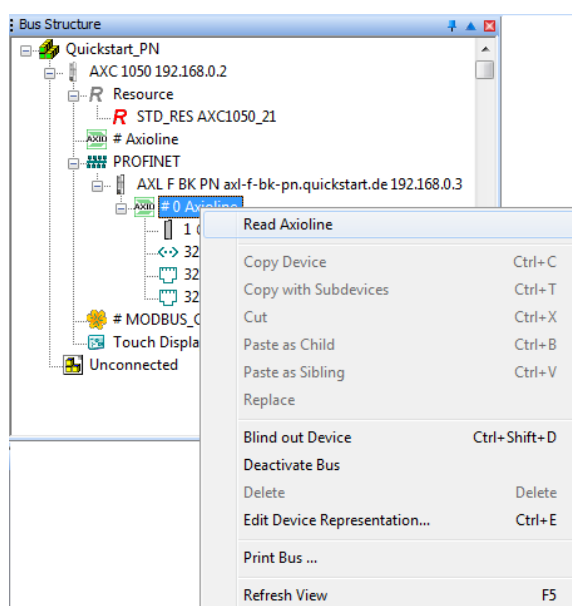


Figure 5-24 “Read Axioline” context menu

All Axioline F modules connected to the PROFINET device (here: to the bus coupler) are displayed in the “Read Axioline” dialog box that opens. Modules marked in black have not yet been read into the PC Worx project; modules marked in green have already been configured. When the “Re-insert all modules” option is selected, modules that have already been configured are also read in again. The previously configured bus configuration is overwritten.

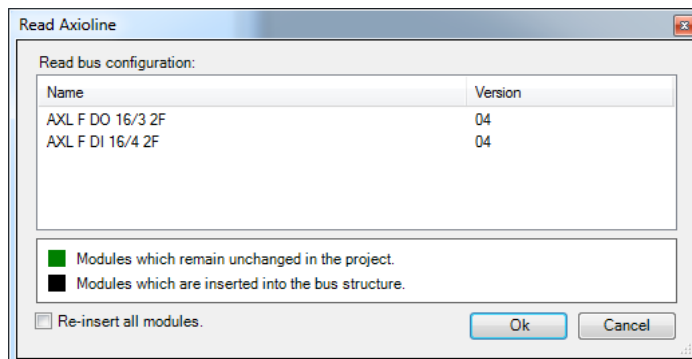


Figure 5-25 Reading in connected Axioline F modules: Axioline F modules that have not yet been read in

- Click on “OK” to read in the Axioline F modules.

The read in Axioline F modules are read in with their default configuration and are displayed in the “Bus Structure” window. If you select a device, the device details will be displayed.

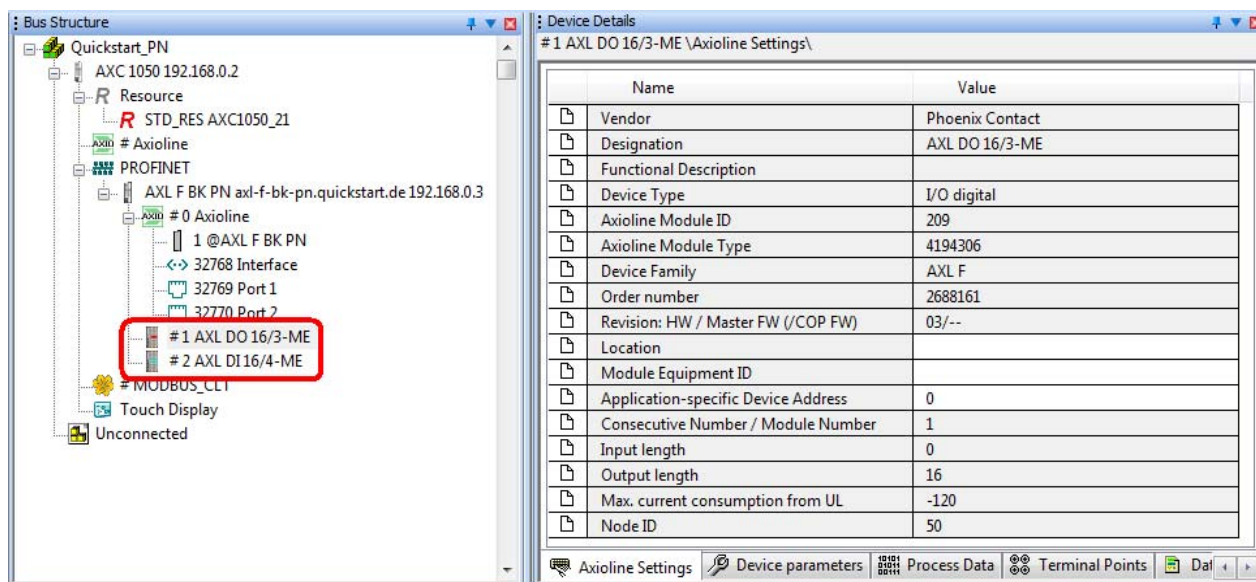


Figure 5-26 Read in Axioline F modules

5.11 Compiling and sending a project, and performing a cold restart after reading the bus topology

The bus configuration is now completely included in the project. At this point you can compile your project in order to detect any errors that may have occurred.

- In order to start up the project, compile it, send it to the controller, and perform a cold restart. Proceed as described in Section 4.7, “Compiling and sending a project and performing a cold restart”.

The AXC 1050 controller and the connected Axioline F modules are now started up. The D LED on the Axioline F modules is permanently on. The Axioline F local bus is running.

5.12 Creating the program

After the bus configuration has been read in completely, the program can be created.

You can

- Create a project yourself according to your requirements (see quick start guide for PC Worx (UM QS EN PC WORX))
- Or
- Insert the “Moving_Light” example program (included in the starter kit) in the project and test the example program.

How to insert and test the “Moving_Light” example program is described in the following.

5.12.1 “Moving_Light” program description

The “Moving_Light” example program is created using the function block diagram (FBD) programming language.

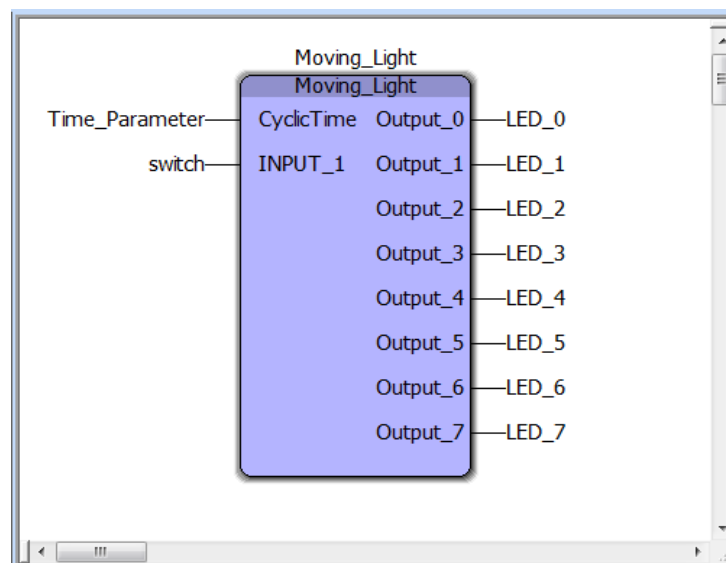


Figure 5-27 “Moving_Light” program code

Figure 5-27 shows the program code of the example project.

In the example program, a running light for the LEDs of digital outputs 1 to 8 of the AXL F DO16/3 2F Axioline F output module is programmed. The “Moving_Light” function block is used for this purpose. The running light is active if a 1 signal is present at input 1 of the AXL F DI16/4 2F Axioline F digital input module.

Variables in the program

The **switch** variable maps the status of input 1 of the AXL F DI16/4 2F Axioline F digital input module. Variables **LED_1** to **LED_8** map the states of outputs 1 to 8 of the AXL F DO16/3 2F Axioline F digital output module.

If the **switch** variable is TRUE, outputs 1 to 8 are connected one after the other for 200 ms each (initial value of variable **Time_Parameter**), and the corresponding LEDs light up. The running light is active as long as the **switch** variable is TRUE.

Visualization

The “FirstSteps_AXC1050_PN_Starterkit.prj” example visualization can be used to display the states of the inputs and outputs and of the LEDs in your web browser.

The value of the **Time_Parameter** variable (initial value: 200 ms) can be modified in the input field. The “Start/Stop” button is used to set the value for the **switch** variable and to start or stop the running light.



For more information about installing and starting up the example visualization on the controller, and the requirements for your web browser, refer to Section 7.

5.12.2 Extracting the “Moving_Light” function block library

The “Moving_Light” function block used in the example program is part of the function block library of the same name. It is available in compressed format as the “Bibliothek_Moving_Light.zip” file on the CD AXC 1050 PN STARTERKIT.

- Save the compressed “Bibliothek_Moving_Light.zip” file from the CD AXC 1050 PN STARTERKIT to the hard disk on your PC.
- Extract the file to the default directory for PC Worx libraries (“**Libraries\Documents\PC WORX\Libraries**”) on the hard disk of your PC.

5.12.3 Importing and compiling the “Moving_Light” function block library in PC Worx

Before the “Moving_Light” function block can be used, the function block library must be imported into PC Worx.



- Switch to the “IEC Programming” workspace.
- In the “Project Tree Window”, right-click to open the context menu for the “Libraries”.
- Select the “Insert... User Library...” commands.

Importing the function block library

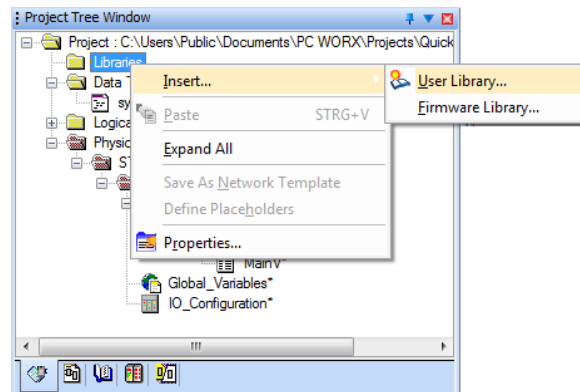


Figure 5-28 “Insert... User Library...” context menu

- Select the “Moving_Light.mwt” file from the folder of the previously extracted “Moving_Light” function block library.
- Then click on “Include”.

The function block library is now included in the “Project Tree Window” below the “Libraries” item.

Before the function block library can be used, it must be compiled.

- Right-click to open the context menu for the “Moving_Light” function block library.
- Select the “Open library as project” command.

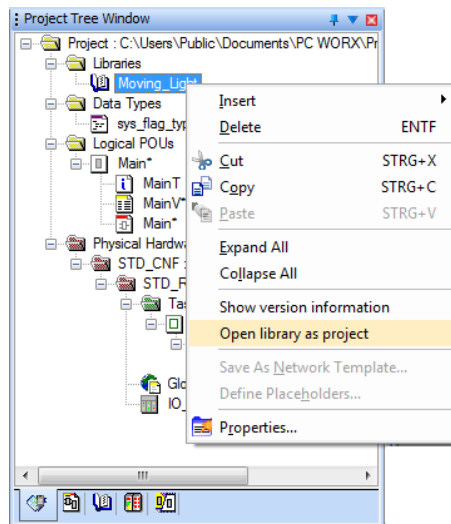


Figure 5-29 Context menu of the function block library: “Open library as project”

The function block library is opened as the PC Worx project “Moving_Light”.



- Select the “Build, Rebuild Project” command.
- Close the “Moving_Light” PC Worx project after compiling is complete.
- Open the “Quickstart_PN” PC Worx project again.

The “Moving_Light” function block library can now be used.

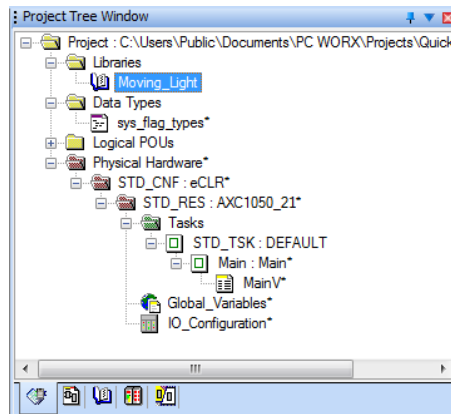


Figure 5-30 Inserted “Moving_Light” function block library

5.12.4 Programming

- Double-click on the “Main” in the “Project Tree Window” to activate the IEC programming interface.

Inserting the function block into the worksheet

- In the “Main” worksheet, click on an empty position where you would like to insert the function block.

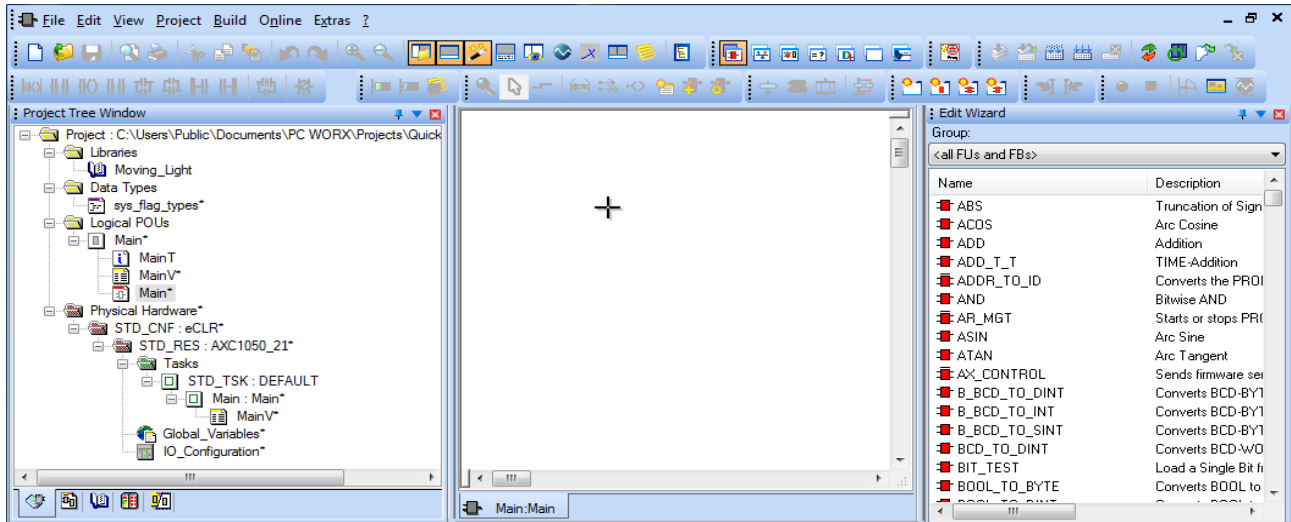


Figure 5-31 “IEC Programming” workspace, empty position on the “Main” worksheet selected

- In the “Edit Wizard” window, double-click on the “Moving_Light” function block.
- Enter the name “Moving_Light” in the “Variable Properties” window and confirm your entry with “OK”.

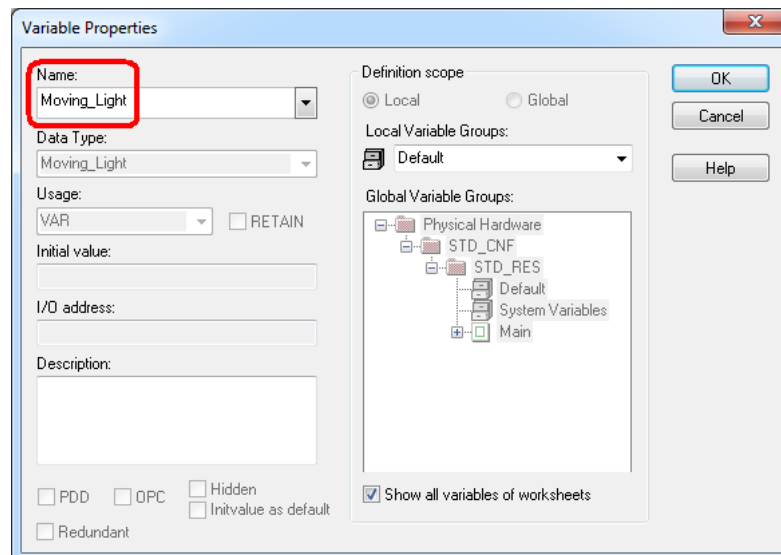


Figure 5-32 “Moving_Light” function block: “Variable Properties” window

The “Moving_Light” function block is inserted in the “Main” worksheet.

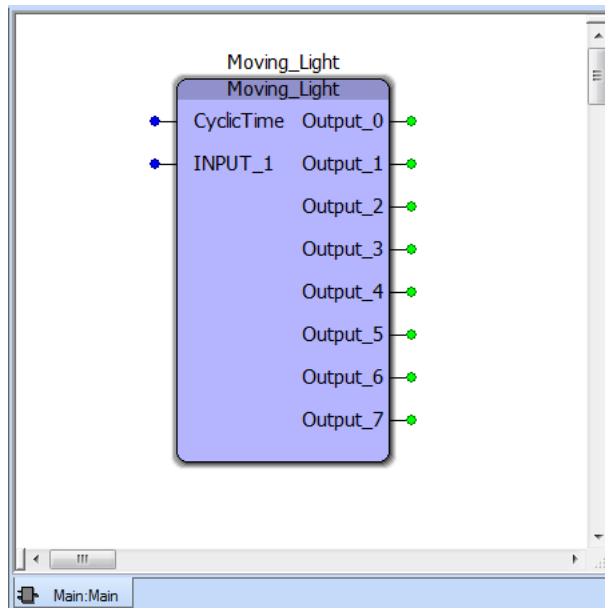


Figure 5-33 “Moving_Light” function block inserted in the “Main” worksheet

Creating variables

After the function block has been inserted in the worksheet, variables must be assigned to the inputs and outputs of the function block.

- Double-click on the “CycleTime” input.

The “Variable Properties” window opens.

- Enter the variable name “Time_Parameter” in the “Name” field.
- Enter the text “Time#200ms” in the “Initial value” field.
- Select the “PDD” checkbox.



When the “PDD” checkbox is activated, process data is transferred to visualization programs (WebVisit in the example). The process data is transferred using a .csv file.

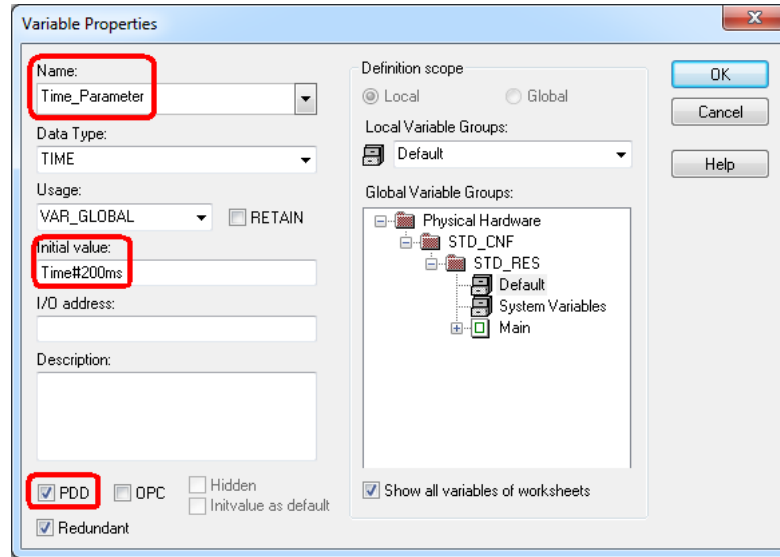


Figure 5-34 Properties of the **Time_Parameter** variable

- Confirm your entries with “OK”.

The **Time_Parameter** variable has been created and connected to the “CyclicTime” input (see Figure 5-35).

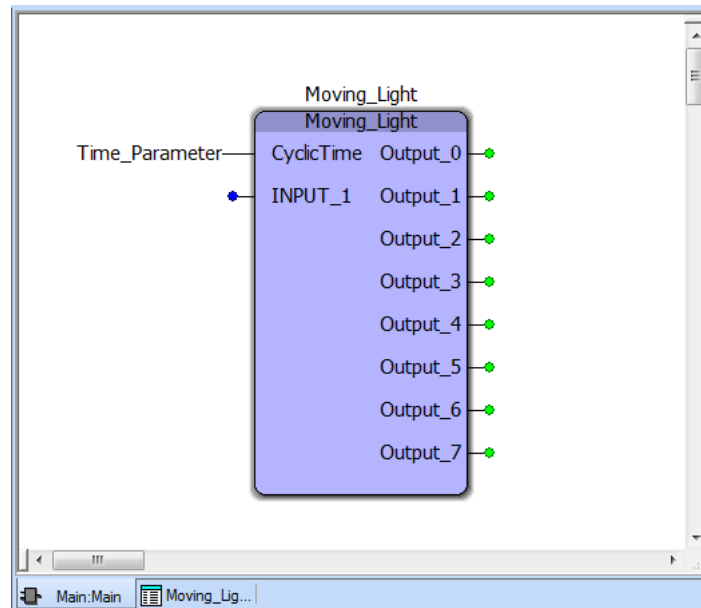


Figure 5-35 **Time_Parameter** variable at the “CyclicTime” input

- Create variables for all other inputs and outputs of the function block:

| Input/output | Variable |
|-----------------------|-----------------|
| INPUT_1 | switch |
| Output_0 ... output 7 | LED_0 ... LED 7 |

- Make sure that the “PDD” checkbox is activated for each variable when creating.

Figure 5-36 shows the function block after all the required variables have been created and connected to the corresponding inputs and outputs.

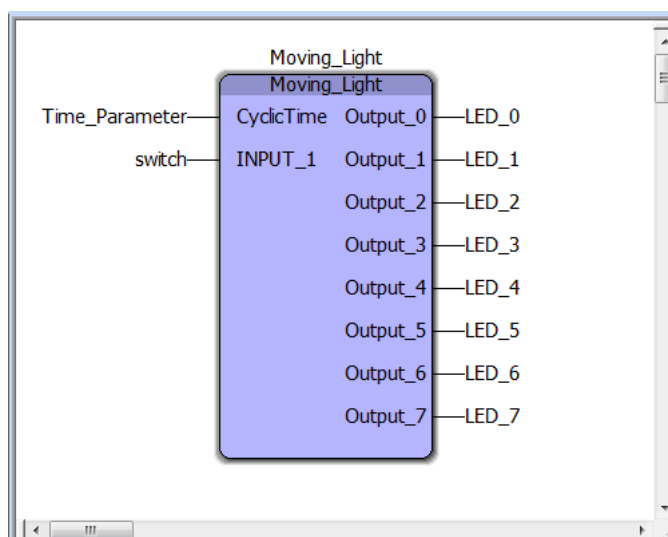


Figure 5-36 All the required variables have been created and connected to the corresponding inputs and outputs of the function block

5.12.5 Compiling after creating the program

At this point you can compile your project in order to detect any errors that may have occurred.



- Select the “Build, Rebuild Project” command.

5.13 Assigning process data

After all the required variables have been created, they must be assigned process data. It is assigned in the “Process Data Assignment” workspace.



- Switch to the “Process Data Assignment” workspace.
- Select the controller in the top right window. The standard configuration is then displayed in the top left window (“Symbols/Variables”).
- In the top left window (“Symbols/Variables”), select the “STD_RES : AXC 1050_21” resource.

- In the top right window, select the device/module for which you would like to link the process data to variables (here: AXL F DO16/3 2F).
- Select the process data item to be linked (here: process data item “OUT00”)
- Using drag and drop, link the selected variable to a shown variable (here: **LED_0**) on the left-hand side.

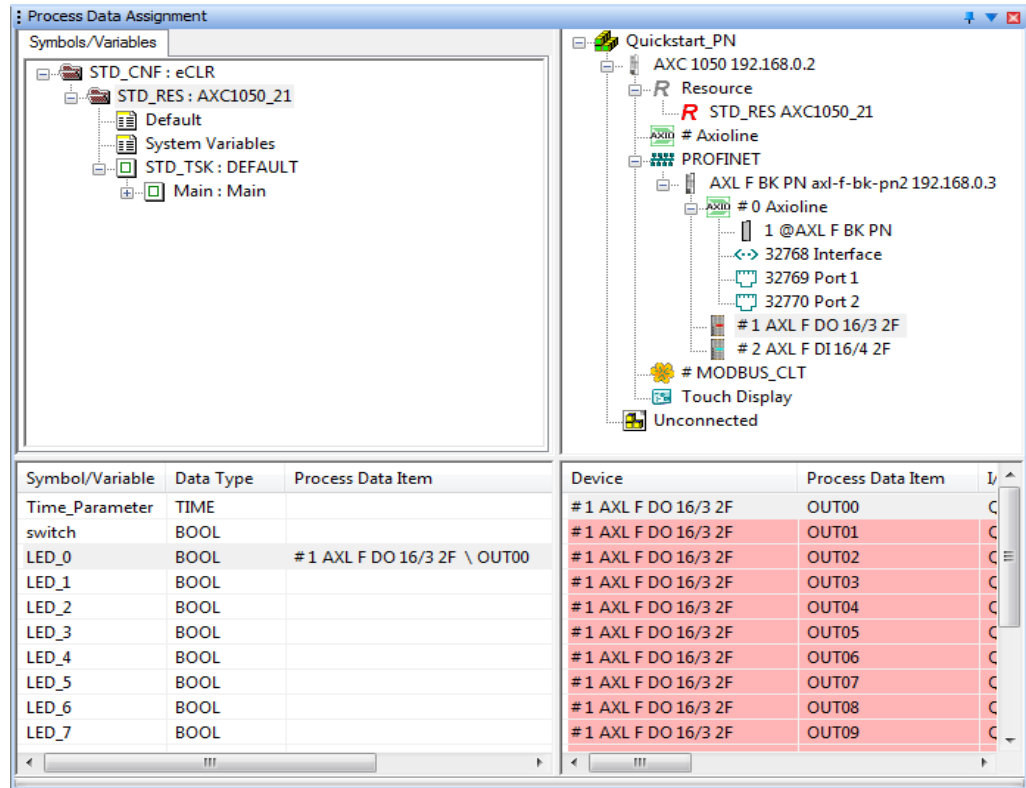


Figure 5-37 Process data item “OUT00” assigned to variable **LED_0**

- Repeat this procedure for all inputs to be evaluated and for all outputs to be controlled. For the example project, this means the following:
- Using drag and drop, link the process data “IN01” ... “IN07” to variables **LED_1 ... LED 7**.

The result of process data assignment is shown in Figure 5-38.

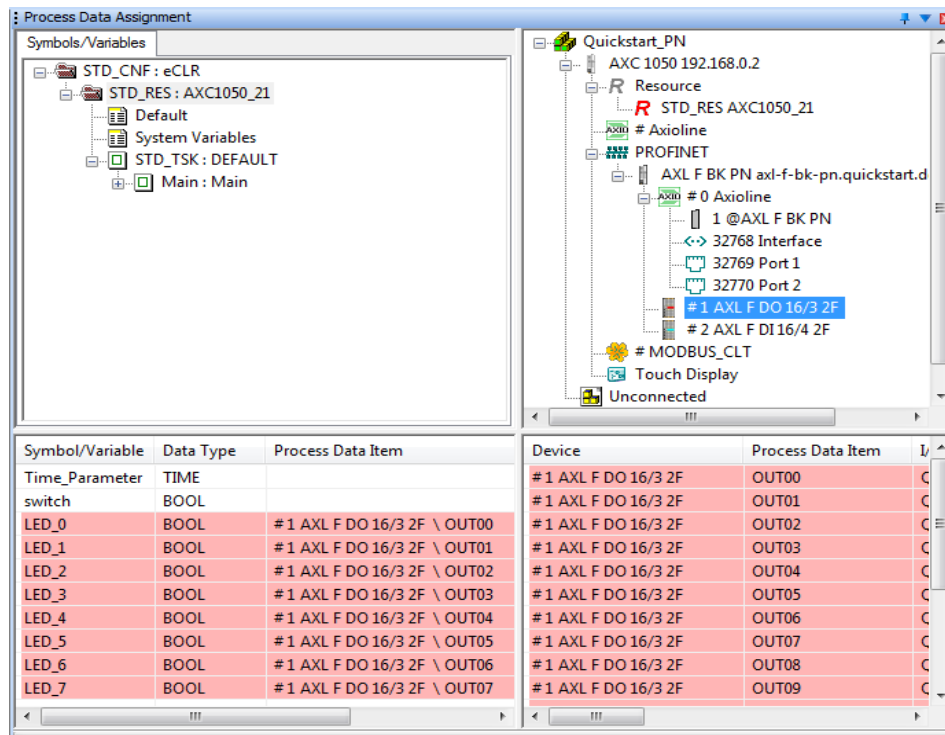


Figure 5-38 All process data used is assigned to variables

5.14 Compiling and sending a project (including program), and performing a cold restart

The project is now complete. In order to start up the project, compile it, send it to the controller, and perform a cold restart.

Compiling a project



- Select the “Build, Rebuild Project” command.

Downloading a project



- Open the “Project Control Dialog” dialog box.
- Click on “More...”.
- Activate the “Permanent as Bootproject” checkbox in the “Download options” dialog box.
- Activate the “Always allow real-time violations during Download Changes” checkbox in the “Download Options” dialog box.
- Close the “Download Options” dialog box.
- Click “Download” in the “Project Control Dialog” dialog box.

Performing a cold restart

- Click “Cold” in the “Project Control Dialog” dialog box.

5.15 Operation and debug mode – Viewing values on-line

While the program is running on the controller, it is possible to activate debug mode in PC Worx. In debug mode, the operation of the program can be monitored online.



- Activate debug mode.



To view values online in debug mode, the example program must be started first. To do this, proceed as follows:

- Switch to the “IEC Programming” workspace.
- Double-click on “Main” in the “Project Tree Window”.
- Right-click to open the context menu for the **switch** variable.
- Select the “Debug dialog” item.
- In the window that opens (“Debug: STD_RES”), activate the “TRUE” value.
- Click on the “Overwrite” button.

The example program is started.

The status of all global variables in the program is displayed under “MainV” in the “Project Tree Window”.

The screenshot shows the PC Worx software interface. On the left is the 'Project Tree Window' showing a hierarchical view of the project. The 'Main' folder is expanded, showing 'MainT', 'MainV', and 'Main'. The 'MainV' folder is selected. On the right is the 'Debug dialog' window, which displays a table of global variables and their online values.

| Name | Online value | Type | Usage |
|----------------|--------------|--------------|-------------|
| Default | | | |
| Moving_Light | | Moving_Light | VAR |
| Time_Parameter | 0.200 | TIME | VAR_EXTE... |
| switch | TRUE | BOOL | VAR_EXTE... |
| LED_0 | FALSE | BOOL | VAR_EXTE... |
| LED_1 | FALSE | BOOL | VAR_EXTE... |
| LED_2 | FALSE | BOOL | VAR_EXTE... |
| LED_3 | FALSE | BOOL | VAR_EXTE... |
| LED_4 | FALSE | BOOL | VAR_EXTE... |
| LED_5 | FALSE | BOOL | VAR_EXTE... |
| LED_6 | TRUE | BOOL | VAR_EXTE... |
| LED_7 | FALSE | BOOL | VAR_EXTE... |

Figure 5-39 Debug mode: “MainV”

The program status of a POU is displayed by opening the worksheet (in Figure 5-40 under “Main” in the “Project Tree Window” or under “Main:Main” on the tab in the workspace).

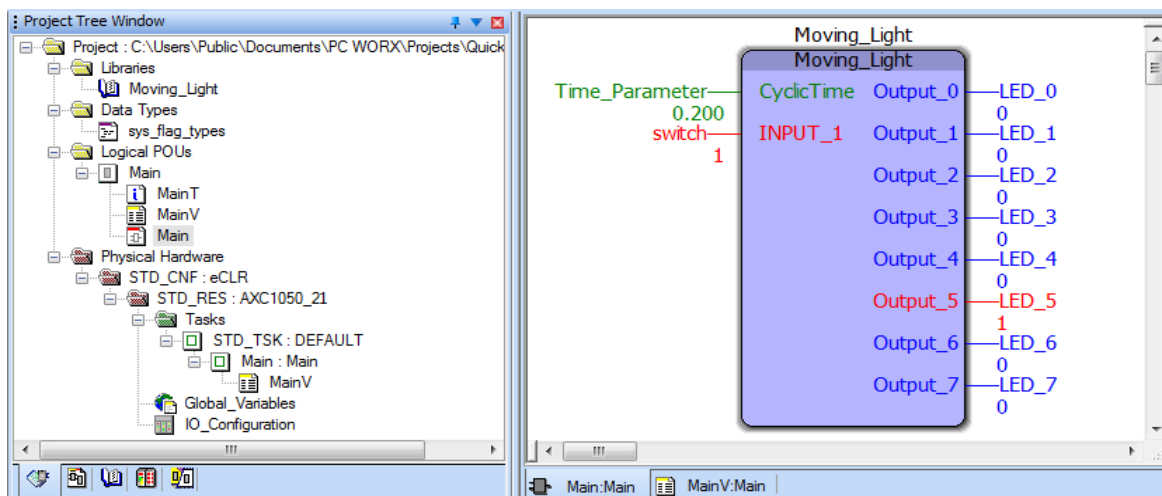


Figure 5-40 Debug mode: “Main”

The status of all global variables is displayed under “Global Variables” in the “Project Tree Window”.

| Name | Online value | Type | Usage | Description | Address |
|-------------------------|--------------|------|------------|--------------------|---------|
| Default | | | | | |
| Time_Parameter | 0.200 | TIME | VAR_GLOBAL | | |
| switch | TRUE | BOOL | VAR_GLOBAL | | |
| LED_0 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_1 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_2 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_3 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_4 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_5 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_6 | TRUE | BOOL | VAR_GLOBAL | | %QX6 |
| LED_7 | FALSE | BOOL | VAR_GLOBAL | | %QX6 |
| System Variables | | | | | |
| PLCMODE_ON | FALSE | BOOL | VAR_GLOBAL | PLC status ON | %MX1 |
| PLCMODE_LO... | FALSE | BOOL | VAR_GLOBAL | PLC status LOA... | %MX1 |
| PLCMODE_RUN | TRUE | BOOL | VAR_GLOBAL | PLC status RUN | %MX1 |
| PLCMODE_STOP | FALSE | BOOL | VAR_GLOBAL | PLC status STOP | %MX1 |
| PLCMODE_HALT | FALSE | BOOL | VAR_GLOBAL | PLC status HALT | %MX1 |
| PLCDEBUG_BP... | FALSE | BOOL | VAR_GLOBAL | Breakpoint set | %MX1 |
| PLCDEBUG_FO... | FALSE | BOOL | VAR_GLOBAL | Variable(s) forced | %MX1 |

Figure 5-41 Debug mode: “Global Variables”

- Switch to the window that provides the best display of the required information.



You can now view the relevant visualization (“First_Steps_AXC1050_PN_Starterkit.prj”) for the “Moving_Light” example program.

For more information about installing and starting up the example visualization on the controller, and the requirements for your web browser, refer to Section 7.

5.16 Example project on CD

The example project that has been developed in the previous sections is available in compressed format as the “Quickstart_PN.zwt” file on the CD AXC 1050 PN STARTERKIT.

If you would like to test the project without developing it yourself, you can use the “Quickstart_PN.zwt” file.

- Save the “Quickstart_PN.zwt” file on your PC, e.g., under the default path for PC Worx projects (“\Libraries\Documents\PC Worx\Projects”).
- Open PC Worx.
- Select “Open Project / Unzip Project...” under the “File” menu item.
- Confirm the prompt asking whether the project should be extracted to the directory containing the zwt file with “Yes” if the file is to be extracted to this directory. Or select a different directory. PC Worx extracts the file automatically.
- If PC Worx prompts you to overwrite layouts (e.g., page layouts) or bitmaps, confirm this prompt with “No to all”.

The “Moving_Light” function block used in the example program is part of the function block library of the same name. It is available in compressed format as the “Bibliothek_Moving_Light.zip” file on the CD AXC 1050 PN STARTERKIT.

- Save the compressed “Bibliothek_Moving_Light.zip” file from the CD AXC 1050 PN STARTERKIT to the hard disk on your PC.
- Extract the file to the default directory for PC Worx libraries (“\Libraries\Documents\PC WORX\Libraries”) on the hard disk of your PC.

Before the function block library can be used, it must be compiled. This can be done from the “Quickstart_PN” PC Worx project that is already open.

- Switch to the “IEC Programming” workspace.
- Right-click to open the context menu for the “Moving_Light” function block library.
- Select the “Open library as project” command (see Figure 5-29).
- Select the “Build, Rebuild Project” command.
- Close the “Moving_Light” PC Worx project after compiling is complete.
- Open the “Quickstart_PN” PC Worx project again.

Once PC Worx has unzipped the project, and the function block library has been extracted, saved, and compiled, the example project is available for you to view and edit.



To start up the project completely, the example bus configuration must be configured according to Section 2, “Installing the hardware”.

- Prepare your PC for communication (see Section “Preparing the PC for communication” on page 39).
- Set the IP address and the name specified in the project on the PROFINET controller (see “Assigning the IP address for the PROFINET controller” on page 41).
- Set the IP address and the name specified in the project on the AXL F BK PN controller: Select the AXL F BK PN bus coupler in the “Bus Structure” window. Select the “PROFINET Stationnames” tab.
- Select the “[IOD]: AXL F BK PN” bus coupler. Click on the “Assign IP” button. Click on the “Assign Name” button.
- Activate the PROFINET controller function (see Section “Activating the PROFINET controller function” on page 46).

- Compile the example project, send it to the controller, and perform a cold restart (see Section “Compiling and sending a project and performing a cold restart” on page 30).

The AXC 1050 controller and the connected Axioline F modules are now started up together with the program of the example project.

You can now switch to debug mode, for example, to view the variable states (Section “Operation and debug mode – Viewing values online” on page 66).

Alternatively, you can view the relevant visualization

(“First_Steps_AXC1050_PN_Starterkit.prj”) for the “Moving_Light” example program.

For more information about installing and starting up the example visualization on the controller, and the requirements for your web browser, refer to Section 7.

6 Additional functions of PC Worx

6.1 Changing the PROFINET device name (“DNS Name”)

When a PROFINET connection is established, a PROFINET device name cannot be assigned. Abort these connections first. To do this, proceed as follows:



- Open the “Project Control Dialog” dialog box.
- Click on “Stop”.
- Click the “Reset” button.

The BF LEDs on the devices now are off.

All Ethernet connections are aborted and the PROFINET device names can now be assigned.



- Make sure you are in the bus configuration workspace.
- In the bus configuration, select the PROFINET device which is to be assigned the name.
- Select the “PROFINET Settings” tab in the “Device Details” window.
- Change the device name under “DNS Name”.
- In the “Device Details” window, select the “PROFINET Stationnames” tab.

When the tab is selected, the list is updated. If you wish to update at a later point in time, click on the “Refresh” button.

In order to limit the search results, different options are available:

| | |
|----------------|---|
| No filter set | All PROFINET devices that are available in the network are listed. |
| unnamed | All the PROFINET devices that do not yet have a device name (“DNS Name”) are listed. |
| not in Project | All the PROFINET devices that are not included in the project are listed. |
| same Type | Only the PROFINET devices that are the same type as the PROFINET device selected in the bus configuration are listed. |

AXC 1050 PN STARTERKIT

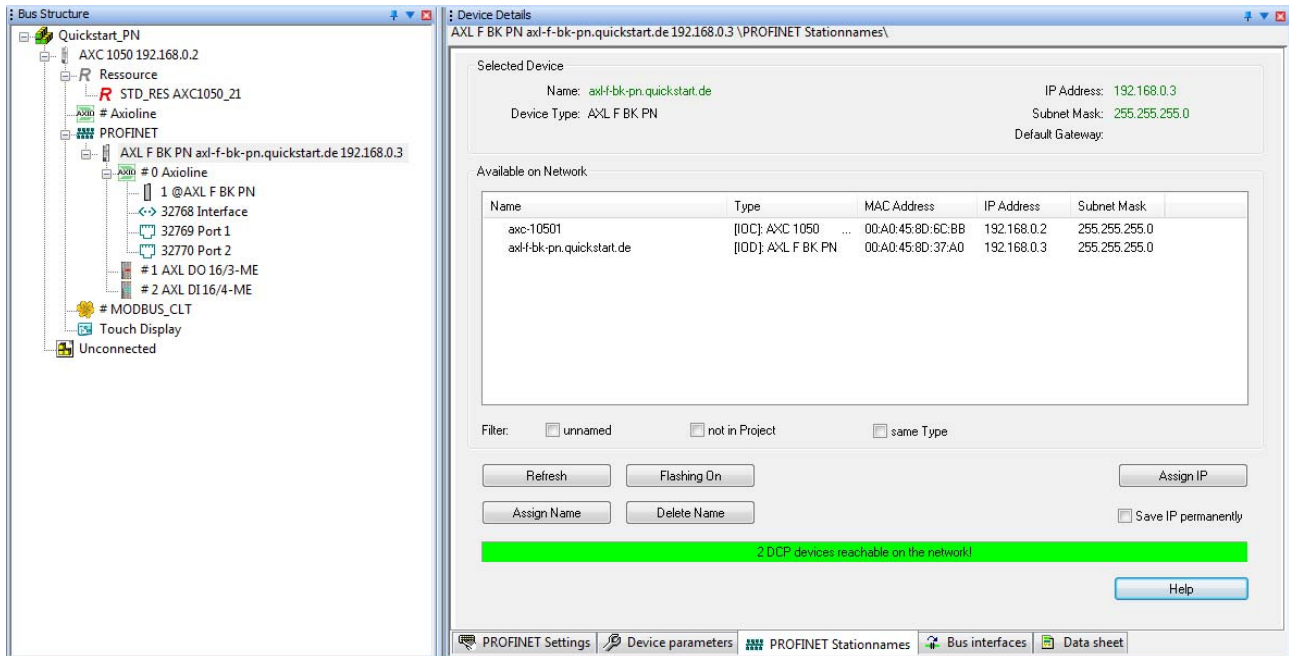


Figure 6-1 List of all PROFINET devices that can be accessed in the network

In the “Selected Device” area, the PROFINET device name specified under “IP Settings” is displayed.

From the list, select the corresponding PROFINET device (e.g., using the indicated MAC address).

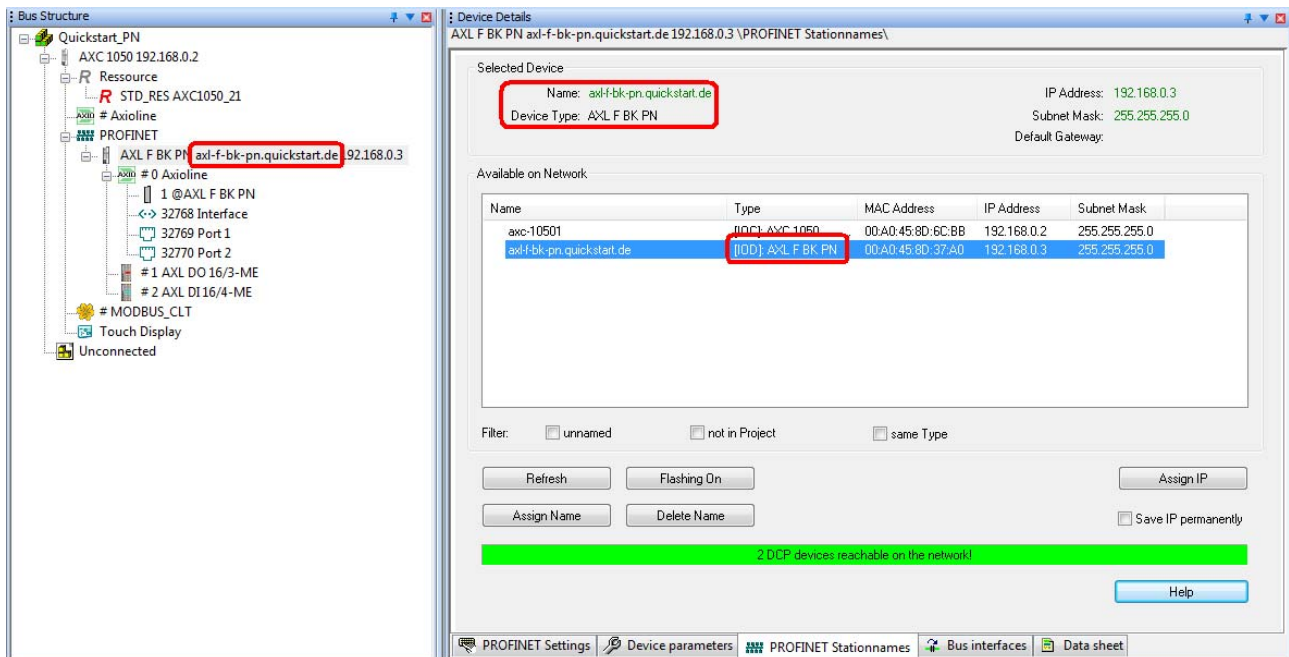


Figure 6-2 Selecting the PROFINET device

A flashing display can be triggered on the PROFINET device via the “Flashing On” button. This display can be used to find the selected PROFINET device in the field.

When the button is activated, the devices in the example system flash as follows:

AXL F BK PN Flashing LNK LED

The flashing process is also displayed in the list.

- Click on the “Assign Name” button to send the name to the PROFINET device.



In case an error message appears, first click on the “Delete Name” button. After that assign the name again.

The PROFINET device name (“DNS Name”) is transferred from the PC to the PROFINET device. The previous parameters are still displayed as the IP parameters. The IP parameters are not assigned until the next step (see Section 6.2).

- Assign the appropriate PROFINET device name to each PROFINET device.
- Compile the project, send it to the controller, and perform a cold restart (see Section 4.7, “Compiling and sending a project and performing a cold restart”).

6.2 Renumbering devices

There is also the option of automatically regenerating the IP parameters (e.g., IP addresses, Domain Postfix) for all devices. The settings from the project information are used for this.

- Select the project node.
- Select “Renumber devices, Whole project” from the context menu and then the menu item for renumbering the devices (e.g., “Append Domain Postfix to all DNS names”).

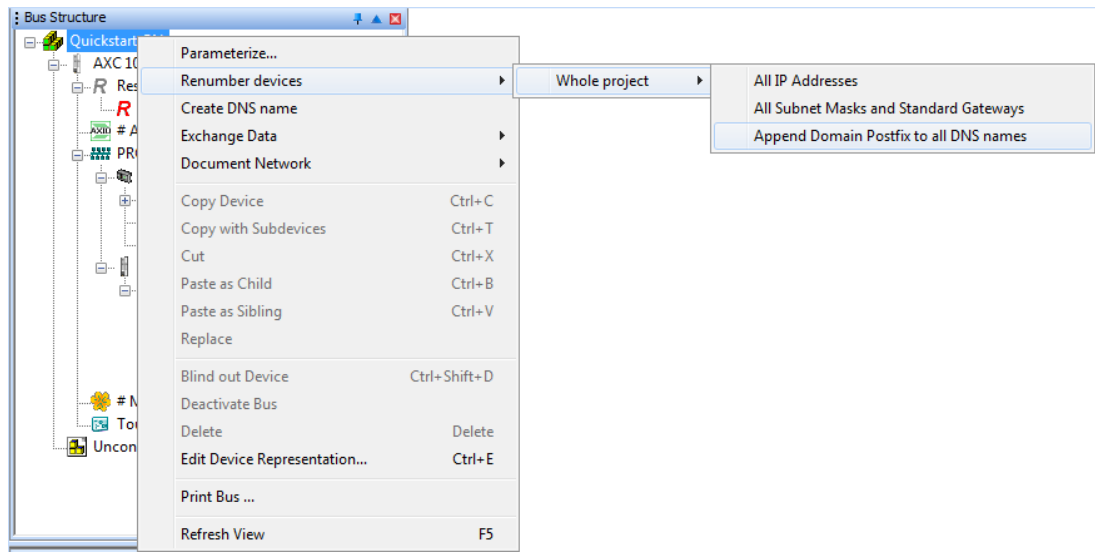


Figure 6-3 Renumbering devices

If, for example, the PROFINET device names were saved without Domain Postfix, the Domain Postfix specified in the project node (here: quickstart.de) will be added to the device names when renumbering. This is the complete PROFINET device name (“DNS Name”).

- To view the IP addresses in the bus configuration in ascending order, renumber the IP addresses as well.

All the PROFINET devices in the project now have the correct IP parameters. These parameters must now actually be assigned to the PROFINET devices.

- Compile the project, send it to the controller, and perform a cold restart (see Section 4.7, “Compiling and sending a project and performing a cold restart”).

6.3 Diagnostics with Diag+

The Diag+ diagnostic tool is used for consistent diagnostics of the Axioline F local bus and PROFINET.

When installing PC Worx, Diag+ is also installed automatically. Diag+ is integrated in PC Worx and can be called from PC Worx.

- Call Diag+ via the “View, Diag+” menu.

The “Diag+” dialog box opens.

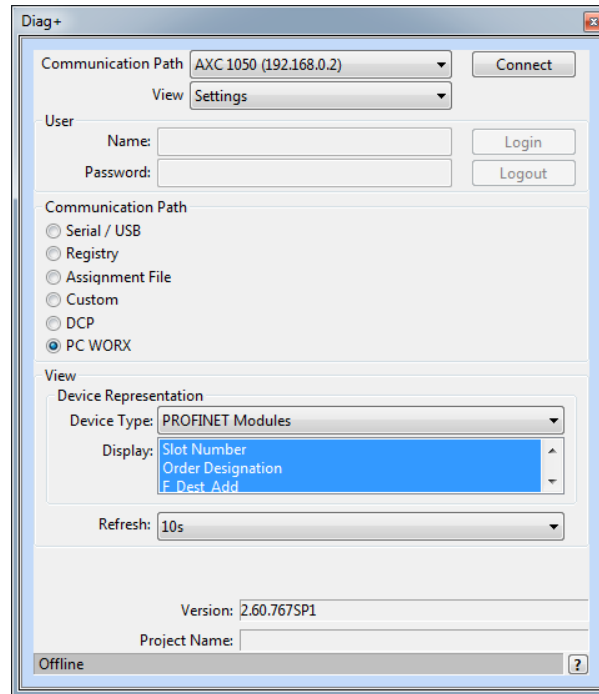


Figure 6-4 Diag+ start screen

The diagnostics represented depend on the communication path:

Table 6-1 Possible diagnostics

| PROFINET controller | PROFINET devices |
|------------------------------|----------------------|
| Diagnostic archive | Axioline diagnostics |
| PROFINET diagnostics | |
| Ethernet diagnostics | |
| Optical PROFINET diagnostics | |
| Axioline diagnostics | |

- Under “Communication Path”, select the device to which you wish to establish a connection. Select the AXL F BK PN bus coupler, for example.

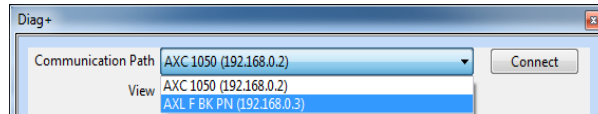


Figure 6-5 Diag+: selecting the communication path

- To activate the communication path, click on “Connect”.
- In the “View, Device Representations” area, select which information is to be represented for the devices. All information marked blue is displayed in the various views.

Device representations

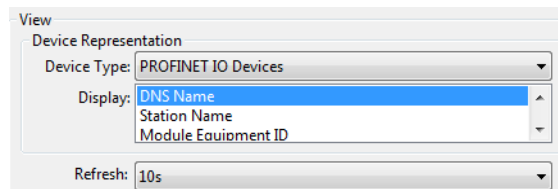


Figure 6-6 Diag+: example for device representation

- At the top under “View”, select which general information is to be represented (here: “Axioline Diagnostics”).

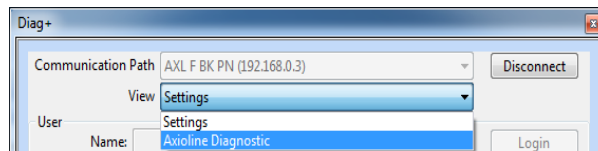


Figure 6-7 Diag+: selecting the information to be represented

View: Axioline diagnostics Diag+ switches to the diagnostic view of the connected Axioline F system configured in PC Worx.

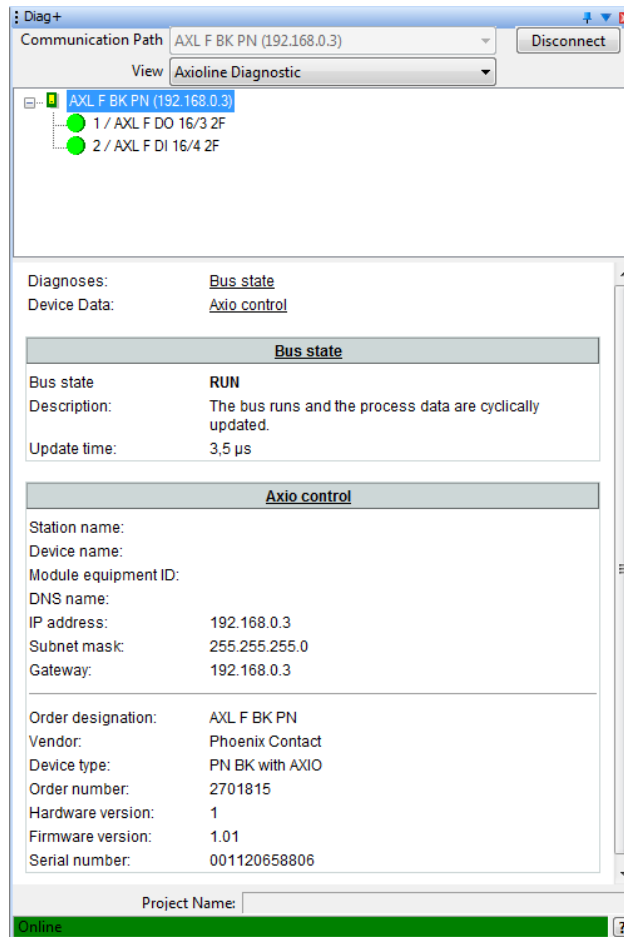


Figure 6-8 Axioline diagnostics: no error

Figure 6-8 shows diagnostics of an error-free Axioline F bus.

Any errors that occur are diagnosed by Diag+ and displayed. In the following example, the supply voltage for the AXL F DI16/4 2F Axioline F module is not present.

When the bus coupler is selected, Diag+ shows the following:

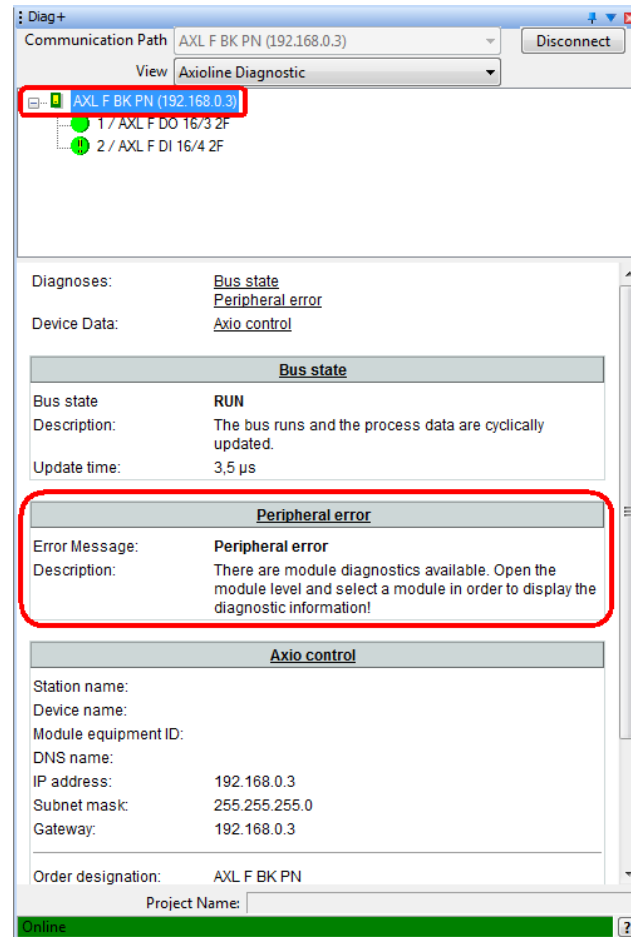


Figure 6-9 AXL F BK PN bus coupler: peripheral error

If you select the module on which the error has occurred, Diag+ displays error details:

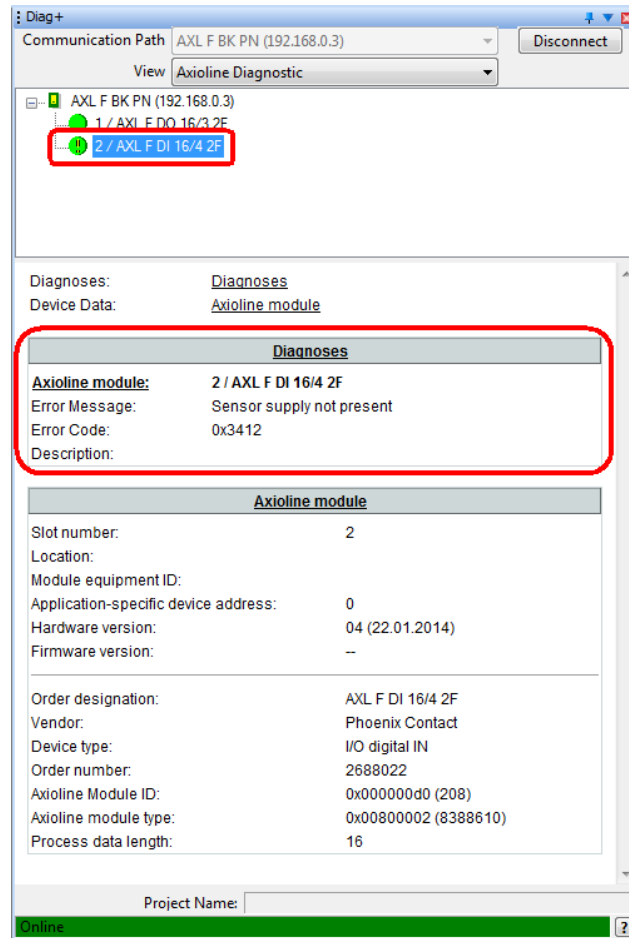


Figure 6-10 AXL F DI16/4 2F Axioline F module: supply voltage not present

Diagnostics can be used to identify errors in your application.

- Remove the error (here: connect the supply connector again).

Context menu of the PROFINET device in Diag+

The context menu of the PROFINET device can be used to influence the state of the Axioline F local bus.

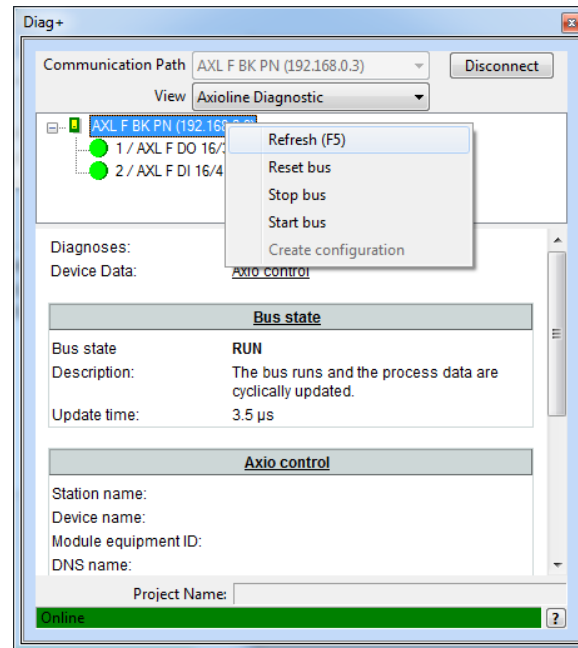


Figure 6-11 Context menu of the PROFINET device

The following menu items can be selected:

Refresh

Refreshes the view in Diag+.

Reset bus

Resets the local bus.



The process data is no longer updated.

Stop bus

Stops cyclic updating of the process data.



The process data is no longer updated.

Start bus

If a valid configuration frame has been generated, the local bus can be started using this menu item. The process data is updated cyclically after the local bus is started.



The process data is updated again.

Create configuration

Generates a valid bus configuration using the Axioline modules connected to the PROFINET device.

View: PROFINET diagnostics

- Select the AXC 1050 PROFINET controller as the “Communication Path”.
- To activate the communication path, click on “Connect”.
- At the top under “View”, select “PROFINET Diagnostic”.

The diagnostic information for the PROFINET devices is displayed.

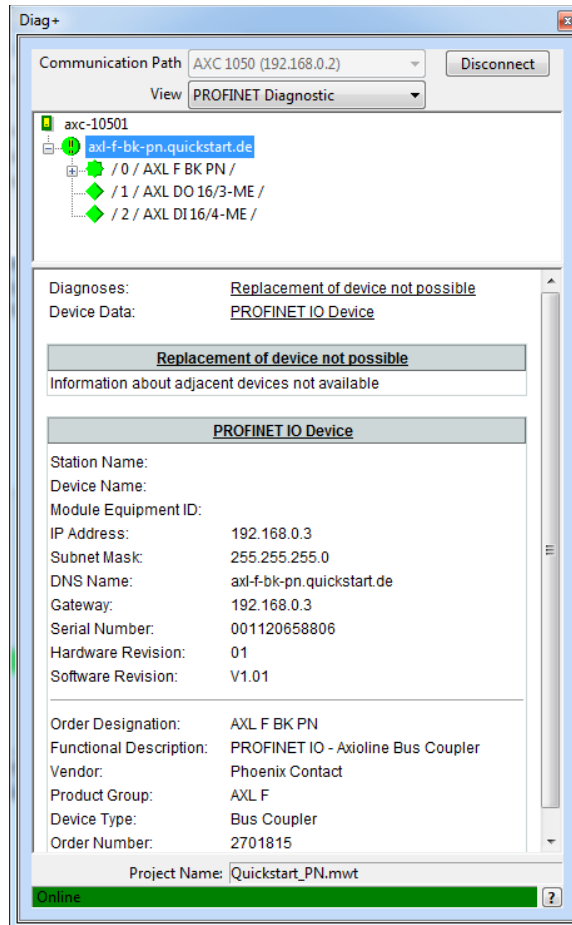


Figure 6-12 PROFINET diagnostics

6.4 Importing Ethernet devices into the project

Ethernet devices, such as the programming PC, can also be imported into your project. This provides you an overview of the complete network.

To import an Ethernet device into the project, proceed as follows:



- Make sure you are in the bus configuration workspace.
- Open the “Universal, Universal, IP” entry in the “Device Catalog”.
- If the “Generic IP Device” entry is not present, import this entry. To do this, proceed as follows:
 - Open the context menu in the device catalog.
 - Select the “Import Device...” command.

- In the "...fdcm10\ETHERNET\Universal" folder, open the "Generic_IP Device-xx.xml" file.

The file will be imported into the device catalog.

- Insert the "Generic IP Device" in the project.

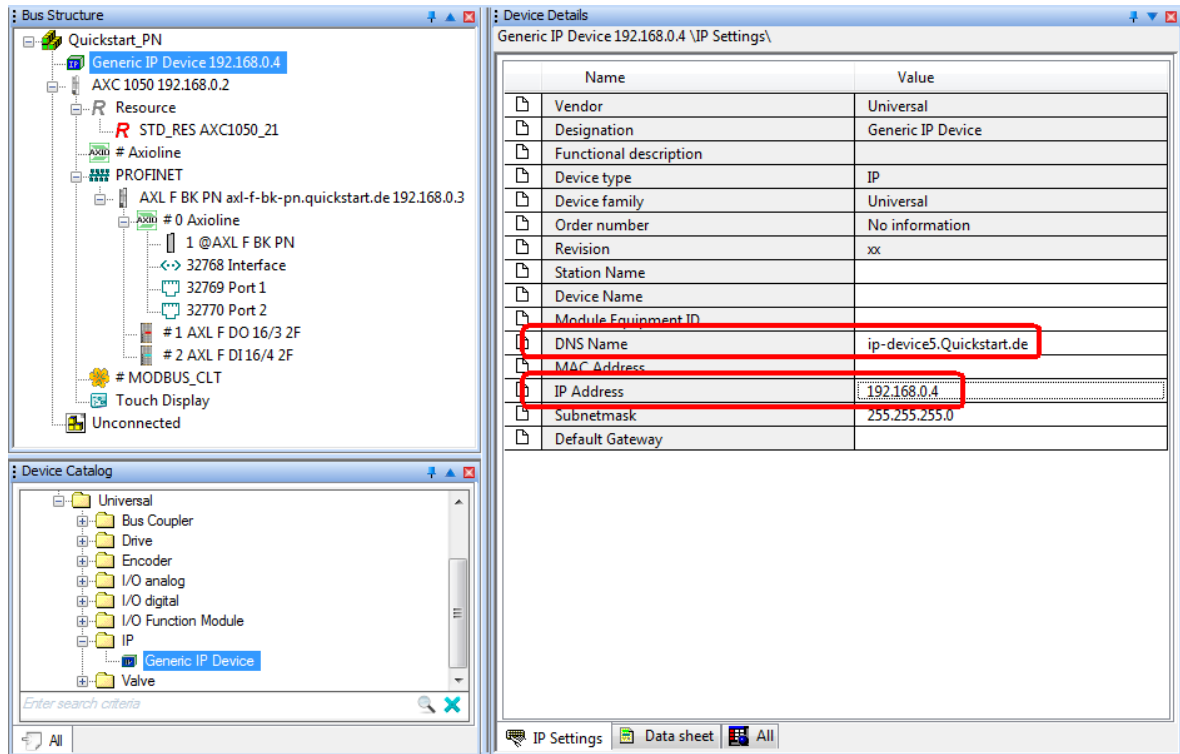


Figure 6-13 Inserting the "Generic IP Device"

The generic IP device is inserted together with a free IP address and a PROFINET device name.

- Adapt the IP address and the DNS name on the “IP Settings” tab.

The programming PC is to be inserted in the example. Its IP address is 192.168.0.100.

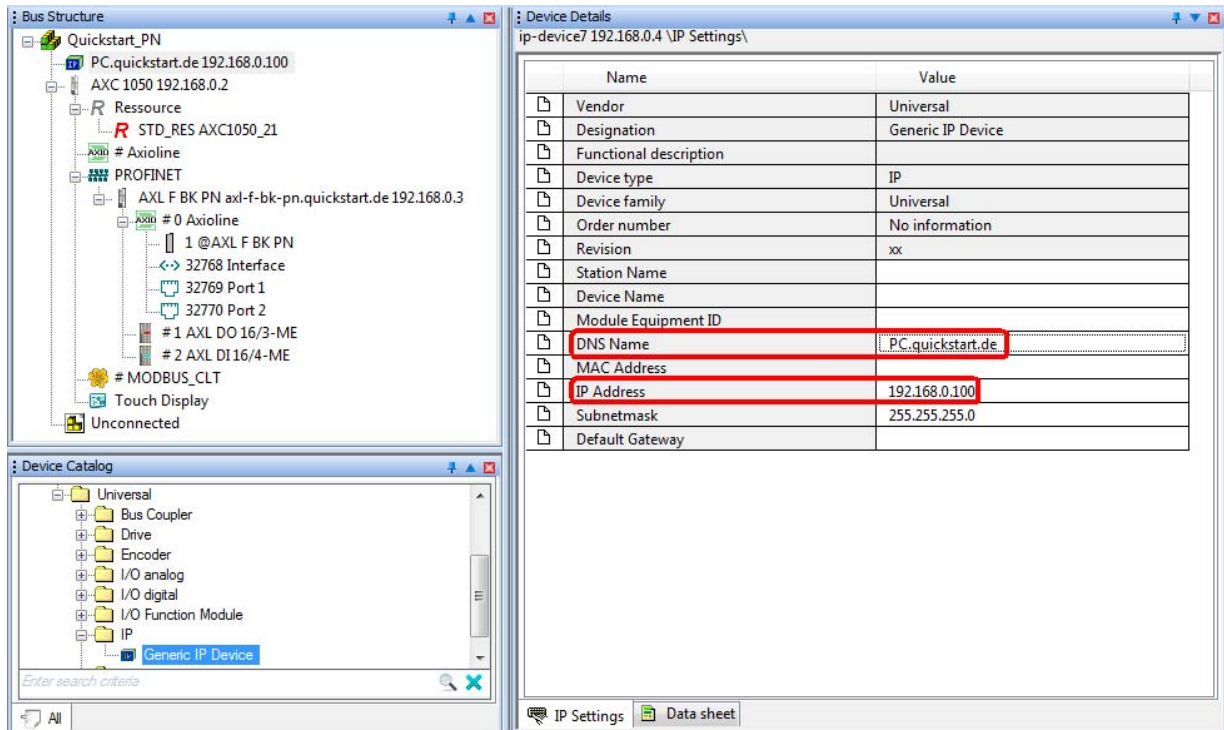


Figure 6-14 Adapting the IP address and PROFINET device name (“DNS Name”)

- Compile the project, send it to the controller, and perform a cold restart (see “Compiling and sending a project and performing a cold restart” on page 30).

6.5 Device description files

Device description files are FDCML or GSD files, which provide a complete description of a device. If the device catalog does not contain a corresponding device description file, it must be imported as described in the following sections.

6.5.1 Phoenix Contact device description files (FDCML files)

When PC Worx is installed for the first time, the device description files for the devices available at the time are also installed. These descriptions are also found in the device catalog. When installing add-ons for PC Worx, new device description files are installed in the “..\FDCML10\...\Phoenix Contact” folder. These new descriptions are not included in the device catalog yet. They must be imported as necessary.

- In PC Worx, select “Phoenix Contact” in the “Device Catalog” window.
- Right-click to open the context menu and select “Import Device...”.

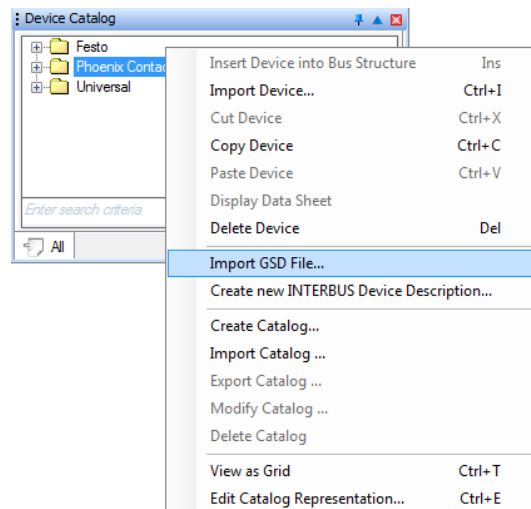


Figure 6-15 Device catalog: import device

- Select the device description file. If you have used the suggested standard installation, the file is located in directory:
C:\Users\Public\Documents\FDCML10\xxx\Phoenix Contact.
 xxx = system (e.g., Axioline, INTERBUS, PROFINET)

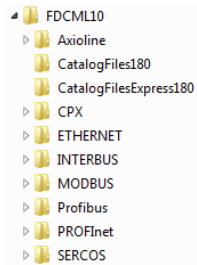


Figure 6-16 Directory for device description files

- Confirm your selection by clicking “Open”.

The “Message Window” indicates whether the device has been imported successfully.

6.5.2 Device description files from other manufacturers (GSD files)

GSD files can be used to integrate devices from manufacturers other than Phoenix Contact in PC Worx.

- Copy the GSD file for your device, e.g., from the Internet to your PC hard disk.
- In PC Worx, select “Phoenix Contact” in the “Device Catalog” window.
- Right-click to open the context menu and select “Import GSD File...”.

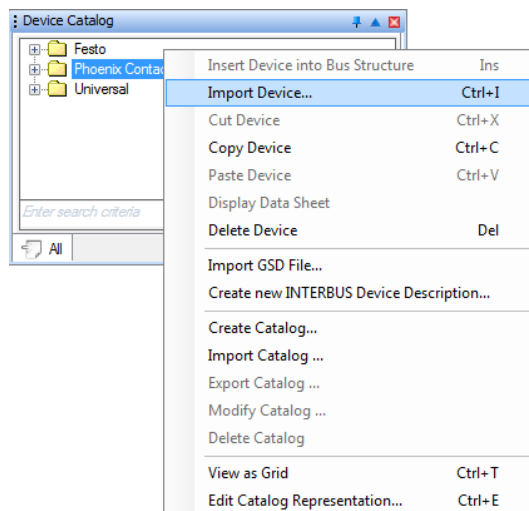


Figure 6-17 Device catalog: import GSD file

- Confirm your selection by clicking “Open”.

PC Worx creates an entry for the device in the device catalog. The entry can be found in directory **C:\Users\Public\Documents\FDCML10\...** (for standard installation).

The “Message Window” indicates whether the device has been imported successfully.

When reading a modular device, only the device, and not its modules, is displayed in the bus configuration.

Manually insert the modules for these devices. To do this, proceed as follows:

- Open the module catalog via “View, Module Catalog”.
- Select the device modules one at a time from the module catalog and drag and drop them in the lower level below the device entry.

7 Visualization with WebVisit

7.1 General

The WebVisit software is used to visualize variables of the controller used under PC Worx. WebVisit is a software tool used to generate web pages. The software runtime component is a web server, which is stored on the controller. The variable values are actually visualized via a Java-compatible standard browser.



For visualization, you need a web browser with Java Standard Edition SE 6 (or later) with at least Java Runtime Environment JRE 6 (Version 1.6.x or later).

Variables in PC Worx

To visualize variables from your PC Worx project in WebVisit, activate the “PDD” checkbox in PC Worx:

- In the “Variable Properties” window (Figure 7-1)
- On the worksheet (Figure 7-2)

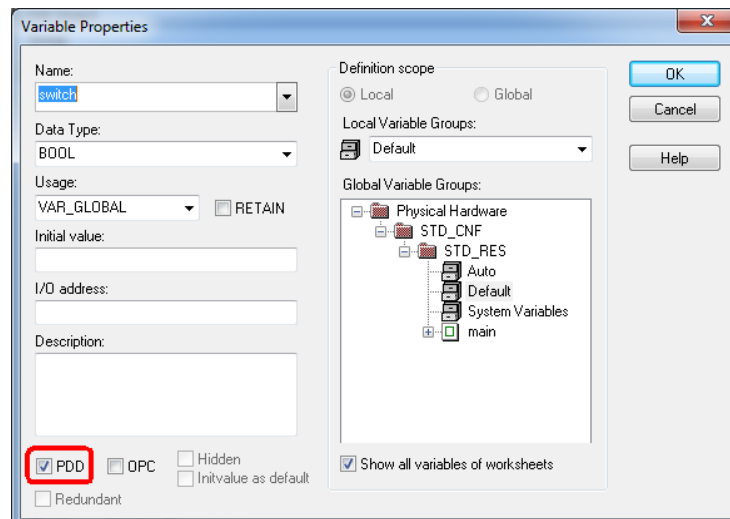


Figure 7-1 Activating the “PDD” checkbox in the “Variable Properties” window

| | Name | Type | Usage | Description | Address | Init | Retain | PDD | OPC |
|--|----------------|------|------------|-------------|---------|------------|--------|-------------------------------------|--------------------------|
| | Default | | | | | | | | |
| | Time_Parameter | TIME | VAR_GLOBAL | | | Time#200ms | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | switch | BOOL | VAR_GLOBAL | | | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_0 | BOOL | VAR_GLOBAL | | %QX6.0 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_1 | BOOL | VAR_GLOBAL | | %QX6.1 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_2 | BOOL | VAR_GLOBAL | | %QX6.2 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_3 | BOOL | VAR_GLOBAL | | %QX6.3 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_4 | BOOL | VAR_GLOBAL | | %QX6.4 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_5 | BOOL | VAR_GLOBAL | | %QX6.5 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_6 | BOOL | VAR_GLOBAL | | %QX6.6 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| | LED_7 | BOOL | VAR_GLOBAL | | %QX6.7 | | | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

Figure 7-2 Activating the “PDD” checkbox on the worksheet (Global_Variables)

When compiling the project in PC Worx, a file called “pdd.csv” is generated, which is used by WebVisit for visualization.

In WebVisit, enter the PC Worx project used (recommended) or the corresponding pdd.csv file. When carrying out standard installation (see Section 3) and naming the PC Worx project according to Section 5, you will find

- The project in directory
\\Libraries\Documents\PC Worx\Projects

And

- The associated pdd.csv file in directory
\\Libraries\Documents\PC Worx\xxx\C\STD_CNF\R\STD_RES
(xxx = project name; in the example: FirstSteps_AXC1050_PN_Starterkit).

When entering this path, the path for the pdd.csv file is updated automatically. Please also refer to Figure 7-3 on page 89.



For additional information on using WebVisit, please refer to the corresponding documentation.

7.2 Saving/extracting the visualization

On the CD AXC 1050 PN STARTERKIT, you will find the visualization of the example project from Section 5 as a WebVisit project in compressed format (“FirstSteps_AXC1050.zip” file).

- Save the compressed WebVisit project (“FirstSteps_AXC1050.zip”) from the CD AXC 1050 PN STARTERKIT to the hard disk of your PC.
- Extract the file to the default directory for WebVisit projects
C:\Program Files (x86)\Phoenix Contact\Software Suite 182\WebVisit ...\Projects on the hard disk of your PC.

7.3 WebVisit: path to the PC Worx project

For the variables from the PC Worx project to be visualized in WebVisit, the path to the PC Worx project and the path to the “pdd.csv” file must be specified in WebVisit.

- Open WebVisit.
- Select the “Open Project” command from the “File” menu.
- In the window that opens, navigate to the “FirstSteps_AXC1050_PN_Starterkit” WebVisit project:
C:\Program Files (x86)\Phoenix Contact\Software Suite 182\WebVisit ...\Projects\FirstSteps_AXC_1050.
- Select the “FirstSteps_AXC1050_PN_Starterkit.prj” file.
- Then click “Open”.

The WebVisit project is opened.

- Select the “Project Configurations” command from the “Project” menu.
- Switch to the “Project Advanced” tab.
- Enter the path to the PC Worx under “PPO Variable List, *.mwt, *.mwe Path”.

The path to the “pdd.csv” file in the “PPO List pdd.csv Path” field is added automatically.

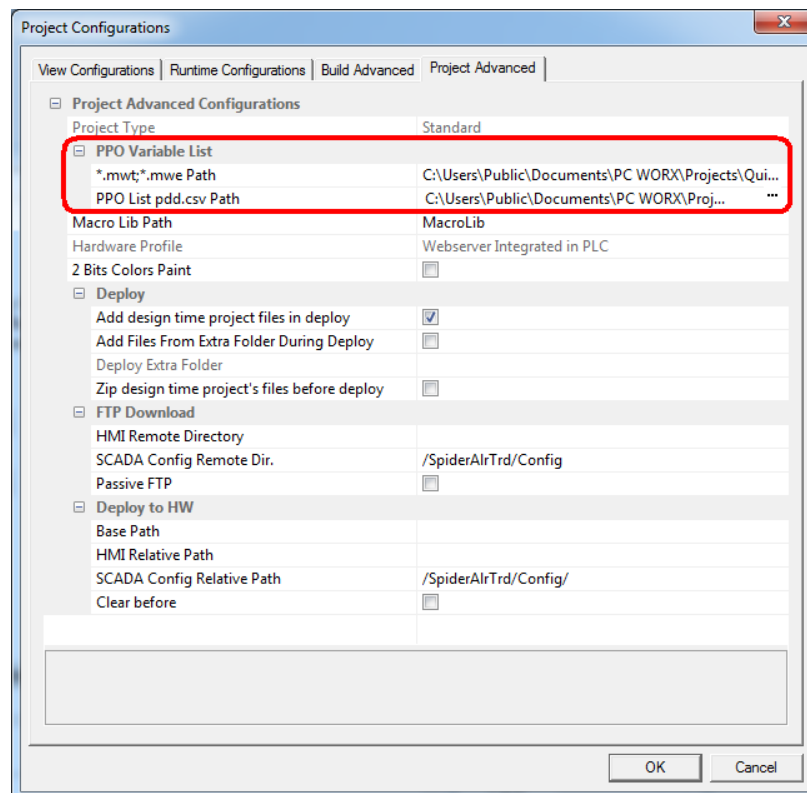


Figure 7-3 PC Worx example project: setting the path

- Confirm your entries with “OK”.

7.4 WebVisit: downloading the visualization to the controller

- Select the “Download Project” command from the “Project” menu.
- In the dialog box that opens, confirm the question “Would you like to make a BuildAll before deploying the HMI Project” with “Yes”.
- In the “Connect” area, set the IP address of the controller under “Server” (“192.168.0.2” in the example, see Figure 7-4).

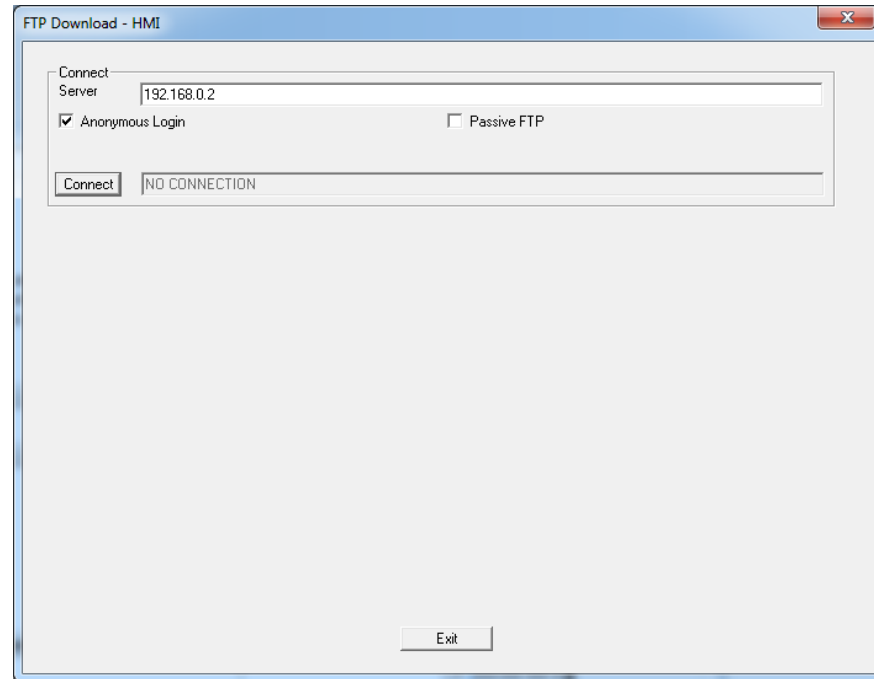


Figure 7-4 WebVisit: FTP download to the controller

- Click on the “Connect” button.

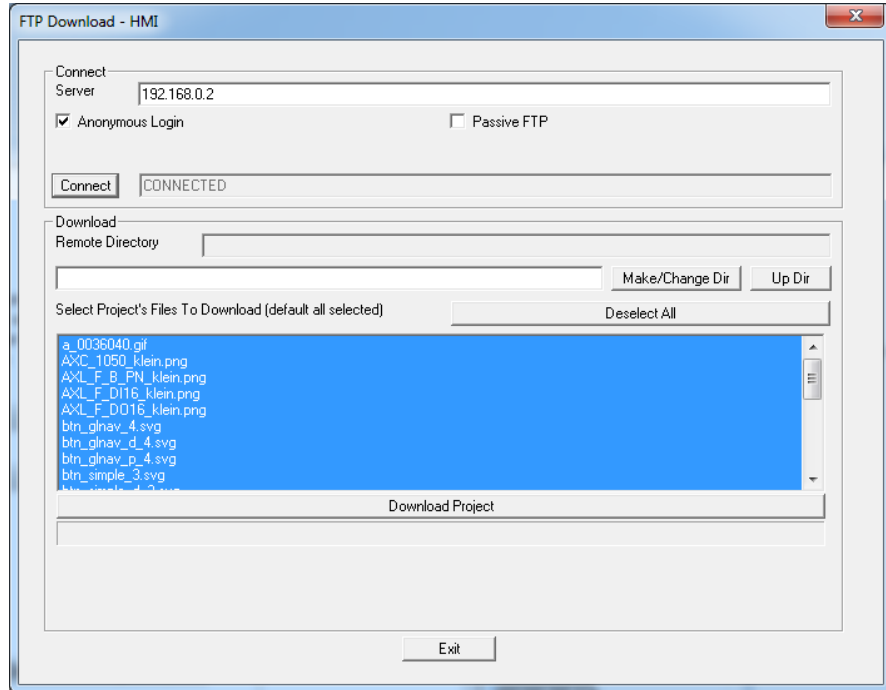


Figure 7-5 WebVisit connected to the controller

- Click on the “Download Project” button.

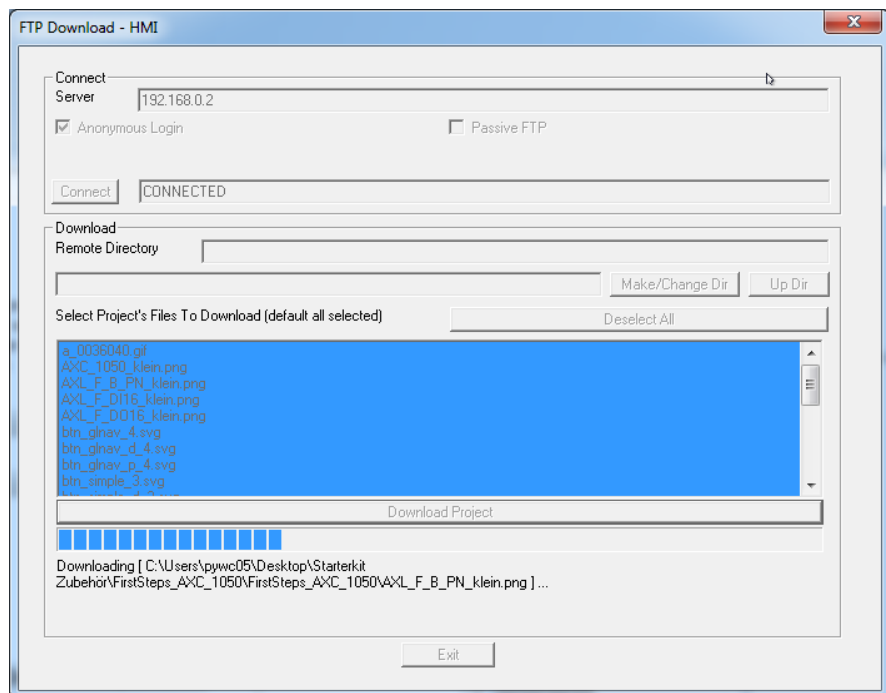


Figure 7-6 Downloading the WebVisit project to the controller

The progress bar shows the transmission status.

- After successful download, close the window by clicking on “Exit”.

The visualization project has been transmitted to the controller. The visualization can now be displayed via your web browser (see Section 7.5).

7.5 Starting the visualization



Please note:

The visualization is only started once the following setting has been made in the Windows® Control Panel.

- In the Windows® Control Panel, open the Java settings window.
- In the settings for temporary files, deactivate the “Keep temporary files on my computer” checkbox.

- In the address window of your java-compatible web browser, enter the IP address of the controller (in the example: “http://192.168.0.2”).

Once all settings have been properly made, your web browser will display the visualization's start page after a short loading time. Here you can see the running light of the PC Worx project as an animated representation of LEDs, which can be activated/deactivated using the “Start/Stop” button.

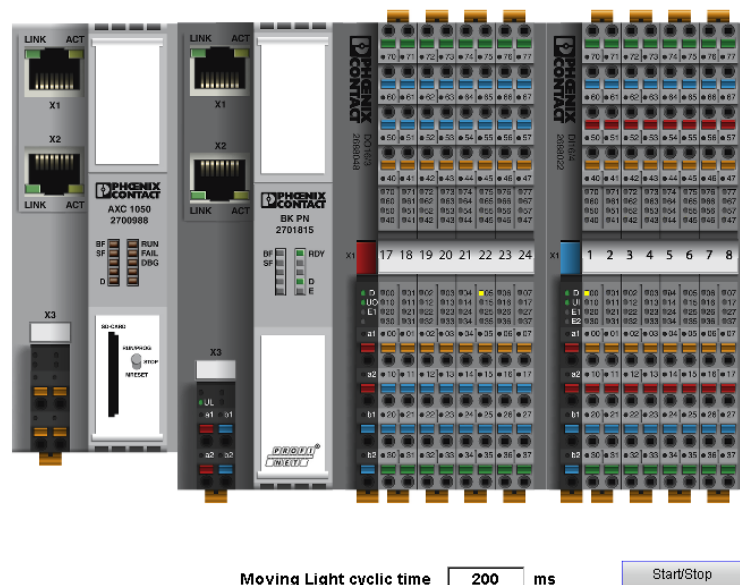


Figure 7-7 Visualization of the example program

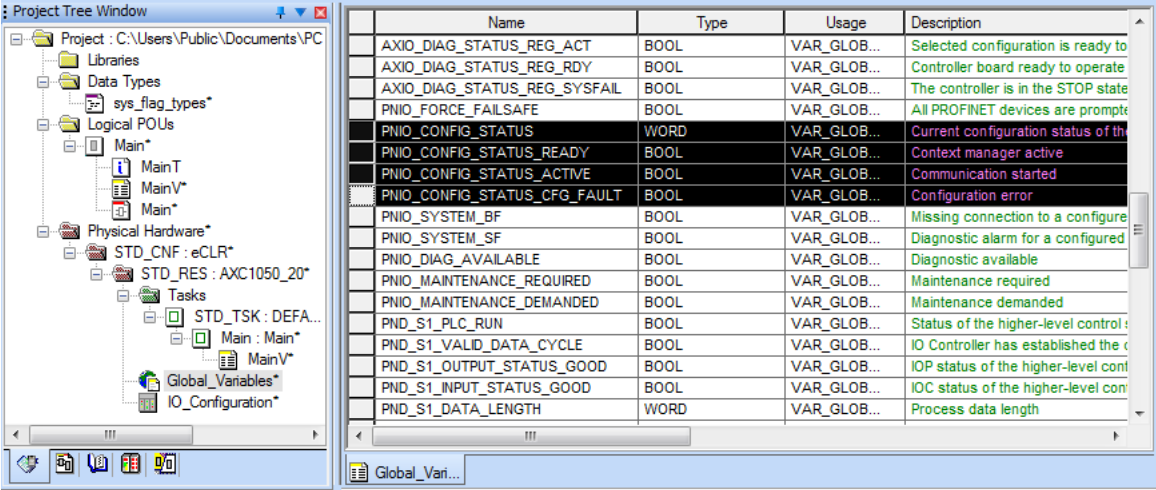
- In the input field, enter a cycle time > 100 ms.
If no cycle time is entered, the default cycle time of 200 ms (initial value of the **Time_Parameter** variable, see Section 5.12) will be used.
LEDs 00 to 07 of the Axioline F digital output module light up one after the other for the cycle time specified.
- Activate the “Start/Stop” button to activate or deactivate animation of the running light in the web browser.

A Status information and system variables

A 1 Status of the PROFINET controller

The PROFINET controller has status information that indicates its configuration status.

This can be requested via the PNIO_CONFIG_STATUS system variable. The system variable is available as a word and as individual bits under "Global_Variables".



| Name | Type | Usage | Description |
|------------------------------|------|-------------|-------------------------------------|
| AXIO_DIAG_STATUS_REG_ACT | BOOL | VAR_GLOB... | Selected configuration is ready to |
| AXIO_DIAG_STATUS_REG_RDY | BOOL | VAR_GLOB... | Controller board ready to operate |
| AXIO_DIAG_STATUS_REG_SYSFAIL | BOOL | VAR_GLOB... | The controller is in the STOP state |
| PNIO_FORCE_FAILSAFE | BOOL | VAR_GLOB... | All PROFINET devices are prompt |
| PNIO_CONFIG_STATUS | WORD | VAR_GLOB... | Current configuration status of th |
| PNIO_CONFIG_STATUS_READY | BOOL | VAR_GLOB... | Context manager active |
| PNIO_CONFIG_STATUS_ACTIVE | BOOL | VAR_GLOB... | Communication started |
| PNIO_CONFIG_STATUS_CFG_FAULT | BOOL | VAR_GLOB... | Configuration error |
| PNIO_SYSTEM_BF | BOOL | VAR_GLOB... | Missing connection to a configure |
| PNIO_SYSTEM_SF | BOOL | VAR_GLOB... | Diagnostic alarm for a configured |
| PNIO_DIAG_AVAILABLE | BOOL | VAR_GLOB... | Diagnostic available |
| PNIO_MAINTENANCE_REQUIRED | BOOL | VAR_GLOB... | Maintenance required |
| PNIO_MAINTENANCE_DEMANDED | BOOL | VAR_GLOB... | Maintenance demanded |
| PND_S1_PLC_RUN | BOOL | VAR_GLOB... | Status of the higher-level control |
| PND_S1_VALID_DATA_CYCLE | BOOL | VAR_GLOB... | IO Controller has established the c |
| PND_S1_OUTPUT_STATUS_GOOD | BOOL | VAR_GLOB... | IOP status of the higher-level cont |
| PND_S1_INPUT_STATUS_GOOD | BOOL | VAR_GLOB... | IOC status of the higher-level con |
| PND_S1_DATA_LENGTH | WORD | VAR_GLOB... | Process data length |

Figure A-1 PNIO_CONFIG_STATUS system variable

PNIO_CONFIG_STATUS_READY

The variable is set if the PROFINET controller has been initialized correctly. No desired configuration has been loaded by PC Worx yet. Connection establishment to the devices has been completed.

PNIO_CONFIG_STATUS_ACTIVE

The variable is set if the desired configuration for the PROFINET controller has been loaded. In this state, the PROFINET controller attempts to establish a connection cyclically to all devices in the desired configuration (in the bus configuration below the PROFINET icon). In this way, it attempts to find each device with the corresponding PROFINET device name ("DNS Name") in the current subnetwork.

If the connection to a PROFINET device has been established successfully, the corresponding PNIO_DATA_VALID process data item is set.

For inaccessible devices, the PROFINET controller cyclically attempts to establish a connection approximately every 5 seconds.

A connection cannot be established, e.g., if the corresponding device is ready, but a correct PROFINET device name ("DNS Name") has not yet been assigned to it (see also Section 5.7).

PNIO_CONFIG_STATUS_FAULT

This system variable is set if an error occurred when configuring the PROFINET controller.

A 2 Status of a PROFINET device

The status of a PROFINET device can be queried using predefined process data.

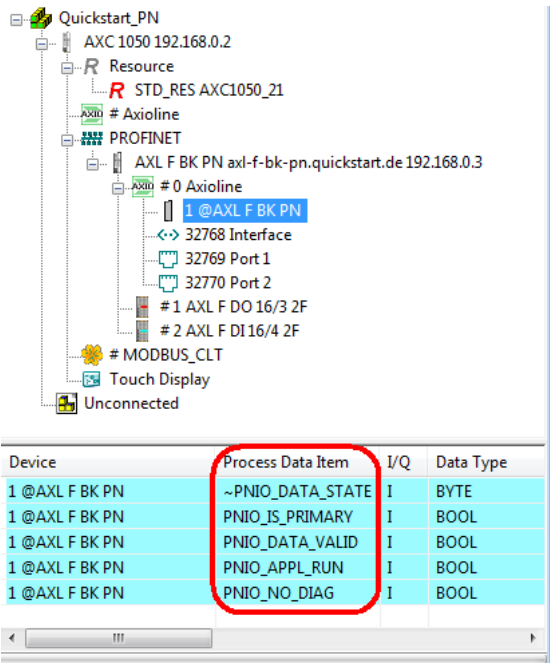


Figure A-2 Predefined process data of a PROFINET device

The ~PNIO_DATA_STATE byte that contains the corresponding status bits is available. A PROFINET device only provides valid data if these bits are set.

PNIO_IS_PRIMARY

This process data item is reserved for future redundancy functions.

PNIO_DATA_VALID

For PROFINET, it may be normal during operation that a PROFINET device within the network cannot be accessed (e.g., undocking devices).

Use this process data item to specify for your application whether or not the operation of all other PROFINET devices is to be continued despite a device failure.

In either case, the application program must receive information as to whether or not a PROFINET device is supplying valid data. This is why each PROFINET device has a PNIO_DATA_VALID process data item.

This bit must be set for the PROFINET device to supply valid data and all other process data to be valid.

PNIO_APPL_RUN

This bit indicates that the application is running on the PROFINET device.

PNIO_NO_DIAG

If this bit is set, there are no device diagnostics present.



Not all bits from byte ~PNIO_DATA_STATE are used. To prevent unused bits being used by mistake, hide them in the control program by linking to a corresponding bit mask.

To query the status, assign the process data to the corresponding variables.



If variables have been created previously and are to be assigned to the predefined process data, proceed as described in Section 5.13.

A 3 PNIO_FORCE_FAILSAFE system variable

In many cases, a machine or system can no longer be operated in the event of a PROFINET device failure. This is normal in INTERBUS systems where neither bus couplers nor isolated disconnection are used.

In the example system, a PROFINET device failure does not affect the operation of all the other PROFINET devices. If you would like to set all the outputs to a safe state in the event of an error, the PNIO_FORCE_FAILSAFE system variable can be used for this. It can be found under “Global_Variables” in the “IEC Programming” workspace.

The screenshot shows the IEC Programming workspace. On the left is the Project Tree Window with a tree structure including Libraries, Data Types, Logical POUs, Main*, Physical Hardware*, and STD_CONF: eCLR*. On the right is a table titled 'Global Variables'.

| Name | Type | Usage | Description |
|------------------------------|-------------|--------------------|--|
| AXIO_DIAG_STATUS_REG_ACT | BOOL | VAR_GLOB... | Selected configuration is ready to |
| AXIO_DIAG_STATUS_REG_RDY | BOOL | VAR_GLOB... | Controller board ready to operate |
| AXIO_DIAG_STATUS_REG_SYSFAIL | BOOL | VAR_GLOB... | The controller is in the STOP state |
| PNIO_FORCE_FAILSAFE | BOOL | VAR_GLOB... | All PROFINET devices are prompt |
| PNIO_CONFIG_STATUS | WORD | VAR_GLOB... | Current configuration status of the |
| PNIO_CONFIG_STATUS_READY | BOOL | VAR_GLOB... | Context manager active |
| PNIO_CONFIG_STATUS_ACTIVE | BOOL | VAR_GLOB... | Communication started |
| PNIO_CONFIG_STATUS_CFG_FAULT | BOOL | VAR_GLOB... | Configuration error |
| PNIO_SYSTEM_BF | BOOL | VAR_GLOB... | Missing connection to a configure |
| PNIO_SYSTEM_SF | BOOL | VAR_GLOB... | Diagnostic alarm for a configured |
| PNIO_DIAG_AVAILABLE | BOOL | VAR_GLOB... | Diagnostic available |
| PNIO_MAINTENANCE_REQUIRED | BOOL | VAR_GLOB... | Maintenance required |

Figure A-3 PNIO_FORCE_FAILSAFE system variable

If the variable is set, all the PROFINET devices output zero values or defined substitute values. The corresponding inputs are set to zero.

B Appendixes

B 1 List of figures

Section2

| | | |
|-------------|--|----|
| Figure 2-1: | Possible installation method for the AXC 1050 PN STARTERKIT | 13 |
| Figure 2-2: | Possible installation of the AXC 1050 PN STARTERKIT – Illustration with supply voltage wiring | 14 |
| Figure 2-3: | Possible installation of the AXC 1050 PN STARTERKIT – Illustration with Ethernet cabling | 15 |

Section3

| | | |
|-------------|---|----|
| Figure 3-1: | Selecting the USB driver | 18 |
| Figure 3-2: | Dialog box for entering the registration code | 19 |

Section4

| | | |
|--------------|--|----|
| Figure 4-1: | User interface | 21 |
| Figure 4-2: | “IEC Programming” workspace | 23 |
| Figure 4-3: | “Bus Configuration” workspace | 24 |
| Figure 4-4: | “Process Data Assignment” workspace | 24 |
| Figure 4-5: | “Project Comparison Result” workspace | 25 |
| Figure 4-6: | “FDT” workspace | 25 |
| Figure 4-7: | “Edit Device Representation...” menu item | 28 |
| Figure 4-8: | Device representation | 28 |
| Figure 4-9: | Examples of different device representations | 29 |
| Figure 4-10: | Example of different device representations of the individual device groups | 29 |
| Figure 4-11: | “Project Control Dialog” dialog box | 30 |
| Figure 4-12: | “Download Options” dialog box | 31 |
| Figure 4-13: | “Project Control Dialog” dialog box | 33 |

Section5

| | | |
|-------------|--|----|
| Figure 5-1: | Sequence for creating a project | 35 |
| Figure 5-2: | “New Project” dialog box: project templates | 36 |
| Figure 5-3: | “Save/Zip project as” window | 37 |
| Figure 5-4: | Project information after creating a new project | 38 |

| | | |
|--------------|---|----|
| Figure 5-5: | Selecting the network card | 39 |
| Figure 5-6: | IP settings | 40 |
| Figure 5-7: | DCP: setting the IP address | 42 |
| Figure 5-8: | Starting IP address assignment via DCP | 42 |
| Figure 5-9: | Network settings: manual assignment of the IP address | 44 |
| Figure 5-10: | “Settings Communication Path” dialog box | 44 |
| Figure 5-11: | Successful download of the IP settings | 45 |
| Figure 5-12: | Activating the network settings: restarting the controller | 45 |
| Figure 5-13: | Activating the network settings: service executed successfully | 46 |
| Figure 5-14: | Successful communication path test | 46 |
| Figure 5-15: | Activating the PROFINET controller function | 47 |
| Figure 5-16: | Reading in PROFINET | 48 |
| Figure 5-17: | Connected PROFINET devices | 48 |
| Figure 5-18: | Selecting the PROFINET device description | 49 |
| Figure 5-19: | Naming the PROFINET device | 50 |
| Figure 5-20: | Bus configuration with read in PROFINET devices | 50 |
| Figure 5-21: | All devices with PROFINET device names | 51 |
| Figure 5-22: | PROFINET settings of the “AXL F BK PN” bus coupler | 52 |
| Figure 5-23: | All PROFINET devices with PROFINET device name and IP address | 54 |
| Figure 5-24: | “Read Axioline” context menu | 55 |
| Figure 5-25: | Reading in connected Axioline F modules: Axioline F modules that have not yet been read in | 56 |
| Figure 5-26: | Read in Axioline F modules | 56 |
| Figure 5-27: | “Moving_Light” program code | 57 |
| Figure 5-28: | “Insert... User Library...” context menu | 58 |
| Figure 5-29: | Context menu of the function block library: “Open library as project” | 59 |
| Figure 5-30: | Inserted “Moving_Light” function block library | 59 |
| Figure 5-31: | “IEC Programming” workspace, empty position on the “Main” worksheet selected | 60 |
| Figure 5-32: | “Moving_Light” function block: “Variable Properties” window | 60 |
| Figure 5-33: | “Moving_Light” function block inserted in the “Main” worksheet | 61 |
| Figure 5-34: | Properties of the Time_Parameter variable | 62 |
| Figure 5-35: | Time_Parameter variable at the “CyclicTime” input | 62 |
| Figure 5-36: | All the required variables have been created and connected to the corresponding inputs and outputs of the function block | 63 |
| Figure 5-37: | Process data item “OUT00” assigned to variable LED_0 | 64 |
| Figure 5-38: | All process data used is assigned to variables | 65 |
| Figure 5-39: | Debug mode: “MainV” | 66 |

| | | |
|--------------|--------------------------------------|----|
| Figure 5-40: | Debug mode: “Main” | 67 |
| Figure 5-41: | Debug mode: “Global Variables” | 67 |

Section6

| | | |
|--------------|---|----|
| Figure 6-1: | List of all PROFINET devices that can be accessed in the network | 72 |
| Figure 6-2: | Selecting the PROFINET device | 72 |
| Figure 6-3: | Renumbering devices | 74 |
| Figure 6-4: | Diag+ start screen | 75 |
| Figure 6-5: | Diag+: selecting the communication path | 76 |
| Figure 6-6: | Diag+: example for device representation | 76 |
| Figure 6-7: | Diag+: selecting the information to be represented | 76 |
| Figure 6-8: | Axioline diagnostics: no error | 77 |
| Figure 6-9: | AXL F BK PN bus coupler: peripheral error | 78 |
| Figure 6-10: | AXL F DI16/4 2F Axioline F module: supply voltage not present | 79 |
| Figure 6-11: | Context menu of the PROFINET device | 80 |
| Figure 6-12: | PROFINET diagnostics | 81 |
| Figure 6-13: | Inserting the “Generic IP Device” | 82 |
| Figure 6-14: | Adapting the IP address and PROFINET device name (“DNS Name”) | 83 |
| Figure 6-15: | Device catalog: import device | 84 |
| Figure 6-16: | Directory for device description files | 84 |
| Figure 6-17: | Device catalog: import GSD file | 85 |

Section7

| | | |
|-------------|---|----|
| Figure 7-1: | Activating the “PDD” checkbox in the “Variable Properties” window ... | 87 |
| Figure 7-2: | Activating the “PDD” checkbox on the worksheet (Global_Variables) .. | 88 |
| Figure 7-3: | PC Worx example project: setting the path | 89 |
| Figure 7-4: | WebVisit: FTP download to the controller | 90 |
| Figure 7-5: | WebVisit connected to the controller | 91 |
| Figure 7-6: | Downloading the WebVisit project to the controller | 91 |
| Figure 7-7: | Visualization of the example program | 92 |

Appendix A

| | | |
|-------------|--|----|
| Figure A-1: | PNIO_CONFIG_STATUS system variable | 95 |
| Figure A-2: | Predefined process data of a PROFINET device | 96 |
| Figure A-3: | PNIO_FORCE_FAILSAFE system variable | 97 |

