User Manual

UM QS EN PC WORX EXPRESS

Order No.: —

PC WorX Express
AUTOMATION

User Manual
PC WorX Express

Designation: UM QS EN PC WORX EXPRESS
Revision: 02
Order No.: —

This user manual is valid for:

<table>
<thead>
<tr>
<th>Designation</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC WorX Express</td>
<td>5.20 SP1 or later</td>
</tr>
<tr>
<td>As part of the AUTOMATIONWORX Software Suite 2008</td>
<td>1.40 SP1 or later</td>
</tr>
</tbody>
</table>
Please observe the following notes

In order to ensure the safe use of the product described, you have to read and understand this manual. The following notes provide information on how to use this manual.

**User group of this manual**

The use of products described in this manual is oriented exclusively to qualified application programmers and software engineers, who are familiar with the safety concepts of automation technology and applicable standards.

Phoenix Contact accepts no liability for erroneous handling or damage to products from Phoenix Contact or third-party products resulting from disregard of information contained in this manual.

**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 messages that follow this symbol to avoid possible injury or death.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="emoji" alt="DANGER" /></td>
<td>This indicates a hazardous situation which, if not avoided, will result in death or serious injury.</td>
</tr>
<tr>
<td><img src="emoji" alt="WARNING" /></td>
<td>This indicates a hazardous situation which, if not avoided, could result in death or serious injury.</td>
</tr>
<tr>
<td><img src="emoji" alt="CAUTION" /></td>
<td>This indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.</td>
</tr>
<tr>
<td><img src="emoji" alt="NOTE" /></td>
<td>This symbol and the accompanying text alerts the reader to a situation which may cause damage or malfunction to the device, either hardware or software, or surrounding property.</td>
</tr>
<tr>
<td><img src="emoji" alt="INFO" /></td>
<td>This symbol and the accompanying text provides additional information to the reader. It is also used as a reference to other sources of information (manuals, data sheets, literature) on the subject matter, product, etc.</td>
</tr>
</tbody>
</table>

The following types of messages provide information about possible property damage and general information concerning proper operation and ease-of-use.
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 data sheets, installation instructions, manuals, etc.) does not constitute any further duty on the part of Phoenix Contact to furnish information on alterations to products and/or technical documentation. Any other agreement shall only apply if expressly confirmed in writing by Phoenix Contact. Please note that the supplied documentation is product-specific documentation only and that you are responsible for checking the suitability and intended use of the products in your specific application, in particular with regard to observing the applicable standards and regulations.

Although Phoenix Contact makes every effort to ensure that the information content is accurate, up-to-date, and state-of-the-art, technical inaccuracies and/or printing errors in the information cannot be ruled out. Phoenix Contact does not offer any guarantees as to the reliability, accuracy or completeness of the information. All information made available in the technical data is supplied without any accompanying guarantee, whether expressly mentioned, implied or tacitly assumed. This information does not include any guarantees regarding quality, does not describe any fair marketable quality, and does not make any claims as to quality guarantees or guarantees regarding the suitability for a special purpose.

Phoenix Contact accepts no liability or responsibility for errors or omissions in the content of the technical documentation (in particular data sheets, installation instructions, manuals, etc.).

The aforementioned limitations of liability and exemptions from liability do not apply, insofar as liability must be assumed, e.g., according to product liability law, in cases of premeditation, gross negligence, on account of loss of life, physical injury or damage to health or on account of the violation of important contractual obligations. Claims for damages for the violation of important contractual obligations are, however, limited to contract-typical, predictable damages, provided there is no premeditation or gross negligence, or that liability is assumed on account of loss of life, physical injury or damage to health. This ruling does not imply a change in the burden of proof to the detriment of the user.
Statement of legal authority

This manual, including all illustrations contained herein, is copyright protected. Use of this manual by any third party is forbidden. Reproduction, translation, and public disclosure, as well as electronic and photographic archiving or alteration requires the express written consent of Phoenix Contact. Violators are liable for damages.

Phoenix Contact reserves all rights in the case of patent award or listing of a registered design, in as far as this concerns software of Phoenix Contact that meets the criteria of technicity or has technical relevance. Third-party products are always named without reference to patent rights. The existence of such rights shall not be excluded.


All other product names used are trademarks of the respective organizations.

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:

www.phoenixcontact.com

Make sure you always use the latest documentation.
It can be downloaded at:

www.download.phoenixcontact.com

A conversion table is available on the Internet at:

www.download.phoenixcontact.com/general/7000_en_00.pdf

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 www.phoenixcontact.com.

Published by

PHOENIX CONTACT GmbH & Co. KG
Flachsmarktstraße 8
32825 Blomberg
Germany
Phone +49 - (0) 52 35 - 3-00
Fax +49 - (0) 52 35 - 3-4 12 00

PHOENIX CONTACT
P.O. Box 4100
Harrisburg, PA 17111-0100
USA
Phone +1-717-944-1300
Fax +1-717-944-1300

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.
# Table of Contents

1 General........................................................................................................................................... 1-1
  1.1 Introduction......................................................................................................................... 1-1
  1.2 Information about this manual............................................................................................. 1-1
  1.3 System requirements............................................................................................................. 1-2
    1.3.1 Supported operating systems ............................................................................ 1-2
    1.3.2 Hardware requirements....................................................................................... 1-2
    1.3.3 Software requirements......................................................................................... 1-2
  1.4 Ordering data ......................................................................................................................... 1-2

2 Installing and enabling the software......................................................................................... 2-1
  2.1 Prior to installation................................................................................................................ 2-1
  2.2 AUTOMATIONWORX Software Suite ............................................................................. 2-1
  2.3 Starting the installation program......................................................................................... 2-2
  2.4 Starting PC WorX Express ................................................................................................. 2-2

3 Helpful information about PC WorX Express ............................................................................ 3-1
  3.1 Online help ........................................................................................................................ 3-1
  3.2 Changing the language ...................................................................................................... 3-1
  3.3 PC WorX Express user interface ....................................................................................... 3-2
  3.4 Toolbars ............................................................................................................................ 3-3
  3.5 Workspaces ....................................................................................................................... 3-4
    3.5.1 Windows in the workspaces .............................................................................. 3-6
    3.5.2 Toggling windows on/off and docking/undocking windows .............................. 3-7
  3.6 "Bus Structure" window ..................................................................................................... 3-8
    3.6.1 Icons in the "Bus Structure" window .................................................................. 3-8
    3.6.2 Display in the "Bus Structure" window ............................................................ 3-10
  3.7 "EXCEL Link" window ....................................................................................................... 3-11
  3.8 "Diag+" window .................................................................................................................. 3-12
  3.9 Visualization ..................................................................................................................... 3-16

4 Sequence for creating a project................................................................................................. 4-1
  4.1 Sequence for creating a project.......................................................................................... 4-2
  4.2 Creating a new project........................................................................................................ 4-3
  4.3 Specifying project information.......................................................................................... 4-4
  4.4 Checking/modifying IP settings for the controller............................................................ 4-6
  4.5 Decision: Working online or offline .................................................................................. 4-6
  4.6 Manually inserting INTERBUS devices.......................................................................... 4-7
  4.7 Compiling after completing the bus topology................................................................. 4-10
  4.8 Creating the program......................................................................................................... 4-11
  4.9 Compiling after creating the program.............................................................................. 4-12
4.10 Generating variables and assigning process data ............................................4-12
  4.10.1 Generating variables .................................................................4-12
  4.10.2 Assigning process data .............................................................4-14
4.11 Switching to working with the system (online)...................................................4-16
4.12 Assigning the IP address for the controller/BootP server ..................................4-16
  4.12.1 Address assignment with PC WorX Express via the serial interface .........4-19
4.13 Setting the communication path .......................................................................4-22
  4.13.1 Communication via Ethernet ......................................................4-23
4.14 Reading INTERBUS .........................................................................................4-24
4.15 Compiling after reading the bus topology .........................................................4-26
4.16 Creating the program ...................................................................................4-26
4.17 Compiling after creating the program ..............................................................4-26
4.18 Assigning process data ....................................................................................4-27
4.19 Compiling and sending a project, and performing a cold restart .......................4-27
  4.19.1 Compiling a project .....................................................................4-27
  4.19.2 Sending a project ........................................................................4-27
  4.19.3 Performing a cold restart ...........................................................4-29
4.20 Operation ......................................................................................................4-30
  4.20.1 Setting the task properties ..............................................................4-30
  4.20.2 Debug mode ................................................................................4-32
  4.20.3 PLC stop/run ..............................................................................4-34
5  Example project for an INTERBUS system .................................................................5-1
  5.1 Describing the project ..................................................................................5-1
  5.2 Sequence for creating the INTERBUS project .............................................5-2
  5.3 Creating a new project ..................................................................................5-4
  5.4 Specifying project information .....................................................................5-4
  5.5 Checking/modifying IP settings for the controller ..........................................5-4
  5.6 Assigning the IP address for the controller/BootP server ...............................5-5
  5.7 Setting the communication path .................................................................5-6
  5.8 Reading INTERBUS ..................................................................................5-6
  5.9 Compiling after completing the bus topology ...............................................5-7
  5.10 Creating the program ................................................................................5-7
  5.11 Compiling after creating the program ........................................................5-7
  5.12 Assigning process data .............................................................................5-8
  5.13 Compiling and sending a project, and performing a cold restart .................5-9
    5.13.1 Compiling a project .....................................................................5-9
    5.13.2 Sending a project .......................................................................5-9
  5.14 Operation ..................................................................................................5-9
# Table of Contents

6  Example program ........................................................................................................... 6-1
   6.1  Program description ............................................................................................... 6-1
   6.2  Function blocks used in the example ..................................................................... 6-2
   6.3  Programming ......................................................................................................... 6-3
      6.3.1  Example program ......................................................................................... 6-4
      6.3.2  Starting programming .................................................................................. 6-5
      6.3.3  Entering a comment – Title of the POU ...................................................... 6-6
      6.3.4  Creating variables ......................................................................................... 6-7
      6.3.5  Creating user variables – Assigning the default flash speed in the example ... 6-8
      6.3.6  Creating system variables – Function blocks for selecting inputs in the example 6-9
      6.3.7  Creating a user-defined function block ....................................................... 6-12
      6.3.8  Adding a user-defined function block ........................................................... 6-14
      6.3.9  Deleting variables ......................................................................................... 6-14
      6.3.10  Setting the initial value ................................................................................ 6-15
   6.4  Compiling after creating the program ................................................................. 6-15
   6.5  Assigning process data ......................................................................................... 6-15
   6.6  Compiling and sending a project (including program), and performing a cold restart 6-15

A  Additional software functions ......................................................................................... A-1
   A 1  Setting the realtime clock .................................................................................... A-1
   A 2  Replacing a control system (hardware replacement) ........................................... A-2
   A 3  Device description files ....................................................................................... A-6
      A 3.1  Phoenix Contact device description files .................................................... A-6
      A 3.2  Creating device description files ................................................................... A-7
   A 4  Visualization ......................................................................................................... A-7
   A 5  INTERBUS topology ............................................................................................ A-7

B  AX OPC Server and WebVisit ..................................................................................... B-1
   B 1  AX OPC Server .................................................................................................. B-1
      B 1.1  Preparatory tasks in PC WorX Express ....................................................... B-1
      B 1.2  OPC Configurator ...................................................................................... B-2
      B 1.3  OPC Test Client ......................................................................................... B-4
   B 2  WebVisit ............................................................................................................. B-8

C  Overview of Phoenix Contact control systems ............................................................ C-1
1 General

1.1 Introduction

PC WorX Express is a new, easy-to-use version of PC WorX software. The software provides a simplified introduction to the world of IEC 61131 programming for the 100 controller class from Phoenix Contact. The graphical user interface is tailored to the requirements of a programming beginner. It provides a simple and quick project overview. Workspace that users can dock and undock, and configurable toolbars, which can be customized to the individual needs of the user, increase the efficiency of the programming. Operation is intuitive thanks to fewer menus and the informative icons. PC WorX Express supports the IEC 61131-3 programming languages – structured text (ST) and ladder diagram (graphical programming), as well as function block diagram (FBD).

PC WorX Express has an integrated bus configurator for configuring network structures, which means that PC WorX Express supports INTERBUS and Ethernet networks. The configurator device catalog lists all the necessary components in easy-to-understand groupings. The components can be added to the hardware configuration using drag & drop. All data configured in PC WorX Express can be easily reused for visualization purposes. The AX OPC server and a web server can be used to connect to the visualization and control level. The OPC and web server variables are selected by simply clicking on them. All system components in an INTERBUS network can undergo complete diagnostics in PC WorX Express using the integrated Diag+ diagnostics tool.

The PC WorX Express programming version is available free of charge and does not use a license mechanism. The software can be downloaded at www.phoenixcontact.com.

PC WorX Express provides 64 KB of input data and 64 KB of output data.

1.2 Information about this manual

This document helps you to parameterize a bus configuration and to program the application program (according to IEC 61131-3) using an example project.

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

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

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.
1.3 System requirements

1.3.1 Supported operating systems

- Windows 2000 SP4
- Windows XP SP2 (recommended)
- Windows Vista

1.3.2 Hardware requirements

<table>
<thead>
<tr>
<th>Hardware requirements for PC WorX Express</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Pentium III 1 GHz, 2 GHz (recommended)</td>
</tr>
<tr>
<td>Main memory</td>
<td>1 GB (minimum), 2 GB (recommended) Windows Vista</td>
</tr>
<tr>
<td></td>
<td>512 MB (minimum), 1 GB (recommended) Windows XP, Windows 2000 SP4</td>
</tr>
<tr>
<td>Hard disk space</td>
<td>1 GB free memory space</td>
</tr>
<tr>
<td>CD-ROM drive</td>
<td>Yes</td>
</tr>
<tr>
<td>Interfaces</td>
<td>1 x serial, Ethernet</td>
</tr>
<tr>
<td>Monitor</td>
<td>SVGA, resolution of 800 x 600 pixels (minimum), SXGA, resolution of 1280 x 1024 pixels (recommended)</td>
</tr>
<tr>
<td>Operating devices</td>
<td>Keyboard, mouse</td>
</tr>
</tbody>
</table>

1.3.3 Software requirements

<table>
<thead>
<tr>
<th>Software requirements for PC WorX Express</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet Explorer</td>
<td>Version 5.5 or later</td>
</tr>
</tbody>
</table>

1.4 Ordering data

<table>
<thead>
<tr>
<th>Designation</th>
<th>Order No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PC WorX Express</td>
<td>2988670</td>
<td>Free IEC 61131-compliant programming software; maximum of 64 KB of input data, 64 KB of output data</td>
</tr>
</tbody>
</table>

This version supports structured text (ST) and ladder diagram (LD)/function block diagram (FBD) programming languages according to IEC 61131-3.
Installing and enabling the software

2 Installing and enabling the software

2.1 Prior to installation

Prior to installation, close all open Windows applications.

2.2 AUTOMATIONWORX Software Suite

PC WorX Express is part of the AUTOMATIONWORX Software Suite.

The AUTOMATIONWORX Software Suite includes the following programs:

- **Config+**
  Easy configuration and startup for INTERBUS networks
- **Diag+**
  User-friendly network diagnostics during startup and operation
- **Diag+ NetScan**
  User-friendly monitoring of multiple INTERBUS networks
- **PC WorX Express**
  IEC 61131 programming environment (structured text, ladder diagram/function block diagram) for the 100 controller class from Phoenix Contact
- **PC WorX**
  Uniform IEC 61131 programming environment for all Phoenix Contact control systems
- **AX OPC Server**
  Software used for data exchange between distributed INTERBUS networks and visualization systems
- **WebVisit**
  Tool for creating web pages for Phoenix Contact control systems

The desired programs can be selected individually or simultaneously for installation.

Diag+ should only be selected if you wish to use it independently of PC WorX Express.

When installing PC WorX Express, Diag+ is installed as part of PC WorX Express.

PC WorX Express can be used as soon as it is started.

If other programs in the software suite have been installed, they run in demo mode with limited resources. A registration code is required to enable the full version. You will receive the registration code when a full version of the relevant program is purchased.
2.3 Starting the installation program

- Insert the "AUTOMATIONWORX Software Suite" CD in your CD-ROM drive. The installation program usually starts automatically after a few seconds.
- If this is not the case, start the "SETUP.EXE" file from the "[Drive]\Setup" directory on the CD-ROM. This file calls the installation wizard, which guides you through the installation process.
- Follow the instructions in the installation program.

The installation program generates all the directories required for operation and copies the files for the selected programs.
- Following successful installation, restart your computer to activate the changes to the configuration files. To do this, click on "Finish" at the end of the installation process.

2.4 Starting PC WorX Express

- For installation using the default settings, start PC WorX Express via "Start, Programs, Phoenix Contact, AUTOMATIONWORX Software Suite <Version>, PC WorX Express 5.xx".

PC WorX Express can be used as soon as it is started.
3 Helpful information about PC WorX Express

3.1 Online help

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

3.2 Changing the language

When installing the software, the language in which PC WorX Express 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".

![Image of the language selection dialog box](image_url)

Figure 3-1 Changing the language setting

The selected language is activated the next time PC WorX Express is started.

- Close and restart the program.
3.3 PC WorX Express 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 workspace.
3.4 Toolbars

The program contains several toolbars with different icons, which enables frequently used operations to be executed quickly. Alternatively, these operating steps can be called via menu items or predefined shortcuts. The toolbars that are displayed vary depending on the workspace that is set.

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.

Icons for selecting the workspace

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

![Icons for selecting the workspace](image)

- Activate IEC programming workspace.
- Activate bus configuration workspace.
- Activate process data assignment workspace.

Frequently used icons for compiling and debugging

![Frequently used icons for compiling and debugging](image)

- Make (compile project; corresponds to “Build, Make” in the menu bar).
- Switch debug mode on/off.
- Display project control dialog box.
3.5 Workspaces

PC WorX Express is divided into three workspaces:
- IEC programming
- Bus configuration
- Process data assignment

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 3-3 to Figure 3-5 show the default workspaces. The windows that you wish to display (message window, etc.) can be defined at any time for each workspace. Table 3-1 provides an overview of the windows that can be usefully added to the default setting.

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

![IEC programming workspace](image)

Figure 3-3 IEC programming workspace
Bus configuration workspace

Figure 3-4  Bus configuration workspace

Process data assignment workspace

Figure 3-5  Process data assignment workspace
### 3.5.1 Windows in the workspaces

Table 3-1 lists which windows are set by default when PC WorX Express is started for the first time and which windows can be usefully added.

<table>
<thead>
<tr>
<th>Window</th>
<th>IEC programming</th>
<th>Bus configuration</th>
<th>Process data assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>D</td>
<td>A</td>
<td>D</td>
</tr>
<tr>
<td>Project tree window</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Message window</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Edit wizard</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Cross references window</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Watch window</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Logic analyzer</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Status bar</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Bus Structure</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Device Details</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Device Catalog</td>
<td>Yes</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>Process Data Assignment</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>Connected Bus</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>EXCEL Link</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
<tr>
<td>INTERBUS Topology</td>
<td></td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>

Key:

1. When inserted manually
2. When using Excel

D  Default

A  Useful addition
3.5.2 Toggling windows on/off and docking/undocking windows

Toggling on/off

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

For the windows recommended for the relevant workspace, please refer to Section 3.5, "Workspaces" on page 3-4 and onwards.

Docking/undocking

For each window, you can specify whether it is to be docked in the other windows or not. 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 gray window frame or the title bar 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.
3.6 "Bus Structure" window

The "Bus Structure" window is used to display and edit the bus topology.

3.6.1 Icons in the "Bus Structure" window

3.6.1.1 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 3-2 Icons for logical device functions (examples)

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Resource Icon" /></td>
<td>Control system resource&lt;br&gt;When creating the project using a template, the control system resource is inserted below this icon.&lt;br&gt;When creating the project using the wizard, the control system resource is inserted below this icon.</td>
</tr>
<tr>
<td><img src="image" alt="INTERBUS Icon" /></td>
<td>INTERBUS master&lt;br&gt;Below this icon, insert the INTERBUS devices that are directly connected to the device that this icon refers to (local bus and remote bus).</td>
</tr>
</tbody>
</table>

For the example in Figure 3-6:

- Standard resource: ILC 150 ETH controller
- Device number 0.1: IB IL AO 1/U/SF analog output module, with bus segment number 0 and position number 1
- Device number 0.2: IB IL AI 2/SF-ME analog input module, with bus segment number 0 and position number 2

![INTERBUS controller](image) Overall structure with INTERBUS controller and INTERBUS devices

Figure 3-6 Example for the representation of devices in the bus configuration
### 3.6.1.2 Validity of actions

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

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>Insert at the same level</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Insert in the lower level</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Replace</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Not permitted</td>
</tr>
</tbody>
</table>

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

### 3.6.1.3 Display of status information

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

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>The device is hidden or the bus is deactivated.</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Errors have occurred for the device.</td>
</tr>
<tr>
<td>![Icon]</td>
<td>Warnings have occurred for the device.</td>
</tr>
</tbody>
</table>

- Errors have occurred for the device.
- Warnings have occurred for the device.
3.6.2  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, open a device via the "Edit Device Representation..." menu item.

![Figure 3-7 "Edit Device Representation..." menu item]

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

![Figure 3-8 Device representations]
Examples for different device representations:

The selected representation only affects the representation of a specific device group. Groups include, for example:
- Control systems
- INTERBUS devices

For example, the setting for the Inline Controller does not have any influence on the representation of the Inline modules (see Figure 3-9).

### 3.7 "EXCEL Link" window

The Excel link enables:
- Global variable data to be exported to an Excel file
- Global variable data to be imported from an Excel file

Data that can be exported/imported includes:
- IEC variables
- IEC variable connections to signals
- Associated data and text

The data for all IEC variables can thus be edited in an Excel worksheet rather than in various dialog boxes in the PC WorX Express user interface.

Since IEC variables are edited in this window, it is useful to toggle the window on in the process data assignment workspace.

The following functions can be executed with the Excel link:
- Export existing IEC variables.
- Adjust and import the IEC variables available in an Excel file with a specified format with the IEC variables available in a PC WorX Express project.
- Settings that specify how you wish to identify devices.

For more detailed information about the Excel link, please refer to the online help for PC WorX Express.

To toggle the "EXCEL Link" window on, select the "View, EXCEL Link" menu item.
3.8 "Diag+" window

Diag+ is a diagnostics tool for the seamless diagnostics of INTERBUS.

When installing PC WorX Express, Diag+ is installed automatically and is integrated in PC WorX Express. Diag+ can be called via the INTERBUS master context menu (see Figure 3-10).

Starting Diag+

- Start Diag+ via the INTERBUS master context menu.

Figures 3-10 Calling Diag+: INTERBUS master context menu

When started, Diag+ establishes a connection to the INTERBUS controller (in the example bus configuration: ILC 150 ETH with IP address "192.168.0.2", see Figure 3-10).

For additional information about setting the IP address, please refer to "Checking/modifying IP settings for the controller" on page 4-6.
"Bus Info" tab

In the following, when the displayed dialog box is started, the "Bus Info" tab is displayed (see Figure 3-11).

```
<table>
<thead>
<tr>
<th>Statistics</th>
<th>Device Diagnostic</th>
<th>History</th>
<th>Bus Architecture</th>
<th>Bus Info</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Devices</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cycles Time (ms)</td>
<td>1.30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Approximate Runtime (ms)</td>
<td>0.003</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of Cycles</td>
<td>20767</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of Disturbed Cycles</td>
<td>k</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hard Ratio</td>
<td>500 k/100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Insensitive Devices</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality Bit</td>
<td>Not set</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Warning Bit</td>
<td>Not set</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

Figure 3-11 "Bus Info" tab (start screen)

"Settings" tab

- Select the "Settings" tab (see Figure 3-11).

```
<table>
<thead>
<tr>
<th>Communication Path to the Controller Board</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbol Name</td>
</tr>
<tr>
<td>Path Info</td>
</tr>
<tr>
<td>File Selection</td>
</tr>
<tr>
<td>View</td>
</tr>
<tr>
<td>Exclusive Rights</td>
</tr>
<tr>
<td>Version</td>
</tr>
</tbody>
</table>
```

Figure 3-12 "Settings" tab
Under "View", select:
- Which device information is to be displayed
- How often the diagnostic data is to be updated

View in Figure 3-12:
- Device information: "Device Number" specified
- Update diagnostic data: "Manual", e.g., when changing tabs

**"Bus Architecture" tab**

- Select the "Bus Architecture" tab (see Figure 3-13).

An error is simulated, the IB IL AI 2/SF-ME Inline terminal is disconnected.

![Figure 3-13 "Bus Architecture" tab (no error)](image)

![Figure 3-14 "Bus Architecture" tab (error)](image)
"Solution" tab

- Select the "Solution" tab.

Here, information is provided about how to remove the error.

![Figure 3-15 "Solution" tab](image)

- Remove the error (here: insert Inline terminal again).

"Action" tab

- If the bus is not restarted automatically, the "Action" tab can be used, for example, to acknowledge errors, reset the controller, and start the bus.

![Figure 3-16 "Action" tab](image)

- Click on "Start Bus" to start up the entire bus configuration again following error removal.
Depending on the bus configuration that you have set, peripheral faults may have to be reset before the bus can be restarted following error removal.

- In this case, click on “Acknowledge all Peripheral Faults” to reset the error message for the corresponding module.
- Click on “Start Bus” again to start up the entire bus configuration again following error removal.

Successful error removal with error-free bus operation is indicated in the status bar:

![Status Bar](image)

Figure 3-17  "Action" tab: Bus running without errors (RUN)

### 3.9 Visualization

The data generated in PC WorX Express can be used in other programs, e.g., to visualize processes (e.g., Visu+, WebVisit).

The AX OPC Server is provided in order to enable the use of data in OPC clients.

See also "AX OPC Server and WebVisit" on page B-1.
4 Sequence for creating a project

This section describes the general procedure for creating a project.

**The aim of this section is** to provide an overview of the various options available for achieving a particular objective and to explain some points in detail that are only referred to briefly in later sections.

**The aim is not** to create a comprehensive project or to have an operational project at the end of the section. If you create a project using the description in this section, please note that you must enter the relevant versions for your project.

If you have little or no experience in creating projects, please proceed as described in Section 5, "Example project for an INTERBUS system". This section provides a brief description of project creation for an INTERBUS system.
4.1 Sequence for creating a project

Figure 4-1  Sequence for creating a project
4.2 Creating a new project

- Select the "New Project..." command from the "File" menu to create a new project using a template.

  The tree structure and the selection of the control system are now prepared.

- Select the control system.

  For control systems, there may be several templates depending on the hardware and firmware version of the control system. Select the template that corresponds to the hardware and firmware version of the control system used.

- Confirm your selection with "OK".

---

"Large Icons" view has been selected in Figure 4-2. The view can be switched via the icons.

The ">" character in the selection stands for "or later" and not "greater than" in a mathematical sense. The specified version is the first supported version.

If a control system is inserted in a project without a template, depending on the hardware and firmware version, the processor type of the control system used must be specified. A detailed table that lists the processor type and scope of functions according to the control system used can be found in "Overview of Phoenix Contact control systems" on page C-1.
Select the "File, Save Project As/Zip Project As..." command.
Enter a project name and save the project.

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.

The project information highlighted in Figure 4-4 on page 4-4 cannot be modified. Notes on adjusting the IP settings are provided in the following.
Sequence for creating a project

First and Last IP Address, Subnetmask, Default Gateway

During project creation, PC WorX Express 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 another address area (e.g., a global network), adapt the start and end address on the project node (here: Quickstart_PCWExpress) to your application.

If the IP parameters are modified in the project settings, the IP addresses of the controller may have to be modified manually (see “Assigning the IP address for the controller/BootP server” on page 4-16).

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

For information about 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, please modify the end address first.

If you manually assign IP parameters that are outside the defined area, they will not be accepted and/or a warning displayed.

If you are using a default gateway:

- Assign the address to the default gateway under "Default Gateway" in the "Device Details" window.

Please note:
- This address must be within the specified IP address area.
- Address “0.0.0.0” cannot be entered.
4.4 Checking/modifying IP settings for the controller

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

NOTE: If any modifications are made to the project information 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 as the IP settings.

Adapt these settings, if necessary.

- 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, if necessary.

If an IP address has still not been assigned, assign one according to "Assigning the IP address for the controller/BootP server" on page 4-16.

Figure 4-5 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.

4.5 Decision: Working online or offline

If your system is installed, you can work online. In this case, skip this section and proceed to Section 4.13, "Setting the communication path".

If your system is not installed or you wish to work offline, proceed as described in Section 4.6, "Manually inserting INTERBUS devices".
4.6 Manually inserting INTERBUS devices

If the bus configuration is not actually available yet or you wish to create the configuration offline, the bus configuration can be created manually.

- Click to select a point in the "Bus Structure" window where a device is to be inserted (e.g., INTERBUS node for the control system).
- In the device catalog, open the product range for the inserted device (e.g., IL for Inline) under "Phoenix Contact".
- Under the product range, open the product group (e.g., I/O analog).
- Select the device to be inserted (e.g., IB IL AO 1/U/SF).

![Device Catalog](image)

Figure 4-6 Selecting the device

- Hold down the mouse button and drag the selected device to the insertion point. Please refer to the information about the mouse pointer in "Icons in the "Bus Structure" window" on page 3-8.
  Alternatively, double-click on the device to be inserted. This device is appended to the highlighted device as the next device in the bus configuration.
• Insert all the other devices.

Examples for bus configurations

Example 1: Simple bus configuration

Folders in the device catalog where the devices listed in Figure 4-8 can be found:

- ILC 150 ETH
- ILC 1xx - PLC
- IB IL A....
- IL - I/O analog
Sequence for creating a project

Example 2: Bus configuration with branch terminals

Folders in the device catalog where the devices listed in Figure 4-9 can be found:

- ILC 150 ETH
- IBS IL 24 RB-T, IBS IL 24 BK-T/U
- IB IL A....
- IB IL 24 D....

Example 3: Bus configuration with bus coupler for connecting a remote bus branch

Folders in the device catalog where the devices listed in Figure 4-10 can be found:

- ILC 150 ETH
- IBS IL 24 RB-LK
- IBS IL 24 BK-RB-LK, IBS IL 24 BK-T/U
- IB IL A....
- IB IL 24 D....
- FLS IB M12 DIO 8/8
4.7 Compiling after completing the bus topology

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

Compiling a project

- When compiling a project for the first time, select the "Rebuild Project" command from the "Build" menu. For subsequent compiling processes, the "Make" command in the "Build" menu can also be used (see also "Compiling (additional information)" on page 4-10).

- 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.

The results of the compiling process are displayed in the message window together with details of the number of errors and warnings.

Compiling (additional information)

There are two options for compiling:
1. "Build, Make"
2. "Build, Rebuild Project"

Rebuild 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 sent to the PLC.

The "Rebuild Project" command should only be used if errors occurred when compiling with "Make" or your project was unpacked without frontend.
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 using the shortcut <F9>.

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

After the compiler has been started, the message window appears automatically if it was closed before. This window indicates the steps the compiler is currently executing. 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 sent to the controller.

4.8 Creating the program

- Create the program.

To program the example program, proceed as described in "Example program" on page 6-1.

It is now assumed that you have created the program. If you skip this point, this may result in deviations.
4.9 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, Make” command.

4.10 Generating variables and assigning process data

This section provides a general description for generating variables and assigning process data. Process data assignment for the example bus configuration is explained in later sections.

4.10.1 Generating variables

Usually, variables are either generated during program creation or they are created individually. When all the previous steps for creating a project including programming have been performed, variables have already been created.

Variables can also be generated automatically for the process data of all devices. The variable names are assigned automatically according to the following pattern:

\[ <I \text{ or } Q>_<\text{INTERBUS segment}>_<\text{INTERBUS position}>_<\text{PD name}> \]

Key:

- I or Q: I = Input; Q = Output
- PD name: Name of the process data item

For direct inputs/outputs, no INTERBUS data is required.

To generate variables, proceed as follows:

- Switch to the process data assignment workspace.
- Select the control system in the top right window. The standard configuration is displayed in the top left window, "Symbols/Variables".
- In the top left window, select the resource or program (in Figure 4-12: resource “STD_RES : ILC150_2”).

Please observe the following:

- If the resource has been selected, global variables are generated/displayed that can be used in all POUs of the project (VAR_GLOBAL).

(POU = Program Organization Unit; see online help for PC WorX Express or IEC 61131-3)
Sequence for creating a project

- In the top right window, select the device for which you would like to link the process data to variables (e.g., IB IL AO 1/U/SF in Figure 4-12).
- Select the process data item for which you would like to generate a variable (*Voltage output 0 V...10 V* in Figure 4-12).
- In the context menu for the process data item, select the "Create Variable" menu item.

Figure 4-12 Generating a variable for process data item "Voltage output 0 V...10 V"

Figure 4-13 Variable generated for process data item "Voltage output 0 V...10 V"

Figure 4-14 shows examples of automatically generated variables.

Figure 4-14 Examples of generated variables
4.10.2 Assigning process data

Process data and control variables are assigned in the process data assignment workspace.

Please observe the following:

- If the resource has been selected, global variables are generated/displayed that can be used in all POUs of the project (VAR_GLOBAL).

(POU = Program Organization Unit; see online help for PC WorX Express or IEC 61131-3)

- Switch to the process data assignment workspace to assign the variables to the process data.
- Select the control system 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 standard resource (STD_RES: ILC150_2 in Figure 4-15).
- In the top right window, select the device for which you would like to link the process data to variables (e.g., IB IL AO 1/U/SF in Figure 4-15).
- Select the process data item to be linked ("Voltage output 0 V...10 V" in Figure 4-15).
- Variables are created when the program is created. Using drag & drop, link the selected variable with one of the variables (Output_Analog in Figure 4-15) on the left-hand side.

In the bottom left window, the assignment between variables and process data is displayed.

- Repeat this procedure for all inputs to be evaluated and for all outputs to be controlled.
Sequence for creating a project

The result of the process data assignment is displayed in the following figure.

Figure 4-16  All used process data assigned to variables

System variables (e.g., ONBOARD_INPUT_BIT7) are not displayed in this process data view.
4.11 Switching to working with the system (online)

The following work can only be carried out if a system has been installed and a connection has been established between your PC with PC WorX Express and the controller.

4.12 Assigning the IP address for the controller/BootP server

By default upon delivery, ILC 1xx controllers have no preset IP address. When setting the IP address for the first time, a BootP server can be used or the address can be set manually via the serial interface. The IP address can be changed later via the serial connection or Ethernet using the PC WorX Express software.

**Bootstrap protocol (BootP)**

In an Ethernet network, BootP is used to assign an IP address to a BootP client using a BootP server. The controller (e.g., ILC 150 ETH (BootP client)) sends a Boot_Request as a broadcast in the network. The MAC address of the transmitter is sent with the Boot_Request to provide unique identification. If the BootP server has been activated in PC WorX Express, PC WorX Express responds with a Boot_Reply. PC WorX Express uses this Boot_Reply to inform the ILC 150 ETH of its IP address and subnet mask. Please ensure that:
- The BootP server knows the MAC address sent by the BootP client
- A corresponding IP address and subnet mask have been assigned in PC WorX Express for the MAC address

Once the IP data has been transferred to the ILC 150 ETH successfully, PC WorX Express sends a corresponding acknowledgment message.

**PC/network adapter**

To determine whether your network permits the default IP settings assigned by PC WorX Express (see Figure 4-4 on page 4-4), proceed as follows:
- In the Windows Control Panel, check the settings for your PC network adapter.
- If necessary, adjust these settings so that the ILC 150 ETH can be accessed in your network using the IP address used in the example project.

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

If any modifications are made to the project information 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" (see Figure 4-4 on page 4-4 and Figure 4-19 on page 4-18).
Assigning IP settings

The IP settings for the controller are made when the project is created. To set the IP address in PC WorX Express, proceed as described below.

Please note that by default upon delivery, BootP is preset on ILC 1xx controllers.

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

After assigning the IP parameters, PC WorX Express automatically creates a link via TCP/IP as a communication path to the ILC 150 ETH.

- Establish an Ethernet connection between your PC and the controller.
- In the PC WorX Express menu bar, select the "Extras... BootP/SNMP/TFTP Configuration..." menu.

![Figure 4-17](image)

Figure 4-17 "Extras... BootP/SNMP/TFTP Configuration..." menu

- Activate the "BootP Server active" checkbox.

![Figure 4-18](image)

Figure 4-18 "BootP Server active" checkbox
• Switch to the bus configuration workspace.
• Select the Inline Controller node ("ILC 150 ETH" in the example).
• Select the "IP Settings" tab in the "Device Details" window.
• Enter the MAC address of the controller (see Figure 4-19 on page 4-18). This is printed on the device and starts with 00.A0.45.

Figure 4-19  Entering the MAC address

• Perform a cold restart for the controller. To do this, switch the supply voltage off and then on again after about 2 seconds.

The controller is assigned the IP address, which is specified in the project for the controller. The following message appears in the message window in the "Bus Configurator" tab.

Figure 4-20  Message window following BootP

The IP address is now permanently stored on the controller Flash memory.
4.12.1 Address assignment with PC WorX Express via the serial interface

If the IP parameters were assigned using the BootP server, this section can be skipped.

Please note that the option to assign the IP address via the serial interface is only available for ILC 150 ETH and ILC 155 ETH Inline Controllers. The ILC 150 GSM/GPRS does not have a serial interface.

Please also note that the serial interface can only be used to assign the IP address. The Ethernet connection is the only communication path between PC WorX Express and the controller (e.g., for program download).

- Establish a serial connection between your PC and the controller using the connecting cable.

If the IP parameters were assigned using the BootP server, this section can be skipped.

- Switch to the bus configuration workspace.
- Select the controller node (e.g., “ILC 150 ETH”).
- Select the “Extended Settings” tab in the “Device Details” window.
- Set the IP address to be assigned to the controller in the “Network Settings” area under “Manual definition of the TCP/IP settings” (see Figure 4-21: 192.168.0.2). Please note that the use of a BootP server is not deactivated.

Ordering data:
Connecting cable for connecting the controller to a PC (V.24 (RS-232)) for PC WorX Express, length 3 m (order designation PRG CAB MINI DIN, Order No. 2730611).

- Click on “Send”.

Figure 4-21 Assigning the IP address for the controller
In the “Settings Communication Path” dialog box that opens, specify the serial interface on your PC at which the serial connection to the controller has been established (COM1 in the example).

Figure 4-22 “Settings Communication Path” dialog box

- Click "OK".
- The transfer of the IP address is displayed at the bottom of the “Extended Settings” tab:

Figure 4-23 “Extended Settings” tab – “Establishing connection...”

The following messages are displayed when transfer is successful:
- Offline
- Establishing connection... (yellow background)
- Online (green background)
- Service executed successfully! (green background)
- Offline
- Click on "Reset Control System" under "Activate Network Settings".

Figure 4-24 "Reset Control System" button
Sequence for creating a project

- Confirm the "Warning!" dialog box that appears with "Yes".

![Warning! Dialog Box]

Figure 4-25  "Warning!" dialog box – "Do you really want to reset the control system?"

- In the "Settings Communication Path" dialog box that opens, specify the serial interface on your PC at which the serial connection to the controller has been established (COM1 in the example, see Figure 4-22).
- Click "OK" to reset the control system.

Successful transfer of the IP address is indicated at the bottom of the "Extended Settings" tab by the following messages (see Figure 4-23):
- Offline
- Establishing connection... (yellow background)
- Online (green background)
- Service executed successfully! (green background)
- Offline

The controller needs some time before it is ready to operate again. It indicates that it is ready via the FR/RDY LEDs that are flashing cyclically.
• Switch to the "Communication" tab.

For the following check to determine whether the controller can be reached via the transferred IP address from PC WorX Express, an Ethernet connection must be established between your PC and the controller.

• Check the transferred IP address by clicking on "Test" (see Figure 4-26). The communication path has been tested successfully if a green status indicator appears in the window. If a red status indicator appears, check the communication path and, if necessary, change it as described above.

![Figure 4-26  Controller communication path tested successfully](image)

Once the test is passed, the controller can be addressed via the assigned IP address in your network. If you would like to change the network settings, this can now be done via the Ethernet connection.

### 4.13 Setting the communication path

For a control system that supports Ethernet (e.g., ILC 1xx ETH), the communication path is automatically set to "Ethernet" by default with the IP parameters of the controller specified under IP Settings. This Ethernet connection is used to send your project to the controller.

Please note that the serial interface can only be used to assign the IP address for ILC 1xx ETH controllers. The Ethernet interface is the only communication path between PC WorX Express and the controller (e.g., for program download).
4.13.1 Communication via Ethernet

An Ethernet connection to the controller is required for communication via Ethernet. Furthermore, the IP address must also be set in the controller.

- In the "Device Details" window, select the "Ethernet" interface type in the "Communication" tab.
- Enter the IP address set in the Inline Controller in the "IP Address" field or select it from the menu.

![Ethernet communication path](image)

Figure 4-27  "Ethernet" communication path

- Check the specified communication path and therefore the IP address by clicking on "Test". The communication path has been tested successfully if a green status indicator appears in the window. If a red status indicator appears, check the communication path and the settings and change, if necessary. If a red status indicator appears, an IP address may not have been assigned or the IP address needs to be changed. To assign and change the TCP/IP settings, please proceed as described in "Assigning the IP address for the controller/BootP server" on page 4-16 (general) or "Assigning the IP address for the controller/BootP server" on page 5-5 (example project).
- Save your settings in the project via "Apply".

**NOTE:** The Inline Controller can be accessed via a network using the Ethernet interface and the TCP/IP protocol. Please note that with all networked devices/computers, there is a risk that third parties may access the Inline Controller or the PC WorX Express computer and make changes, either intentionally or unintentionally. Prevent unauthorized access to the Inline Controller and your PC.
4.14 Reading INTERBUS

If you have inserted the INTERBUS devices manually, skip this section.

The bus configuration must actually be available and power must be supplied to the modules.

An INTERBUS system that is connected to an Inline Controller can be read once the communication path has been established to the Inline Controller (see "Setting the communication path" on page 4-22).

In the following example two INTERBUS modules (IB IL AO 1/U/SF and IB IL AI 2/SF-ME) are connected to the ILC 150 ETH.

Procedure

- Select the "Connected Bus" command from the "View" menu to read the connected INTERBUS system.

Establishing a connection

- Select the controller in the "Connected Bus" window.

This activates the "online" operating state and the controller reads the connected INTERBUS configuration.
Once the controller has read the connected INTERBUS system, the INTERBUS devices must be imported into the project.

- Select the controller in the "Connected Bus" window.
- Open the context menu and select the "Import to Project, With Device Description" command.

![Importing devices into the project](image)

The "Select Device" window that opens lists the modules, whose device description corresponds to that of the connected devices.

!["Select Device" window](image)

- Select the device that is actually connected in the INTERBUS system, and repeat this step until all the devices are linked to their device description.
Complete bus configuration

![Diagram of bus structure]

Figure 4-32 Complete bus configuration for the example project

Disconnecting the connection

- Disconnect the link by selecting "Offline" under "Selected Control System" in the "Connected Bus" window.

![Screenshot of "Connected Bus" window]

Figure 4-33 "Connected Bus" window: "Selected Control System - Offline"

4.15 Compiling after reading the bus topology

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

- Select the "Build, Make" command.

4.16 Creating the program

If the program has been created offline, skip this section.

To program the example program, proceed as described in "Example program" on page 6-1.

4.17 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, Make" command.
4.18 Assigning process data

If the process data has been assigned offline, skip this section.

To assign the process data, proceed as described in "Generating variables and assigning process data" on page 4-12.

4.19 Compiling and sending a project, and performing a cold restart

4.19.1 Compiling a project

- To compile a project, proceed as described in "Compiling after completing the bus topology" on page 4-10.
- Select the "Build, Make" command.

4.19.2 Sending a project

When working online, the project is written to the main memory of the controller.
- Open the project control dialog box.

If only the "Close" button is enabled in the project control dialog box that opens, there is no valid connection to the controller. In this case, check the communication path.

Before sending the project to the controller, the download options can be set.
- Click on "More...".
Download options

The "Download Options" dialog box opens.

![Download Options dialog box]

In the dialog box in Figure 4-35, 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 (ILC 150 ETH in the example).

"Options" area
- Permanent as Bootproject
- Include Sources
- Include User-Libraries
- Include Page-Layouts
- Include Backend-Code

"Bootproject" area
- "Download" button
- "Activate" button
- "Delete" button
Sequence for creating a project

4.19.3 Performing a cold restart

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

- To do so, click on "Cold" in the project control dialog box.

If the system cannot be started up, a corresponding error message appears in the message window of PC WorX Express and in Diag+.

For an explanation of the error message, please refer to the Diagnostics Guide (IBS SYS DIAG DSC UM E, Order No. 2747293).

For example, the ILC 150 ETH controller has been started up successfully if the green FR LED is on.
4.20 Operation

4.20.1 Setting the task properties

If no settings have been made for the task, a default task is used. If necessary, change the properties of the task (in the example below a default task is changed to a cyclic task).

- Switch to the IEC programming workspace.
- Switch to the project tree window in the “Hardware” tab.
- Select “STD_TSK : DEFAULT”.
- In the context menu select the “Properties...” menu item.

![Figure 4-37 Standard task context menu: "Properties"](image)

- Select the “CYCLIC” type in the “Type” tab. This selects a cyclic task.

![Figure 4-38 Standard task: "CYCLIC" task type](image)

- Apply your settings by clicking on “Apply”. 
Sequence for creating a project

- Change the task setting for "Interval", e.g., to 1000 ms.

![Task settings for STD_TSK](image)

Figure 4-39 Changing the task settings: "Interval"

- Confirm your settings with "OK".
- Compile the project, send it to the controller, and perform a cold restart.

If the process is now running too slowly, change the settings.

- In the standard task context menu select the "Settings..." menu item.

![Standard task context menu: "Settings"](image)

Figure 4-40 Standard task context menu: "Settings"

- Change the setting, e.g., to 250 ms.

![Task settings for STD_TSK](image)

Figure 4-41 Changing the task settings: "Interval"

- Compile the project, send it to the controller, and perform a cold restart.
4.20.2 Debug mode

In order to use debug mode, a program must have been created in your project. The method of operation of the program can be monitored in debug mode.

- Activate debug mode.
- Switch to the IEC programming workspace.
- Open "Physical Hardware" by clicking on the "Hardware" tab in the project tree window.
- To display the global variables, double-click on "Global_Variables".

The status of all global variables used in the program is displayed in the table that opens.

![Figure 4-42 Variables in debug mode: Global variables](image)

The following states are illustrated in Figure 4-42:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Online value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONBOARD_INPUT_BIT0</td>
<td>TRUE</td>
<td>Switch 0 ON</td>
</tr>
<tr>
<td>ONBOARD_INPUT_BIT1</td>
<td>TRUE</td>
<td>Switch 1 ON</td>
</tr>
<tr>
<td>ONBOARD_OUTPUT_BIT0</td>
<td>TRUE</td>
<td>Output 1 ON</td>
</tr>
<tr>
<td>ONBOARD_OUTPUT_BIT3</td>
<td>FALSE</td>
<td>Output 4 OFF</td>
</tr>
</tbody>
</table>

- Open the project tree by clicking on the "Project" tab in the project tree window.

To display the status of the local variables used in the POU, switch to the relevant POU and open the variables worksheet by clicking on the "Variables Worksheet" icon in the toolbar.
Local variables of the “FlashLight” POU

• Double-click on the “FlashLight” POU and then on the “Variables Worksheet” icon in the toolbar.

The status of all local variables used in the “FlashLight” POU is displayed in the table that opens.

Figure 4-43 Variables in debug mode: Local variables of the “FlashLight” POU

The following states are illustrated in Figure 4-43:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Online value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TON_1</td>
<td>–</td>
<td>Timer for the flashlight</td>
</tr>
<tr>
<td>start</td>
<td>TRUE</td>
<td>Start flag for the timer</td>
</tr>
<tr>
<td>control</td>
<td>1</td>
<td>Variable</td>
</tr>
<tr>
<td>TimeFlashLight</td>
<td>0.200</td>
<td>Time value for the flashlight, input variable (flash speed)</td>
</tr>
</tbody>
</table>

Local variables of the “Main” POU

• Double-click on the “Main” POU and then on the “Variables Worksheet” icon in the toolbar.

The status of all local variables used in the “Main” POU is displayed in the table that opens.

Figure 4-44 Variables in debug mode: Local variables of the “Main” POU
The following states are illustrated in Figure 4-44:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Online value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActTimeFlashLight</td>
<td>0.200</td>
<td>Specification of the flash speed as 200 ms</td>
</tr>
<tr>
<td>FlashLight</td>
<td>–</td>
<td>Function block for controlling the &quot;FlashLight&quot;</td>
</tr>
</tbody>
</table>

The program status of a POU is displayed by opening the corresponding worksheet (in Figure 4-45 under "Main" in the project tree window or under "Main:Main" on the tab in the worksheet).

4.20.3 PLC stop/run

If the PLC is set to STOP, all outputs are set to their safe state. On control system startup, process values are output again.
5  Example project for an INTERBUS system

The creation of a project is described in detail in Section 4, "Sequence for creating a project".

This section covers:
– All steps to be taken in reference to the corresponding section in Section 4, "Sequence for creating a project"
– Project-specific settings
– Information and special notes for an INTERBUS project

5.1  Describing the project

In the following, an example project is developed using function block diagram (FBD) and structured text (ST).

In order to obtain the best possible results, please use the same identifiers and names as used in this user manual.

Project name  Quickstart_PCWExpress

Project hardware
– Control system: ILC 150 ETH Inline Controller
– I/O modules: IB IL AO 1/U/SF, IB IL AI 2/SF-ME

Figure 5-1  Example bus configuration

Please note that the example bus configuration shown above contains Inline I/O modules, which are not wired in the example in this Quick Start Guide. The modules are used in "Reading INTERBUS" on page 4-24 and "Reading INTERBUS" on page 5-6 to show how a connected INTERBUS structure is read in. An example project using the illustrated modules is described in UM QS EN ILC 150 CONSTR.KIT.
5.2 Sequence for creating the INTERBUS project

The sequence for creating the INTERBUS project is shown in Figure 5-2 on page 5-3.

When implementing the project some of the tasks can be performed offline (without a connection to the INTERBUS system).

All tasks related to communication must be performed online (with a connection to the INTERBUS system).

In Figure 5-2 on page 5-3 and in the description in this section, it is assumed that the system has been fully installed and all tasks are performed online. This is the quickest way to start up an INTERBUS system.

If the INTERBUS system has not yet been fully installed or if you would like to perform as many preparatory tasks as possible offline, follow the sequence described in Section 4, "Sequence for creating a project".

### Table 5-1 Inputs and outputs used including process data and assigned variables

<table>
<thead>
<tr>
<th>Device</th>
<th>Input/output</th>
<th>Signal at</th>
<th>Variable</th>
<th>Process data item</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILC 150 ETH</td>
<td>Input I1</td>
<td>Connector 3 terminal point 1.1</td>
<td>ONBOARD_INPUT_BIT0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Input I2</td>
<td>Connector 3 terminal point 1.1</td>
<td>ONBOARD_INPUT_BIT1</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Output Q1</td>
<td>Not wired</td>
<td>ONBOARD_OUTPUT_BIT0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Output Q2</td>
<td>Not wired</td>
<td>ONBOARD_OUTPUT_BIT3</td>
<td>–</td>
</tr>
<tr>
<td>IB IL AO 1/U/SF</td>
<td>Output O1</td>
<td>Not wired</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>IB IL AI 2/SF-ME</td>
<td>Input I9</td>
<td>Not wired</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>
Example project for an INTERBUS system

Figure 5-2  Sequence for creating the INTERBUS project

- **OFFLINE**
  - Start
  - Create new project
  - Specify project information
  - Check/modify IP settings for controller
  - System installed?
    - Yes: Set communication path
    - No: Manually insert INTERBUS devices
  - Create program
  - Assign process data
  - Assign IP address for controller/BootP server

- **ONLINE**
  - INTERBUS devices inserted?
    - No: Read INTERBUS
    - Yes: Program created?
      - Yes: Operation (end)
      - No: Process data assigned?
        - Yes: Create program
        - No: Assign process data
  - Compile and send project, and perform cold restart

7632A007
5.3 Creating a new project

See also “Creating a new project” on page 4-3.

- Select the "New Project..." command from the "File" menu.
- Select the control system (here: ILC 150 ETH Rev. > 01/2.00) and confirm your selection with "OK".
- Select the "File, Save Project As/Zip Project As..." command.
- Specify the project name "Quickstart_PCWExpress" and save the project.

5.4 Specifying project information

See also “Specifying project information” on page 4-4.

- Switch to the bus configuration workspace.
- Adapt the project information to your project.

5.5 Checking/modifying IP settings for the controller

See also “Checking/modifying IP settings for the controller” on page 4-6.

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

**NOTE:** If any modifications are made to the project information 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 as the IP settings.

Adapt these settings, if necessary.

- 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, if necessary.
- If an IP address has still not been assigned, assign one according to "Assigning the IP address for the controller/BootP server" on page 5-5.

The IP parameters that are assigned here for the controller are also implemented as the IP parameters for the communication path via TCP/IP.
5.6 Assigning the IP address for the controller/BootP server

An IP address must first be assigned to the controller in order to enable communication. For the ILC 150 ETH controller BootP is activated by default upon delivery for assigning the IP address via Ethernet. In this case, when setting the IP address for the first time a BootP server can be used. The corresponding procedure is described below. For all other options, please refer to "Assigning the IP address for the controller/BootP server" on page 4-16.

- Establish an Ethernet connection between your PC and the controller.
- In the menu bar, select the "Extras, BootP/SNMP/TFTP-Configuration..." menu.
- Activate the "BootP Server active" checkbox.
- Switch to the bus configuration workspace.
- Select the controller node (e.g., "ILC 150 ETH").
- Select the "IP Settings" tab in the "Device Details" window.
- Enter the MAC address of the controller. This is printed on the device and starts with 00.A0.45.
- Perform a cold restart for the controller. To do this, switch the supply voltage off and then on again after about 2 seconds.

The controller is assigned the IP address, which is specified in the project for the controller. The IP address is now permanently stored on the controller Flash memory.
5.7 Setting the communication path

See also “Setting the communication path” on page 4-22.

For a control system that supports Ethernet (e.g., ILC 1xx ETH), the communication path is automatically set to “Ethernet” by default with the IP parameters of the controller specified under IP Settings. This Ethernet connection is used to send your project to the controller.

Please note that the serial interface can only be used to assign the IP address for ILC 1xx ETH controllers. The Ethernet interface is the only communication path between PC WorX Express and the controller (e.g., for program download).

For additional information, please refer to “Setting the communication path” on page 4-22.

5.8 Reading INTERBUS

See also “Reading INTERBUS” on page 4-24.

- Select the “Connected Bus” command from the “View” menu to read the connected INTERBUS system.
- Select the controller in the "Connected Bus" window ("ILC 150 ETH" controller selected in the example).

This activates the “online” operating state and the controller reads the connected INTERBUS configuration.

Once the controller has read the connected INTERBUS system, the INTERBUS devices must be imported into the project.
- Open the context menu for the controller and select the "Import to Project, With Device Description" command.

The "Select Device" window that opens lists the modules, whose device description corresponds to that of the connected devices.
- Select the device that is actually connected to the INTERBUS system, and repeat this step until all the devices are linked to their device description. According to the example, the connected I/O modules are: IB IL AO 1/U/SF and IB IL AI 2/SF-ME.
5.9 Compiling after completing the bus topology

- Disconnect the link to the controller by selecting "Offline" under "Selected Control System" in the "Connected Bus" window.

![Complete bus configuration for the example project](image)

Figure 5-3 Complete bus configuration for the example project

---

5.10 Creating the program

To program the example program, proceed as described in "Example program" on page 6-1.

![Make](image)

It is now assumed that you have created the program. If you skip this point, this may result in deviations.

---

5.11 Compiling after creating the program

- Select the "Build, Make" command.
5.12 Assigning process data

See also “Generating variables and assigning process data” on page 4-12.

Please note that the process data assignments described in this section for the example in “Example program” on page 6-1 do not have to be made. They are not required for the example. The process data assignments described in this section are simply intended to show an example of the procedure for assigning process data.

- Switch to the process data assignment workspace to assign the variables to the process data.
- Select the control system 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 standard resource (in the example: STD_RES : ILC150_2 (in Figure 5-4)).
- In the top right window, select the device for which you would like to link the process data to variables (IB IL AI 2/SF-ME in Figure 5-4).
- Select the process data item to be linked ("AI 1 Voltage" in Figure 5-4).
- Variables are created when the program is created. Using drag & drop, link the selected variable with one of the variables (Input_Analog in Figure 5-4) on the left-hand side.
  - If you would like to link further process data but no corresponding variables have been created yet, select "Create Variable" in the context menu.

The created variable is displayed in the bottom left window.
- Repeat this procedure for all inputs to be evaluated and for all outputs to be controlled.

The result of the process data assignment is displayed in the following figure.

![Figure 5-4 All used process data assigned to variables](image-url)

System variables (e.g., ONBOARD_INPUT_BIT7) are not displayed in this process data view.
5.13 Compiling and sending a project, and performing a cold restart

See also "Compiling and sending a project, and performing a cold restart" on page 4-27.

5.13.1 Compiling a project

- Select the "Build, Make" command.

5.13.2 Sending a project

- Open the project control dialog box.

If only the "Close" button is enabled in the project control dialog box that opens, there is no valid connection to the controller. In this case, check the communication path.

Before sending the project to the controller, set the download options.
- Click on "More...".

The "Download Options" dialog box opens.
- In the "Options" area, activate the "Permanent as Bootproject" checkbox.
- Close the "Download Options" dialog box.
- In the project control dialog box click on "Download".

The project is now in the main memory of the controller.

5.14 Operation

See "Operation" on page 4-30.
6 Example program

An example project is described in the previous section (Section 5, "Example project for an INTERBUS system").
This section describes the procedure for creating the program in the software.

6.1 Program description

A flashlight is programmed in the example program. The flash speed can be set to two levels. The program is created in function block diagram (FBD) and structured text (ST).
The program controls the following sequence:
Outputs 1 and 4 of the controller are controlled alternately. This is indicated by LEDs Q1 and Q4, which flash alternately. Depending on the states of inputs 1 and 2 of the controller, the flash speed can be set. The states of inputs 1 and 2 are displayed via the I1 and I2 LEDs.
The entire program is implemented in the "Main" and "Flashlight" POUs. The ONBOARD_INPUT_BIT0 and ONBOARD_INPUT_BIT1 variables (system variables) map the status of the inputs.
The time for controlling the flash speed is set within the "Main" POU according to the ONBOARD_INPUT_BIT0 and ONBOARD_INPUT_BIT1 input variables. In addition, the FlashLight user-defined function block is called.
The ONBOARD_OUTPUT_BIT0 and ONBOARD_OUTPUT_BIT3 output variables are controlled within the "FlashLight" POU.
The basic steps for creating this program are described below.
### 6.2 Function blocks used in the example

#### AND (ANDing) function block

Table 6-1  AND (ANDing) function block

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Variable name</th>
<th>Data type</th>
<th>Usage</th>
<th>Initial value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IN1</td>
<td>ONBOARD_INPUT_BIT0</td>
<td>BOOL</td>
<td>VAR_GLOBAL</td>
<td></td>
<td>Switch position switch 1</td>
</tr>
<tr>
<td>IN2</td>
<td>ONBOARD_INPUT_BIT1</td>
<td>BOOL</td>
<td>VAR_GLOBAL</td>
<td></td>
<td>Switch position switch 2</td>
</tr>
<tr>
<td>OUT</td>
<td></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

#### SEL_TIME selection function block

Table 6-2  Binary selection of a time value

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Variable name</th>
<th>Data type</th>
<th>Usage</th>
<th>Initial value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td></td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>IN0</td>
<td>ActTimeFlashLight</td>
<td>TIME</td>
<td></td>
<td>500 ms</td>
<td>Flash speed: Initial value</td>
</tr>
<tr>
<td>IN1</td>
<td>time#200ms</td>
<td>VAR</td>
<td></td>
<td>200 ms</td>
<td>Time value</td>
</tr>
<tr>
<td>OUT</td>
<td>ActTimeFlashLight</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>Flash speed: Current value</td>
</tr>
</tbody>
</table>

#### FlashLight user-defined function block

Table 6-3  Binary selection of a time value

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Variable name</th>
<th>Data type</th>
<th>Usage</th>
<th>Initial value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TimeFlashLight</td>
<td>ActTimeFlashLight</td>
<td>TIME</td>
<td>VAR_INPUT</td>
<td>–</td>
<td>Flash speed: Current value</td>
</tr>
</tbody>
</table>
6.3 Programming

Aim of this section
This section shows you how to carry out the following in PC WorX Express:
- Start programming (see Section 6.3.2)
- Enter a comment (see Section 6.3.3)
- Create variables (see Section 6.3.4)
  - Create user variables (see Section 6.3.5)
  - Create system variables (see Section 6.3.6)
- Add a function block from the edit wizard (see Section 6.3.6)
- Create a user-defined function block (see Section 6.3.7)
- Add a user-defined function block (see Section 6.3.8)
- Delete variables (see Section 6.3.9)
- Set the initial value (see Section 6.3.10)

The following section shows the complete example program (see Section 6.3.1).
6.3.1 Example program

In order to create the complete example program, proceed as described in the sections below.

For additional information about the PC WorX Express software, please refer to the online help.

"Main" POU – Requesting switches 0 and 1 and calling the "FlashLight" user-defined function block

"FlashLight" POU – Flashlight

Figure 6-1 "Main" POU (function block diagram (FBD))

Figure 6-2 "FlashLight" POU (structured text (ST))
6.3.2 Starting programming

For programming, proceed as follows:

• Switch to the IEC programming workspace.
• Double-click on "Main" (A) in the project tree window to activate the IEC programming interface.
• Click on the desired position in the "Main" worksheet (B) where you want to add the programming (e.g., the comment or the function block).

Figure 6-3 IEC programming workspace
6.3.3 Entering a comment – Title of the POU

- Click on the desired position in the worksheet to set a cursor.
- Select the “Text (Comment)…” command from the “Objects” menu to add a comment.

In the "Comment" dialog box that opens, enter the desired comment for the "Main" POU ("Select time for flashlight / Blinkgeschwindigkeit auswählen" in the example).
6.3.4 Creating variables

The onboard inputs and outputs of the ILC 150 ETH are mapped to system variables. The variable for specifying the flash speed in the example (ActTimeFlashLight) is created as a user variable.

Scope of validity/usage

In PC WorX Express, the use of a variable must be declared.

A variable that is used within a POU is declared as a local variable using one of the following keywords: VAR, VAR_INPUT, VAR_OUTPUT or VAR_IN_OUT. A variable that is used in the entire project is declared as a global variable using the keyword VAR_GLOBAL.

Make sure that the "Hide external variables" option is selected in the "Options" dialog box ("Extras, Options..." menu).

Figure 6-6 Options – Hide external variables

When creating variables using the "Variable Properties" dialog box, the scope of validity of a new variable that is to be created is simply declared via the "Usage" option.

Figure 6-7 Variable Properties – Usage

When selecting a variable that has already been declared, its validity can be determined via the icons in the "Name" pull-down menu.

Figure 6-8 Variable Properties – Icons

: Local variable

: Global variable
6.3.5 Creating user variables – Assigning the default flash speed in the example

- Click on the desired position in the worksheet to set a cursor.
- Click on “Variable” to add a variable to the worksheet. Alternatively, select the “Variable...” command from the context menu for the cursor.
- In the “Variable Properties” dialog box, enter the variable name “ActTimeFlashLight”.
- Select “TIME” as the data type.
- “VAR” must be set under “Usage”.

![Variable Properties - User variable](image)

- Confirm your entries with “OK”.

Assigning the initial value

- Set a cursor in the worksheet to the left of the “ActTimeFlashLight” variable.
- In the context menu for the cursor, select the “Variable...” command.
- Enter the value “time#500ms” in the “Name” field.
- Confirm your entry with “OK”.

A variable with a fixed value has now been defined. Connect this variable to the “ActTimeFlashLight” variable.

![Connect](image)

- Click on “Connect”.
- In the worksheet, first click on the output of the “time#500ms” variable and then on the input of the “ActTimeFlashLight” variable.

An initial value is now assigned to the “ActTimeFlashLight” variable.
6.3.6 Creating system variables – Function blocks for selecting inputs in the example

- Click on the desired position in the worksheet to set a cursor.
- Select the "AND" function block by double-clicking on it in the "Edit Wizard" window. The function block is inserted in the worksheet.

According to Figure 6-1 on page 6-4, input I1 of the ILC 150 ETH is to be evaluated for the first input signal. The onboard inputs and outputs of the ILC 150 ETH are mapped to system variables.
- Double-click on the first input parameter of the function block to specify variable properties.
- Select the corresponding variable under "Name". For input I1 of the ILC 150 ETH, the corresponding variable is ONBOARD_INPUT_BIT0.

Figure 6-10 Variable Properties – System variable
The following figure shows all the settings for the ONBOARD_INPUT_BIT0 variable. The variable is defined as a global variable (VAR_GLOBAL) under "Usage".

![Variable Properties](image)

Figure 6-11 Variable Properties (system variable)

- Confirm your entries with "OK".

Using the method described above, assign the ONBOARD_INPUT_BIT1 variable to the second output of the "AND" function block.

**SEL_TIME function block**

The "SEL_TIME" function block is then added.

- Click on the desired position in the worksheet to set a cursor.
- Select the "SEL_TIME" function block by double-clicking on it in the "Edit Wizard" window.

The function block is inserted in the worksheet.

- Connect the output of the "AND" function block to input G of the "SEL_TIME" function block.
- Double-click on input IN0 of the function block to assign the "ActTimeFlashLight" variable.
• Select the "ActTimeFlashLight" variable.
• Set "Usage" to "VAR".

![Variable Properties - Usage](image1)
Figure 6-12 Variable Properties – Usage

• Confirm your entries with "OK".
• Double-click on input IN1 of the function block to assign the "time#200ms" time value.
• Set "Usage" to "VAR".
• Enter "time#200ms" under "Name".

![Variable Properties - Time value](image2)
Figure 6-13 Variable Properties – Time value

• Confirm your entries with "OK".
• Repeat the procedure described above for the output of the "SEL_TIME" function block.

The first part of the "Main" POU is now programmed.
### 6.3.7 Creating a user-defined function block

To create the "FlashLight" user-defined function block, proceed as follows.

The "FlashLight" function block is programmed in structured text (ST). The individual sections of this POU are commented on in the corresponding "FlashLight" worksheet (see Figure 6-2 on page 6-4).

For additional notes about ST, please refer to the online help for the PC WorX Express software.

#### Programming the "FlashLight" POU

- In the project tree window, open the context menu for "Logical POUs".
- Select the "Insert..., Function Block" menu item.

![Image of "Logical POUs" context menu – "Insert..., Function Block"]

- In the "Insert" dialog box, enter the name "FlashLight" for the function block to be created.
- Under "Type" select "Function Block" and under "Language" select "ST".

![Image of "Insert" dialog box – Creating a function block]

- Confirm your entries with "OK".
- Switch to the "FlashLight" worksheet by double-clicking on the "FlashLight" POU in the project tree window.
- Create the programming for this POU (see Figure 6-2 on page 6-4).
The following variables must be created/assigned for the "FlashLight" POU:

- ONBOARD_OUTPUT_BIT0 and ONBOARD_OUTPUT_BIT3 system variables (local outputs OUT1/OUT4 of the ILC 150 ETH) are labeled "VAR" within the "FlashLight" POU.

Table 6-4 Variables of the "FlashLight" POU

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Usage</th>
<th>Initial value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TON_1</td>
<td>TON</td>
<td>VAR</td>
<td>–</td>
<td>Flashlight timer</td>
</tr>
<tr>
<td>start</td>
<td>BOOL</td>
<td>VAR</td>
<td>FALSE</td>
<td>Timer start flag</td>
</tr>
<tr>
<td>TimeFlashLight</td>
<td>TIME</td>
<td>VAR_INPUT</td>
<td>–</td>
<td>Time for flashlight (ms), input variable (flash speed)</td>
</tr>
<tr>
<td>control</td>
<td>INT</td>
<td>VAR</td>
<td>0</td>
<td>Control variable</td>
</tr>
<tr>
<td>ONBOARD_OUTPUT_BIT0</td>
<td>BOOL</td>
<td>VAR</td>
<td>–</td>
<td>Local output OUT1</td>
</tr>
<tr>
<td>ONBOARD_OUTPUT_BIT3</td>
<td>BOOL</td>
<td>VAR</td>
<td>–</td>
<td>Local output OUT4</td>
</tr>
</tbody>
</table>

Compiling a project

To add the function block to the "Main" POU properly, the project must be compiled.

Once the programming has been created, compile your project.

- Click on the "Make" icon in the toolbar.

If the programming has been created without any errors, the following message appears in the message window following error-free compilation:

Figure 6-16 Error-free compilation

At the end of the compiling process, the "FlashLight" function block can be selected in the edit wizard.

Figure 6-17 Edit Wizard - "FlashLight" function block
6.3.8  Adding a user-defined function block

- Switch to the "Main" worksheet.
- Carry out the following steps according to the procedure described above (see Figure 6-1 on page 6-4).
- Place the cursor in the desired position in the worksheet.
- Add a corresponding comment for the "FlashLight" function block.
- Add the "ActTimeFlashLight" variable.
- Add the "FlashLight" user-defined function block.
- Connect the "ActTimeFlashLight" variable to the "FlashLight" function block.

The programming of the "Main" POU is now complete.

Variables of the "Main" POU:

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Usage</th>
<th>Initial value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActTimeFlashLight</td>
<td>TIME</td>
<td>VAR</td>
<td>–</td>
<td>Specification of the flash speed</td>
</tr>
<tr>
<td>FlashLight</td>
<td>FlashLight</td>
<td>VAR</td>
<td>–</td>
<td>Function block for controlling the &quot;FlashLight&quot;</td>
</tr>
</tbody>
</table>

6.3.9  Deleting variables

If variables are created during program creation, they may be created in different worksheets. When variables are deleted or renamed in the program, they are not automatically deleted or renamed in the other worksheets.

An error message is therefore displayed when compiling.
- In this case, delete the variables that are not used from the areas marked in the tabs in Figure 6-18 ("Global_Variables" and "MainV:Main").

![Figure 6-18 Deleting variables](image-url)
6.3.10 Setting the initial value

If you would like to set an initial value, proceed as follows:

- Switch to the IEC programming workspace.
- Double-click on the “MainV:Main” item.
- Set the initial value of the variables (here: set ActTimeFlashLight to the initial value “time#50ms”).

![Figure 6-19 Setting the initial value (here: ActTimeFlashLight = time#50ms)](image)

6.4 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, Make" command.

6.5 Assigning process data

In the example described above, no process data is to be connected. If your application deviates from the example described, the process data and control variables should be assigned in the process data assignment workspace.

For instructions on how to carry out process data assignment, please refer to “Generating variables and assigning process data” on page 4-12.

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

Now the project has been completed. In order to start up the project, compile it, send it to the control system, and perform a cold restart.

- Proceed as described in "Compiling and sending a project, and performing a cold restart" on page 4-27.

The project has been successfully started.
A  Additional software functions

A 1  Setting the realtime clock

Some control systems have an internal system clock (e.g., ILC 1xx...). In the "Extended Settings" tab in the bus configuration workspace ("Device Details" window), the time and date can be set for the internal system clock of the control system.

The current control system time is read and displayed every time the "Extended Settings" tab is opened. The display then indicates this value until the "Extended Settings" tab is opened again.

If you want to apply the system time and date from your PC, proceed as follows:
There must be a connection between the PC and control system.

• Switch to the bus configuration workspace.
• Select the control system node (e.g., "ILC 150 ETH").
• Select the "Extended Settings" tab in the "Device Details" window.
• Click on "System Time".
• Then immediately click on "Send" to transmit the data in the "Real-time Clock Settings" area to the control system.

If you want to set other values, proceed as follows:
• Enter the values for the time and date.
• Click on "Send" to transmit the data in the "Real-time Clock Settings" area to the ILC 150 ETH.

Figure A-1  Realtime clock settings
A 2 Replacing a control system (hardware replacement)

To simply replace one control system with another, a wizard is provided for replacing complex devices. It may be necessary to replace the control system, e.g., if you have created the programming for a project with an ILC 150 ETH, but would like to use an ILC 155 ETH for the actual project.

- Switch to the bus configuration workspace.
- In the “Bus Structure” window select the control system to be replaced (ILC 150 ETH in Figure A-2).
- Open the context menu for the control system.
- Select the “Replace...” menu item.

The wizard opens.

- Confirm the window that opens with “Next”.

The window displays a selection of all the control systems, which can replace the control system that is currently used in the project.

If the window is empty, the control system that is currently used cannot be replaced with another one.
Replacing a control system (hardware replacement)

- Select the new control system to be used (e.g., ILC 155 ETH).

![Figure A-3 Selecting the new control system](image)

- Confirm your selection with "Next".

  The wizard replaces the device.

  The wizard indicates whether the control system was replaced successfully and also displays special information.

![Figure A-4 Completing replacement](image)

- Confirm replacement with "Finish".
The bus configuration with the new control system is displayed.

In the example an Inline Controller (ILC 150 ETH) has been replaced by another Inline Controller (ILC 155 ETH).

If you replace a control system with one of a different type (e.g., ILC 150 ETH with ILC 150 GSM/GPRS), the following message is displayed:

- In this case, check the resource settings and correct the settings, if necessary.
- Switch to the IEC programming workspace.
- Switch to the "Physical Hardware" view.
- Select the "Settings" menu item in the context menu of the standard resource (STD_RES : ILC150GSM_2 in the example).
Replacing a control system (hardware replacement)

- In the "Resource settings for ..." dialog box that opens, select the "ILC 1xx v..." entry in the pull-down menu for the "Build settings" (see Figure A-8).

![Resource settings for ILC1500SM_2](image)

Figure A-8 "Resource settings for..." dialog box: "Build settings"

- Confirm your selection with "OK".

A message then informs you that under certain circumstances other parts of your project may have to be adapted.
- Confirm the warning with "OK".
- Compile the project to detect any errors.
- If you have used special POUs or special libraries, adapt them accordingly.
- Compile the project.
A 3 Device description files

Device description files are FDCML or GSD files, which provide a complete description of a device. If a device description file required for your project is missing from the device catalog, import the necessary file.

A 3.1 Phoenix Contact device description files

When PC WorX Express is reinstalled, 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 PC WorX Express service packs, new device description files are installed in folder "...\FDCML10\..\Phoenix Contact". These new descriptions are not included in the device catalog yet. They must be imported where necessary.

- Select "Phoenix Contact" in the "Device Catalog" window in PC WorX Express.
- Open the context menu and select "Import Device...".

![Device Catalog: Import Device](image)

- Select the device description file. If you have used the suggested standard installation, the files are located in directory C:\Documents and Settings\All Users\Documents\FDCML10\xxx\Phoenix Contact. xxx = System (e.g., ETHERNET, INTERBUS)

![Directory for the device description files](image)

- Confirm your selection.

The "Message Window" indicates whether the device has been imported successfully.
A 3.2 Creating device description files

If a device description file is not available, one can be created. For this, a device description editor is included on the AUTOMATIONWORX Software Suite CD.

A 4 Visualization

The data generated in PC WorX Express can be used in other programs, e.g., to visualize processes.

In order to use the data, AX OPC Server and/or a visualization software program are required (see "AX OPC Server and WebVisit" on page B-1).

A 5 INTERBUS topology

PC WorX Express can be used to automatically detect INTERBUS topologies. The INTERBUS structure can be viewed in the "INTERBUS Topology" window.

- Call the "INTERBUS Topology" window via the "INTERBUS Topology" menu item in the "View" menu.
- Refresh the display using the context menu for the "INTERBUS Topology" window.
- In the "Print Selection" window, select "Order of device data".

The following INTERBUS structure is displayed for the example bus configuration:

![INTERBUS Topology](image)

Figure A-11 INTERBUS topology for an INTERBUS system

- PC WorX Express offers the option of printing out the entire bus topology.
  - Activate the context menu in the "INTERBUS Topology" window.
  - In the context menu select the "Print..." menu item.

The other steps for printing depend on your printer installation.
B  AX OPC Server and WebVisit

B 1  AX OPC Server

For information about using AX OPC Server, please refer to the corresponding documentation.

The AX OPC Server includes:
- The “OPC Configurator” for establishing a connection between PC WorX Express and an OPC client (e.g., Visu+).
- An “OPC Test Client” for testing the connection.

The AX OPC Server is also installed as a demo version. To register your license, proceed as follows:
- Start the “OPC Test Client”.
- In the taskbar, activate the context menu for the “OPC Test Client” icon.
- Select the “Register” menu item and enter the registration code.

B 1.1  Preparatory tasks in PC WorX Express

- In order to use a variable in AX OPC Server, activate the “OPC” checkbox:
  - When creating variables in the “Variable Properties” window (Figure B-2)
  - In the variables worksheet (Figure B-3)
The OPC data is also transmitted when the project is sent to the control system.

### B 1.2 OPC Configurator

- Start the "OPC Configurator" program.

The configurator commands can be accessed via context menus. The name of the resource ("TestResource in Figure B-4) can be adapted accordingly to the requirements of your project.

- Select the resource type.
• Open the context menu for the "TestResource" entry.
• Select the "Settings..." menu item.

Figure B-5  "Settings..."

• Under "IP Address" enter the IP address for the control system for which you would like to use the OPC data.

Figure B-6  Specifying the IP address for the control system

This completes the configuration for using OPC data through an OPC client.

Figure B-7  OPC configuration completed
B 1.3  **OPC Test Client**

The Test Client can be used to test the OPC configuration.

- Open the "OPC Test Client" program.
- Set the connection data in the window that opens.

![OPC Test Client: Connection data](image)

- Click on "Browse...".
- In the "Select an OPC Server" window that opens, select the OPC server ("AUTOMATIONWORX OPC-Server 2.1" in Figure B-9).

![Selecting the OPC server](image)

- Confirm your selection with "OK".

The OPC server and the PC where the OPC server is located are entered.

![Connection to OPC server selected](image)

- Confirm your selection with "Connect".
• Activate the context menu for "Private Groups".

![Figure B-11 Private Groups](image)

Figure B-11 Private Groups

• In the window that opens, enter the name of the group (e.g., "Quickstart_PCWExpress") and confirm your entry with "OK".

![Figure B-12 Entering the name of the group](image)

Figure B-12 Entering the name of the group
- Activate the context menu for the created group.
- Select the "Add all Items" menu item.

Figure B-13  Add all Items

All variables defined as OPC variables are displayed.

Figure B-14  OPC variables
The connection between PC WorX Express, OPC, and Test Client can now be tested via the context menu for the individual variables.

Figure B-15  OPC variables

The result is displayed at the bottom of the window.
The WebVisit software is used to visualize the global variables of the Inline Controller used under PC WorX Express. WebVisit is a software tool used to generate web pages. The software runtime component is a web server, which is stored on the control system. The variable values are actually visualized via a Java-compatible standard browser.

To visualize variables from your PC WorX Express project in WebVisit, activate the "PDD" checkbox:
- In the "Variable Properties" window (Figure B-16)
- In the variables worksheet (Figure B-17)

When compiling the project, a file called "pdd.csv" is generated, which is used by WebVisit for the visualization.

In WebVisit, enter the PC WorX Express project used (recommended) or the corresponding pdd.csv file. For standard installation:
- The project is located in directory
  C:\Documents and Settings\All Users\Documents\PC WORX\Projects
- The corresponding pdd.csv file is located in directory C:\Documents and Settings\All Users\Documents\PC WORX\Projects\xxx\C\STD_CNF\STD_RES
  (xxx = Project name)

For information about using WebVisit, please refer to the corresponding documentation.
Overview of Phoenix Contact control systems

Table C-1 provides an overview of Phoenix Contact control systems together with the processor types and hardware and firmware versions used, and the resulting functions.

When creating a new project, several versions are provided for different control systems. The versions differ with regard to the hardware and firmware version used.

If a control system is inserted in a project without a template, depending on the hardware and firmware version, the processor type of the control system used must be specified. Select the processor type according to Table C-1.

Some of the functions depend on the type of processor used. If a project is created in PC WorX Express and a processor type is selected, which does not correspond to the processor used on your control system, this may result in the following:

- Fewer functions can be used even though your processor supports more
- Functions are displayed under PC WorX Express, which your processor does not support

Example:

You have an ILC 150 ETH Inline Controller.

Marked version 01/2.00

This corresponds to:

- Hardware version 01
- Firmware version 2.00

Select the ILC 150 ETH with firmware Version 2.00 according to the table.

If a controller is used with a firmware version, which is not listed in the table, select the older firmware version.

<table>
<thead>
<tr>
<th>Control system</th>
<th>Processor</th>
<th>From HW</th>
<th>From FW</th>
<th>Master/slave function</th>
<th>Supported by AX OPC Server</th>
<th>Function supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>ILC 150 ETH</td>
<td>eCLR</td>
<td>01</td>
<td>2.00/0.10</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>ILC 150 GSM/GPRS</td>
<td>eCLR</td>
<td>01</td>
<td>2.10/0.10</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>ILC 155 ETH</td>
<td>eCLR</td>
<td>01</td>
<td>2.00/0.10</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>