Every conceivable measure has been taken to ensure the accuracy and completeness of this documentation. However, as errors can never be fully excluded, we always appreciate any information or suggestions for improving the documentation.

E-Mail: documentation@wago.com

We wish to point out that the software and hardware terms as well as the trademarks of companies used and/or mentioned in the present manual are generally protected by trademark or patent.
# Table of Contents

1 Notes about this Documentation ................................................................. 7
   1.1 Scope of Validity ................................................................................... 7
   1.2 Copyright ............................................................................................ 7
   1.3 Symbols ................................................................................................ 8
   1.4 Number Notation ............................................................................... 10
   1.5 Font Conventions .............................................................................. 10

2 Important Notes ....................................................................................... 11
   2.1 Legal Bases ......................................................................................... 11
      2.1.1 Subject to Changes ...................................................................... 11
      2.1.2 Personnel Qualification ............................................................ 11
   2.2 Safety Advice (Precautions) ............................................................... 12

3 Overview ..................................................................................................... 13

4 Requirements ............................................................................................. 15
   4.1 System Requirements ......................................................................... 15

5 Licenses ....................................................................................................... 16

6 Installation .................................................................................................. 17

7 Starting ........................................................................................................ 18
   7.1 Purchasing a License .......................................................................... 19
   7.2 Activating the License ........................................................................ 19
   7.3 Activating Automatic Updates ............................................................ 21

8 Graphical User Interface ........................................................................... 22
   8.1 Start View (Projects and Templates) ..................................................... 22
   8.2 Main View (Configuring, Parameterization, Programming) ............... 24
      8.2.1 Quick Access Toolbar ................................................................ 25
      8.2.2 Menu Ribbon ............................................................................... 26
      8.2.2.1 “START” Tab ........................................................................ 26
      8.2.2.2 “VIEW” Tab .......................................................................... 30
      8.2.2.3 “NETWORK” Tab ................................................................. 32
      8.2.2.4 “DEVICE” Tab ...................................................................... 34
      8.2.2.5 “PROGRAM” Tab .................................................................. 36
      8.2.2.6 “DEBUG” Tab ....................................................................... 42
      8.2.2.7 “I/O CHECK” Tab ................................................................. 46
      8.2.2.8 Other Tabs ............................................................................. 48
   8.2.3 Workspace ...................................................................................... 49
      8.2.3.1 General Operating Elements ................................................. 50
      8.2.3.2 General Status Display for Connections ................................. 50
      8.2.3.3 “Network/Devices” Workspace ............................................. 53
         8.2.3.3.1 Network View ................................................................ 54
         8.2.3.3.1.1 Tiles .......................................................................... 55
         8.2.3.3.1.2 Connectors ................................................................. 57
         8.2.3.3.1.3 Network/Bus Connections ........................................ 59
      8.2.3.3.2 Fieldbus-Specific Configurators ....................................... 60
         8.2.3.3.2.1 MODBUS Configurator ............................................. 61
8.2.3.3.2.2 CANopen Configurator ........................................................ 66
8.2.3.3.2.3 PROFIBUS Configurator ..................................................... 80
8.2.3.3.3 Device Detail View ................................................................... 84
8.2.3.3.3.1 I/O Mapping ......................................................................... 87
8.2.3.4 “Programming” Workspace .......................................................... 89
8.2.3.4.1 Program Editor ........................................................................ 90
8.2.3.4.2 Library Manager ....................................................................... 91
8.2.3.4.3 Task Configuration .................................................................. 93
8.2.3.4.4 “PLC_Task” Main Task ............................................................ 94
8.2.3.4.5 Breakpoints ............................................................................... 97
8.2.3.4.6 Call Stack ................................................................................ 97
8.2.3.4.7 Cross Reference List ................................................................. 98
8.2.3.5 User-Defined Workspaces ............................................................. 99
8.2.3.6 Panels ........................................................................................... 100
8.2.3.6.1 Device Structure ...................................................................... 101
8.2.3.6.2 Program Structure ................................................................... 102
8.2.3.6.3 Product Catalog ....................................................................... 105
8.2.3.6.4 Settings .................................................................................... 108
8.2.3.6.5 Message Bar and Message Window ....................................... 110
8.2.3.6.6 Other Panels ............................................................................ 111
8.2.3.6.7 Status Bar ................................................................................ 111
8.3 Backstage View (Files and Settings) ..................................................... 112
8.3.1 “New” Page ...................................................................................... 113
8.3.2 “Open” Page ..................................................................................... 114
8.3.3 “Save” and “Save As” Page ............................................................. 115
8.3.4 “Print” Page ...................................................................................... 116
8.3.5 “Import/Export” Page ....................................................................... 117
8.3.6 “Project Settings” Page ..................................................................... 119
8.3.7 “Product Catalog” Page .................................................................... 120
8.3.8 “Options” Page ................................................................................. 122
8.3.9 “Help” Page ...................................................................................... 124

9 Operating .................................................................................................. 126
9.1 Working with the User Interface ........................................................... 126
9.1.1 Using Online Help ........................................................................... 126
9.1.1.1 Calling Up Online Help ................................................................ 127
9.1.1.2 Search ........................................................................................ 127
9.1.2 Using Panels ..................................................................................... 128
9.1.2.1 Opening and Closing Panels ....................................................... 128
9.1.2.2 Docking, Floating and Grouping Panels ...................................... 128
9.1.2.3 Docking and Floating the Main View ......................................... 130
9.1.2.4 Using Multiple Selection ............................................................. 130
9.1.3 Using Workspaces ............................................................................ 133
9.1.3.1 Opening the “Network/Devices” Workspace .............................. 133
9.1.3.2 Opening the “Programming” Workspace .................................... 136
9.1.3.3 Using User-Defined Workspaces ................................................ 137
9.1.3.3.1 Creating a Workspace ............................................................ 137
9.1.3.3.2 Editing a Workspace ............................................................... 138
9.1.3.3.3 Deleting a Workspace ............................................................. 138
9.1.3.3.4 Restoring a Workspace ........................................................... 139
WAGO-Software Table of Contents 5

9.1.4  Using Key Combinations ................................................................. 140
  9.1.4.1  General ......................................................................................... 140
  9.1.4.2  “START” Tab .............................................................................. 140
  9.1.4.3  “PROGRAM” Tab ....................................................................... 140
  9.1.4.4  “DEBUG” Tab ............................................................................. 141

9.2  Using Projects ....................................................................................... 142
  9.2.1  Creating a New Project ................................................................. 142
  9.2.2  Opening an Existing Project ............................................................. 142
        9.2.2.1  Opening the Project with an Obsolete Device Description ........ 143
  9.2.3  Saving a Project ............................................................................ 145
        9.2.3.1  Saving Projects Automatically .................................................... 145
        9.2.3.2  Restoring Projects after a Software/Hardware Crash ............... 146

9.3  Configuration and Parameterization ..................................................... 147
  9.3.1  Offline Configuration ....................................................................... 147
        9.3.1.1  Adding Devices in the Project ..................................................... 147
  9.3.2  Online Configuration ........................................................................ 148
        9.3.2.1  Scanning the Network and Devices ............................................. 148
        9.3.2.2  Scanning the Modules of Several Devices .................................. 153
        9.3.2.3  Connecting and Disconnecting Devices ...................................... 154
  9.3.3  Parameterizing Devices .................................................................... 155
        9.3.3.1  Parameterizing Devices via the “Settings” Panel ....................... 156
        9.3.3.2  Using Device-Specific Configurators .......................................... 157
        9.3.3.3  Uploading and Downloading Parameters .................................... 158
  9.3.4  Configuring Networks ...................................................................... 159
        9.3.4.1  Creating a Simple MODBUS Network ....................................... 159
        9.3.4.2  Creating a Simple CANopen Network ........................................ 166
        9.3.4.3  Creating a Simple PROFIBUS Network ..................................... 171

9.4  Testing ................................................................................................... 177
  9.4.1  Performing an Input/Output Test for Digital Input/Output Modules ........................................................................ 177
  9.4.2  Performing a Systematic Input/Output Test (I/O CHECK) ............ 177

9.5  Programming ......................................................................................... 181
  9.5.1  Creating Libraries ............................................................................. 184
  9.5.2  Adding Libraries ............................................................................... 185
  9.5.3  Creating Programs ........................................................................... 187
  9.5.4  Performing I/O Mapping .................................................................. 188
  9.5.5  Compiling an Application ................................................................ 190
  9.5.6  Excluding Programs for Compiling.................................................. 191
  9.5.7  Debugging an Application ................................................................. 192
  9.5.8  Simulating an Application ................................................................ 193
  9.5.9  Creating Visualizations .................................................................... 194
  9.5.10  Connecting ........................................................................................ 197
  9.5.11  Downloading Applications to the Device ......................................... 198
  9.5.12  Importing CODESYS 2 and CODESYS 3 Projects ......................... 199
  9.5.13  Importing/Exporting Program Elements ......................................... 202

9.6  Setting and Managing ........................................................................... 204
  9.6.1  Documenting and Printing a Project ................................................ 204
  9.6.2  Managing Devices ........................................................................... 205
        9.6.2.1  Adding Devices ........................................................................... 205
        9.6.2.2  Deleting Devices ........................................................................ 206
9.6.2.3 Displaying Device Type Versions/Device Descriptions of a Device ................................................................. 207
9.6.2.4 Showing Device Type Versions/Device Descriptions of Several Devices .......................................................... 207
9.6.2.5 Replacing Device Type Versions/Device Descriptions .......... 208
9.6.2.6 Replacing Older Versions Automatically .......................... 210
9.6.3 Performing Updates ............................................................ 211
9.6.4 Archiving ........................................................................... 213
9.6.4.1 Creating an Archive .......................................................... 213
9.6.4.2 Opening an Existing Archive ........................................... 214
9.6.5 Deinstalling........................................................................ 215
9.7 Interfaces .............................................................................. 216
9.7.1 smartDESIGNER ............................................................... 216
9.7.1.1 Importing a Project from smartDESIGNER ...................... 217
9.7.1.2 Exporting a Project for smartDESIGNER ......................... 217

List of Figures .................................................................................. 218

List of Tables ................................................................................. 223
1 Notes about this Documentation

---

Note

Always retain this documentation!
This documentation is part of the product. Therefore, retain the documentation during the entire service life of the product. Pass on the documentation to any subsequent user. In addition, ensure that any supplement to this documentation is included, if necessary.

---

Note

Note about working with this documentation!
Please read the “Overview” section, it is an introduction to the software and provides a general description of functions. The following sections describe program installation and launch. Next, familiarize yourself with the e!COCKPIT graphical user interface. The subsequent sections contain operating instructions for using the software.

1.1 Scope of Validity

This documentation applies to the “e!COCKPIT” software.

1.2 Copyright

This Manual, including all figures and illustrations, is copyright-protected. Any further use of this Manual by third parties that violate pertinent copyright provisions is prohibited. Reproduction, translation, electronic and phototechnical filing/archiving (e.g., photocopying) as well as any amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG, Minden, Germany. Non-observance will involve the right to assert damage claims.
1.3 Symbols

**DANGER**

Personal Injury!
Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

**DANGER**

Personal Injury Caused by Electric Current!
Indicates a high-risk, imminently hazardous situation which, if not avoided, will result in death or serious injury.

**WARNING**

Personal Injury!
Indicates a moderate-risk, potentially hazardous situation which, if not avoided, could result in death or serious injury.

**CAUTION**

Personal Injury!
Indicates a low-risk, potentially hazardous situation which, if not avoided, may result in minor or moderate injury.

**NOTICE**

Damage to Property!
Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

**NOTICE**

Damage to Property Caused by Electrostatic Discharge (ESD)!
Indicates a potentially hazardous situation which, if not avoided, may result in damage to property.

**Note**

Important Note!
Indicates a potential malfunction which, if not avoided, however, will not result in damage to property.
**Information**

**Additional Information:**
Refers to additional information which is not an integral part of this documentation (e.g., the Internet).
### 1.4 Number Notation

Table 1: Number Notation

<table>
<thead>
<tr>
<th>Number Code</th>
<th>Example</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Decimal</td>
<td>100</td>
<td>Normal notation</td>
</tr>
<tr>
<td>Hexadecimal</td>
<td>0x64</td>
<td>C notation</td>
</tr>
<tr>
<td>Binary</td>
<td>'100'</td>
<td>In quotation marks, nibble separated with dots (.)</td>
</tr>
<tr>
<td></td>
<td>'0110.0100'</td>
<td></td>
</tr>
</tbody>
</table>

### 1.5 Font Conventions

Table 2: Font Conventions

<table>
<thead>
<tr>
<th>Font Type</th>
<th>Indicates</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>italic</em></td>
<td>Names of paths and data files are marked in italic-type. e.g.: C:\Program Files\WAGO Software</td>
</tr>
<tr>
<td>Menu</td>
<td>Menu items are marked in bold letters. e.g.: <strong>Save</strong></td>
</tr>
<tr>
<td>&gt;</td>
<td>A greater-than sign between two names means the selection of a menu item from a menu. e.g.: <strong>File &gt; New</strong></td>
</tr>
<tr>
<td><strong>Input</strong></td>
<td>Designation of input or optional fields are marked in bold letters, e.g.: <strong>Start of measurement range</strong></td>
</tr>
<tr>
<td><strong>“Value”</strong></td>
<td>Input or selective values are marked in inverted commas. e.g.: Enter the value “4 mA” under <strong>Start of measurement range</strong>.</td>
</tr>
<tr>
<td><strong>[Button]</strong></td>
<td>Pushbuttons in dialog boxes are marked with bold letters in square brackets. e.g.: <strong>[Input]</strong></td>
</tr>
<tr>
<td><strong>[Key]</strong></td>
<td>Keys are marked with bold letters in square brackets. e.g.: <strong>[F5]</strong></td>
</tr>
</tbody>
</table>
2 Important Notes

This section describes the legal principles and system requirements for using the software in compliance with intended purpose, underlying provisions and stated specifications.

2.1 Legal Bases

2.1.1 Subject to Changes

WAGO Kontakttechnik GmbH & Co. KG reserves the right to provide for any alterations or modifications that serve to increase the efficiency of technical progress. WAGO Kontakttechnik GmbH & Co. KG owns all rights arising from the granting of patents or from the legal protection of utility patents. Third-party products are always mentioned without any reference to patent rights. Thus, the existence of such rights cannot be excluded.

2.1.2 Personnel Qualification

Any steps related to the use of WAGO software may only be performed by qualified employees with sufficient knowledge of handling the respective PC system used.

Steps in which files are created or changed on the PC system may only be performed by qualified employees with sufficient knowledge in the administration of the PC system used in addition to the aforementioned.

Steps in which the behavior of the PC system in a network is changed may only be performed by qualified employees with sufficient knowledge in the administration of the network used in addition to the aforementioned.
2.2 Safety Advice (Precautions)

**Note**
Use up-to-date security software!
Secure operation of the PC system can be at risk as a result of malware such as viruses and Trojans, as well as related threats such as denial-of-service attacks. Therefore, make sure that the latest security software and definitions are always installed on the PC system.

**Note**
Disable or uninstall software that is no longer required!
The vulnerability of a PC system against malware and related threats increases with the number of installed or active software components (applications and services). Therefore, uninstall or disable software components that are not needed for the purpose at hand.

**Note**
Check the runtime system if access problems occur!
To access devices from e!COCKPIT, e!RUNTIME must be set in your device as the runtime system. In the event of access problems, check the device settings with your tool (hardware dependent) or from the Web-based management system.

**Note**
Adjust the screen resolution if necessary!
The CODESYS programming environment in e!COCKPIT is optimized for a screen resolution of 96 DPI. If editors are displayed out of focus, you may be using a different resolution. To adjust the resolution:

1. Right-click on the desktop.
2. Click Customize.
3. Click Display.
4. Select “100%.”
### 3 Overview

**e!COCKPIT** is an integrated development environment with seamless data retention for all automation tasks:

- **Hardware configuration and parameterization**

Hardware component configuration is an essential element of automation applications. All devices are configured so that the controller software produces optimum results during runtime. Controllers, fieldbus couplers/controllers, modules, input and output devices, as well as their communication relations can be configured.

Integrated configurators assist with device and network configuration using standard operating procedures: For example, it is possible to arrange devices in the project via Drag & Drop, use Copy & Paste to quickly multiply individual devices or entire network branches, or set parameter values for multiple devices simultaneously.

Besides the organization of devices in a tree structure, **e!COCKPIT** also enables the graphical display of a network topology – in the Network view – as a display form. This makes it possible to monitor complex interrelationships between devices and their current statuses. The network topology is likewise used for configuring different communication protocols. **e!COCKPIT** simplifies connecting controllers to fieldbuses.

Fieldbus-specific device description files (e.g., EDS), device drivers and libraries allow systems from other suppliers to be integrated into the topology along with WAGO devices. As such, the software has information on device specifications, i.e., device data or supported functions.

- **Programming with integrated e!RUNTIME**

**e!COCKPIT** integrates the **e!RUNTIME** programming software that is based on CODESYS 3. This enables software development in the standard IEC 61131-3 programming languages: Structured text (ST), ladder diagram (LD), function block diagram (FBD), instruction list (IL), sequential function chart (SFC) and “Continuous Function Chart” (CFC). For flexibility, all programming languages can be combined with one another. Created programs can be checked easily via simulations on the development PC. Existing programs can be reused and further developed.

- **Visualization (operation and monitoring)**

**e!COCKPIT** uses Drag & Drop to streamline user interfaces for the operation and visualization of a plant. The integrated visualization editor offers direct access to the program's variables in order to simulate the human machine interface (HMI) and PLC program on the PC. Using Unicode and the latest standards, such as HTML 5 or CSS, also prevents dependency on particular languages and target systems.
• Diagnostics for target-oriented development and commissioning

Whether in the office for development or at the machine for commissioning: Knowing the current, detailed status of the automation network is vital in order to obtain, audit and enable rapid fault localization and debugging. e!COCKPIT offers powerful diagnostic options for this purpose: Error messages are displayed immediately. Through the structured wiring test function, erroneous wiring can be systematically identified.

• Other useful functions

e!COCKPIT comes with an extensive range of IEC libraries. The software also offers several convenience functions, such as automatic updates, context-sensitive menus or user-defined workspaces.

Note

References to the e!RUNTIME programming environment!
Setting options in e!COCKPIT resulting from the integrated e!RUNTIME and CODESYS programming environment are described in the online help feature. Opening the online help in e!COCKPIT will also display the CODESYS documentation in the tree view next to the e!COCKPIT documentation. Individual setting windows or names within the CODESYS documentation may differ from how they are shown in e!COCKPIT.
4 Requirements

4.1 System Requirements

Table 3: System Requirements

<table>
<thead>
<tr>
<th>Components</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows 7/Windows 8</td>
</tr>
<tr>
<td>Memory</td>
<td>Min. 2 GB</td>
</tr>
<tr>
<td>Free hard disk storage</td>
<td>850 MB (x86) or 2 GB (x64) for .Net 4.5 and 3 GB for eCOCKPIT</td>
</tr>
<tr>
<td>Processor</td>
<td>Core2 Duo CPU or newer</td>
</tr>
<tr>
<td>Screen resolution</td>
<td>1,366 x 768 pixels</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td>Network card, standard web browser with Java support, keyboard and mouse or touch screen</td>
</tr>
</tbody>
</table>
5 Licenses

*e!COCKPIT* software is protected by licensing mechanisms.

For productive, unlimited software usage, a license is required, and is available by email or phone from WAGO Kontakttechnik GmbH & Co. KG after purchasing the software.

The software can also be used fully without a license key for 30 days. This trial period only includes the days of actual use. When loading the software, a prompt appears with the number of days remaining. Access without a license key is not possible after the trial period.

Depending on the type of software license, an Internet connection may be required for activation. Please refer to your license certificate for the corresponding information.

Activating a license key is described in section “Starting.”
6 Installation

*e!COCKPIT* software is installed from a setup file. This contains a CODESYS programming environment, the .NET Framework and communication drivers for WAGO 750 Series devices. The device description files (Device Type Packages) for WAGO 750 Series devices are also automatically imported.

1. Open the *e!COCKPIT Website*.

2. To receive a download link for the *e!COCKPIT* software, click the [Registration] link and complete the form.

3. Download the software.

4. Start the installation process by double-clicking the setup file and follow the steps described in the Installation Wizard:
   - Select the installation language.
   - Accept the license agreement.
   - Select the target directory for installation.

5. Click [Install] to start the installation.

If an older version is already installed, a dialog appears in which you can choose to accept already installed devices.

6. Click [Finish] to complete installation.
7 Starting

1. Launch e!COCKPIT via Start > Programs > “WAGO Software” > “e!COCKPIT” or via the link on your desktop.

A start screen will be displayed while the software is loaded.

If you have already purchased and activated a license, the lower area of this screen will display the name under which your e!COCKPIT application is registered.

If you have not yet acquired a license, you can see the remaining time on the start screen.

![Figure 1: Display of the Start Screen during Loading](image)

This also opens a dialog via which you can purchase/activate a license (see the following section).

![Figure 2: Display of the License Status](image)

2. To test e!COCKPIT without entering a license key, click [Continue evaluation].
7.1 Purchasing a License

1. If you wish to purchase a license, click [Purchase license] after launching the software.

You will be taken to a Website where you can place the order (Online catalogue/eShop).

7.2 Activating the License

When purchasing the software, access data is transmitted by email or telephone.

1. To enter a purchased license key and activate the license, click [Enter license].

2. Enter your license key and customer name in the dialog.

   **Note**

   Pay attention to exact spelling! Your access is personalized. Ensure that you enter your customer name exactly as it is written in the email that you received when purchasing the software.

   ![Figure 3: Entering the License Key](image)

3. Accept the license conditions and confirm that you are a commercial user.

4. To activate the license, click [Add licenses].

If the software has already been launched, open this dialog in the Backstage view: “Help” page, [Product Licensing], [Enter License(s)].
Figure 4: Entering Licenses

The license is created in your name and displayed on the “Help” page of the Backstage view.
7.3 Activating Automatic Updates

After the program is launched, the “Automatic updates” dialog is displayed.

![Automatic updates dialog]

Figure 5: Activating Automatic Updates

1. To automatically search for updates when the program launches, click [Yes].

If you choose [No], you can enable the Update function later: In the Backstage view (“FILE” tab) under [Help] > [Updates], tick the “Check for update and license information when starting e!COCKPIT” checkbox (Internet connection required on starting the program).

The update process compares your license information with the records of WAGO Kontakttechnik GmbH & Co. KG.

For additional information on performing updates, see the Section “Operating” > “Setting and Managing” > “Performing Updates.”
8 Graphical User Interface

The graphical user interface consists of three views:

- **Start View**
  For launching the program and using templates.

- **Main View**
  For configuring, parameterization and programming devices.

- **Backstage View**
  For setting general options and file handling settings.

These views are described in the following sections. Start view is shown first when the software is launched.

8.1 Start View (Projects and Templates)

Start view enables you to create new projects or use existing templates. If projects were previously created, the most recently used projects can be accessed quickly from this page.

Figure 6: Start View
Table 4: Key for Figure “Start View”

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>New</td>
</tr>
<tr>
<td></td>
<td>Displays templates that can be used for new projects</td>
</tr>
<tr>
<td>2</td>
<td>Last used</td>
</tr>
<tr>
<td></td>
<td>Displays previous projects with the relevant path</td>
</tr>
<tr>
<td>3</td>
<td>Browse</td>
</tr>
<tr>
<td></td>
<td>Switches to Backstage view, e.g., to open projects from other sources</td>
</tr>
</tbody>
</table>
8.2 Main View (Configuring, Parameterization, Programming)

The main view is divided into a quick access ribbon, menu ribbon, workspace and status bar. The dockable panels enable the workspace to be adjusted to suit the application.

The panels within the main view can be adjusted to create a suitable layout for the relevant task (e.g., creating networks, device configuration or programming). Besides fixed areas offering basic functions, other panels can be arranged as dockable windows depending on the project's needs.

![Main View Diagram](image)

Figure 7: Main View

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Quick access toolbar</td>
</tr>
<tr>
<td>2</td>
<td>Menu ribbon</td>
</tr>
<tr>
<td>3</td>
<td>Workspace</td>
</tr>
<tr>
<td>4</td>
<td>Status Bar</td>
</tr>
</tbody>
</table>

Table 5: Key for Figure “Main View”
8.2.1 Quick Access Toolbar

The quick access toolbar is located in the upper area of the window. The quick access toolbar is a toolbar that is independent of the currently selected tab and is always visible, making it ideal for frequently used commands.

![Quick Access Toolbar](image)

Figure 8: Quick Access Toolbar (Standard View)

It contains the following buttons by default:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Save" /></td>
<td>Save</td>
<td>Saves the project</td>
</tr>
<tr>
<td><img src="image" alt="Undo" /></td>
<td>Undo</td>
<td>Undoes the last action</td>
</tr>
<tr>
<td><img src="image" alt="Repeat" /></td>
<td>Repeat</td>
<td>Repeats the last action</td>
</tr>
<tr>
<td><img src="image" alt="Customize" /></td>
<td>Customize</td>
<td>Opens the menu for customizing the quick access toolbar</td>
</tr>
<tr>
<td><img src="image" alt="Display" /></td>
<td>Display</td>
<td>Displays the quick access toolbar above/below the menu ribbon</td>
</tr>
<tr>
<td><img src="image" alt="Minimize" /></td>
<td>Minimize</td>
<td>Minimizes/maximizes the menu ribbon</td>
</tr>
<tr>
<td><img src="image" alt="Minimize" /></td>
<td>Minimize</td>
<td>Minimizes the window</td>
</tr>
<tr>
<td><img src="image" alt="Maximize" /></td>
<td>Maximize</td>
<td>Maximizes the window</td>
</tr>
<tr>
<td><img src="image" alt="Close" /></td>
<td>Close</td>
<td>Closes e!COCKPIT (Alt + F4)</td>
</tr>
</tbody>
</table>

Table 6: Buttons on the Quick Access Toolbar

![Customize Quick Access Toolbar](image)

Figure 9: Customizing the Quick Access Toolbar

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Save" /></td>
<td>Save</td>
<td>Adds or removes the [Save] button (Save project) to or from the quick access toolbar</td>
</tr>
<tr>
<td><img src="image" alt="Undo" /></td>
<td>Undo</td>
<td>Undoes the last action</td>
</tr>
<tr>
<td><img src="image" alt="Repeat" /></td>
<td>Repeat</td>
<td>Repeats the last action</td>
</tr>
<tr>
<td><img src="image" alt="Additional Functions..." /></td>
<td>Additional Functions...</td>
<td>Currently not available</td>
</tr>
<tr>
<td><img src="image" alt="Display Above/Below Menu Ribbon" /></td>
<td>Display Above/Below Menu Ribbon</td>
<td>Displays the quick access toolbar above/below the menu ribbon</td>
</tr>
<tr>
<td><img src="image" alt="Minimize Menu Ribbon" /></td>
<td>Minimize Menu Ribbon</td>
<td>Minimizes/maximizes the menu ribbon</td>
</tr>
</tbody>
</table>

![Display e!COCKPIT Help](image)

Figure: Displays the e!COCKPIT online help

The CODESYS help function is displayed in the tree structure underneath the e!COCKPIT help
Table 6: Buttons on the Quick Access Toolbar

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>☰</td>
<td>Minimize Menu Ribbon</td>
<td>Displays only the tab names in the menu ribbon. The menu ribbon opens if a tab is clicked. It is then minimized afterward.*</td>
</tr>
<tr>
<td>☰</td>
<td>Extend Menu Ribbon</td>
<td>Displays the menu ribbon permanently*</td>
</tr>
</tbody>
</table>

*Toggling between a Minimized or Extended ribbon is also possible by double-clicking a tab name.

8.2.2 Menu Ribbon

Tabs are arranged under the quick access toolbar. Selecting a tab opens the particular tab ribbon with relevant command buttons based on task and group.

Relevant tabs are displayed according to the workspace. For example, the “PROGRAM” and “DEBUG” tabs are only displayed in the “Programming” workspace.

The buttons on the tabs are context-sensitive, i.e., only the functions that can actually be used at the moment are displayed and activated. Functions that cannot be executed within the relevant context are shown in gray.

The individual tabs are described below.

8.2.2.1 “START” Tab

The “START” tab contains buttons for general project editing and setting the user interface.

Table 7: “START” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>“Clipboard” Group</td>
<td>Insert</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cut</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Copy</td>
</tr>
</tbody>
</table>

Figure 10: Menu Ribbon (Standard View)
## Table 7: “START” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
</table>
|        | **Edit** | Opens one or several project elements for editing  
Double-clicking a project element triggers the same function.  
“Network/Devices” Workspace: Moves to the Device Detail view of the selected device and displays the “Settings” panel to edit device parameters.  
“Programming” Workspace: Depending on the selection in the Program Structure, displays the CODESYS program, project information or library managers |
|        | **Delete** | Deletes one or several project elements “Network/Devices” workspace: Removes the selected devices from the project “Programming” Workspace: Depending on the selection in the Program Structure, deletes the CODESYS program, project information or library managers |
|        | **Invert Selection** | Inverts the devices/elements selection |
|        | **Settings** | Opens the “Settings” panel for the selected object (device, program etc.), see Section “Workspace” > “Panels” > “Settings” |
|        | **Workspaces** Group | |
|        | **Network/Devices** | Displays the preconfigured “Network/Devices” workspace (see Section “Network/Devices workspace”) |
|        | **Programming** | Displays the preconfigured “Programming” workspace (see Section “Programming workspace”) |
|        | **<Workspace>** | Opens the user-defined workspace (see Section “User-Defined Workspaces”) |
|        | **Manage** | Opens a context menu for managing workspaces |

![Figure 11: Managing workspaces](image)
Table 7: “START” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Workspace" /></td>
<td>User-defined workspaces</td>
<td>Selects the relevant workspace that is targeted for deletion or editing</td>
</tr>
<tr>
<td><img src="image" alt="Delete workspace" /></td>
<td>Deletes the selected workspace. The button of the user-defined workspace is removed from the menu ribbon</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Edit workspace" /></td>
<td>Used for customizing the workspace name and icon</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Create new workspace" /></td>
<td>Opens a dialog for creating a new workspace from the current window constellation</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Designation" /></td>
<td>Field for entering the workspace name</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Change image" /></td>
<td>[Change image] opens an Explorer window for selecting a pictogram (JPG, PNG or BMP). [Accept] accepts the name and pictogram of the new workspace as a button in the menu ribbon. [Cancel] closes the dialog without saving the workspace.</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Save the current workspace" /></td>
<td>Saves the changed workspace</td>
<td></td>
</tr>
<tr>
<td><img src="image" alt="Print" /></td>
<td>“Print” Group</td>
<td>Prints the content of the active editor in the “Programming” workspace (available in the “Programming” workspace)</td>
</tr>
<tr>
<td><img src="image" alt="Preview" /></td>
<td>Creates a print preview of the content for the active editor (available in the “Programming” workspace)</td>
<td></td>
</tr>
</tbody>
</table>
**Note**

**Restore standard workspaces!**

If one of the three predefined workspaces has been changed, you can restore it via

[Options] > [Standard workspaces] > [Reset standard workspaces] in the

“FILE” tab.
8.2.2.2 “VIEW” Tab

The “VIEW” tab enables you to display or close workspaces and panels. When a panel is displayed, the relevant buttons are highlighted in blue in the menu ribbon.

![Figure 13: “VIEW” Tab](image)

The “VIEW” tab contains the following operating elements.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Device Structure" /></td>
<td>Devices/Program structure</td>
<td>Opens the “Device Structure” or “Program Structure” depending on which workspace is opened (see Section “Panels” &gt; “Device Structure” or “Program Structure”)</td>
</tr>
<tr>
<td><img src="image" alt="Product Catalog" /></td>
<td>Product Catalog</td>
<td>Opens the “Product Catalog” panel (see Section “Workspaces” &gt; “Product Catalog”)</td>
</tr>
<tr>
<td><img src="image" alt="Messages" /></td>
<td>Messages</td>
<td>Opens the message list (see Section “Panels” &gt; “Message Bar and Message Window”)</td>
</tr>
<tr>
<td><img src="image" alt="Settings" /></td>
<td>Settings</td>
<td>Opens the “Settings” panel (see Section &gt; “Panels” &gt; “Settings”)</td>
</tr>
<tr>
<td><img src="image" alt="Breakpoints" /></td>
<td>Breakpoints</td>
<td>Opens the Breakpoints list (see Section &gt; “Programming” &gt; “Breakpoints” workspace)</td>
</tr>
<tr>
<td><img src="image" alt="Call Stack" /></td>
<td>Call Stack</td>
<td>Opens the call stack (see Section &gt; “Programming” &gt; “Call Stack” workspace)</td>
</tr>
<tr>
<td><img src="image" alt="Monitoring" /></td>
<td>Monitoring</td>
<td>Opens a selection field with the commands “Monitoring 1, 2, 3, 4” or “View all forced values”, which display the corresponding watch list. “View all forced values” is a special view of all currently forced values. It enables “Program Organization Units” (POUs – program organization units of type program, function or function block) and a user-defined list of expressions/variables to be monitored.</td>
</tr>
<tr>
<td><img src="image" alt="Cross reference list" /></td>
<td>Cross reference list</td>
<td>Opens the cross reference list with the references of a variable in the project (see Section “Programming Workspace” &gt; “Cross Reference List”)</td>
</tr>
</tbody>
</table>
Table 8: “VIEW” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>“Workspaces” Group</strong></td>
<td></td>
</tr>
<tr>
<td><img src="image1" alt="Network/Devices" /></td>
<td>Network/Devices</td>
<td>Displays the preconfigured “Network/Devices” workspace (see Section “Network/Devices workspace”)</td>
</tr>
<tr>
<td><img src="image2" alt="Programming" /></td>
<td>Programming</td>
<td>Displays the preconfigured “Programming” workspace (see Section “Programming workspace”)</td>
</tr>
<tr>
<td></td>
<td><strong>“Windows” Group</strong></td>
<td></td>
</tr>
<tr>
<td><img src="image3" alt="Tabbed" /></td>
<td>Tabbed</td>
<td>Arranges the windows of the main view in individual tabs</td>
</tr>
<tr>
<td><img src="image4" alt="Tile horizontally" /></td>
<td>Tile horizontally</td>
<td>Arranges the windows of the main view vertically (tiled horizontally)</td>
</tr>
<tr>
<td><img src="image5" alt="Tile vertically" /></td>
<td>Tile vertically</td>
<td>Arranges the windows of the main view horizontally (tiled vertically)</td>
</tr>
</tbody>
</table>
8.2.2.3  “NETWORK” Tab

The “NETWORK” tab contains functions that specifically relate to the display and scanning of devices in the network. The tab is displayed in the “Network/Devices” workspace.

![NETWORK tab image]

Figure 14: “NETWORK” Tab

The “NETWORK” tab contains the following operating elements:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![View group symbol]</td>
<td>Network</td>
<td>Opens the Network view If the Network view is active, this button is highlighted in blue in the menu ribbon.</td>
</tr>
<tr>
<td>![Display group symbol]</td>
<td>Physical connections</td>
<td>Filters the displayed physical connections arranged according to ETHERNET, CANopen or PROFIBUS</td>
</tr>
<tr>
<td>![Display group symbol]</td>
<td>Logical connections</td>
<td>Filters the displayed logical connections according to PROFINET, MODBUS, EtherCAT, Data Server (CODESYS) or network variables (CODESYS)</td>
</tr>
<tr>
<td>![Grid symbol]</td>
<td>Grid</td>
<td>Shows or hides the grid in the Network view</td>
</tr>
<tr>
<td>![Scan settings group symbol]</td>
<td>Scan</td>
<td>Starts the scan operation via the selected PC interface</td>
</tr>
<tr>
<td>![Scan settings group symbol]</td>
<td>Interface</td>
<td>Selects the interface used for the scan (e.g., COM1, ETHERNET)</td>
</tr>
<tr>
<td>![Scan settings group symbol]</td>
<td>Settings</td>
<td>Opens the Settings dialog for the selected interface</td>
</tr>
</tbody>
</table>
## “Connection” Group

| Connect  | Opens an online connection to one or several selected devices. The CODESYS application is connected with the target system. An ongoing online simulation is closed. The status of the button depends on the devices selected: “Connect” if no device is connected, “Disconnect” if all selected devices are connected. If the selected devices have different connection states, all devices are disconnected on the first click and connected with a second click. Connection requirements:
- error-free program compilation
- correctly configured communication device settings
Connection to the selected device (login) is performed for all applications. |
| Disconnect | Disconnects the development and target systems (controller or simulated device) and returns to offline mode |
8.2.2.4  “DEVICE” Tab

The “DEVICE” tab configures devices. The tab is displayed in the “Network/Devices” workspace.

Figure 15: “DEVICE” Tab

The “DEVICE” tab contains the following operating elements.

Table 10: “DEVICE” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>“View” Group</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device details</td>
<td>Opens the Device Detail view</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If the Device Detail view is active, this button is highlighted in blue in the menu ribbon</td>
</tr>
<tr>
<td></td>
<td><strong>“Connection” Group</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>PC interface</td>
<td>Selects the communication interface to be used for the selected device and displays it</td>
</tr>
<tr>
<td></td>
<td>Addressing</td>
<td>Sets the address for the selected device and the relevant interface, such as the IP address</td>
</tr>
<tr>
<td></td>
<td>Connect</td>
<td>Opens an online connection to one or several selected devices. The CODESYS application is connected with the target system. An ongoing online simulation is closed. The status of the button depends on the devices selected: “Connect” if no device is connected, “Disconnect” if all selected devices are connected. If the selected devices have different connection states, all devices are disconnected on the first click and connected with a second click. Connection requirements: - error-free program compilation - correctly configured communication device settings The connection to the selected device (login) is performed for all applications.</td>
</tr>
<tr>
<td></td>
<td>Disconnect</td>
<td>Disconnects the development system and target systems (controller or simulated device) and returns to offline mode</td>
</tr>
</tbody>
</table>
Table 10: “DEVICE” Tab Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Scan Modules" /></td>
<td><strong>Scan modules</strong></td>
<td>Identifies and displays any modules present</td>
</tr>
<tr>
<td><img src="image" alt="Download" /></td>
<td><strong>Download</strong></td>
<td>Opens a connection to the selected device and downloads parameters to it</td>
</tr>
<tr>
<td><img src="image" alt="Upload" /></td>
<td><strong>Upload</strong></td>
<td>Opens a connection to the device, reads device parameters and displays them in the software</td>
</tr>
<tr>
<td><img src="image" alt="Multiple Download" /></td>
<td><strong>Multiple download</strong></td>
<td>Opens a dialog with a list of all project applications and parameters These can be selected and the selected device(s) can be loaded.</td>
</tr>
</tbody>
</table>

![Figure 16: “Multiple download” Dialog](image)

Project devices, applications and parameters that can be selected via checkboxes are displayed in the left-hand area.

**“All applications”** selects all applications of the project.

**“All parameters”** selects all of the project parameters.

In the right-hand area, the standard CODESYS options are listed that refer to the changes, online change and (renewed) download of the application (see CODESYS Online Help, “Multiple download”).

[OK] loads the selected applications and parameters to the selected devices.

[Cancel] closes the dialog.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Replace" /></td>
<td><strong>Replace</strong></td>
<td>Opens a dialog to replace the device description of the selected device</td>
</tr>
<tr>
<td><img src="image" alt="Replace All" /></td>
<td><strong>Replace all</strong></td>
<td>Opens a dialog for replacing or updating the device descriptions of all devices contained in the project</td>
</tr>
<tr>
<td><img src="image" alt="Version Information" /></td>
<td><strong>Version information</strong></td>
<td>Opens a dialog with version information for the device description of the selected device</td>
</tr>
</tbody>
</table>
The “DEVICE” tab may show additional functions, depending on the selected device. The contents of device-specific tab groups are described in the relevant device documentation.

8.2.2.5 “PROGRAM” Tab

The “PROGRAM” tab contains functions for programming and loading the application onto a device. Actions of this tab always refer to active applications, i.e., applications that are connected or simulated online.

![Figure 17: “PROGRAM” Tab](image)

The “PROGRAM” tab contains the following operating elements:

Table 11: “PROGRAM” Tab Operating Elements

<table>
<thead>
<tr>
<th>Button “Debug” Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Syntax check project library" /></td>
<td>Checks the syntax of the created library. The button is visible if a library template was selected in the Start or Backstage view. The library is compiled in the Backstage view &gt; [Import/Export] &gt; [IEC library].</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>“Connection” Group</th>
<th>Description</th>
</tr>
</thead>
</table>
| ![Connect](image) | Opens an online connection to one or several selected devices. The CODESYS application is connected with the target system. An ongoing online simulation is closed. The status of the button depends on the devices selected: “Connect” if no device is connected, “Disconnect” if all selected devices are connected. If the selected devices have different connection states, all devices are disconnected on the first click and connected with a second click. Connection requirements: | - program compiled error-free 
- correctly configured communication device settings 
Connection to the selected device (login) is performed for all applications. |

| ![Disconnect](image) | Disconnects the development and target systems (controller or simulated device) and returns to offline mode |


Table 11: “PROGRAM” Tab Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>“Online” Group</strong></td>
<td></td>
</tr>
</tbody>
</table>
| ![Simulate application](image) | Can be selected if there is no online connection  
Starts Simulation mode  
The mode starts and debugs an application on a simulated target device in order to test the online behavior of an application in advance without requiring the actual device. No communication settings are required for the simulated device.  
When a function is called for the first time, a prompt asks whether the application is to be created or loaded. |
| ![Close simulation](image) | Closes the simulation mode; the simulation is also closed if the [Connect] button is clicked. |
| ![Start](image)         | Starts the application on the controller or in the simulation                                                                           |
| ![Stop](image)          | Stops the application on the controller or in the simulation                                                                               |
| ![Reset](image)         | All Reset commands cause a restart in online mode.  
The set breakpoints in the project are deactivated.  
**Reset (warm):**  
Resets all variables of the active program to their initial and default values apart from the retentive variables (Retain, Persistent). Behavior is the same as switching the controller off and on while the program is running (e.g., in the event of a power failure).  
**Reset (cold):**  
The same as the “Reset warm” command, but also resets the retentive variables of the active application to their initialization values and default values (behavior is same as the start of the program currently loaded onto the controller).  
**Reset (original):**  
Resets the values of all variables of the active application to their initialization values and deletes the program on the controller. |
| **“Functions” Group**   |                                                                                                                                              |
| ![Program download](image) | Builds and compiles the active application in online mode, i.e., a syntax check is executed, application code is generated and then downloaded to the controller  
The compilation log is created in the project directory.  
All variables are reinitialized apart from the persistent variables. |
<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Multiple download" /></td>
<td>Opens a dialog with a list of all project applications and parameters. These can be selected and the selected device(s) can be loaded.</td>
</tr>
<tr>
<td><img src="image" alt="Boot application download" /></td>
<td>Saves the boot application under the name <code>&lt;Application_name&gt;.app</code> on the target device. This automatically begins when the controller is started. If the controller is in offline mode, the current application is first stored as a boot application in the project directory in order to download it later to the target system.</td>
</tr>
<tr>
<td><img src="image" alt="Source code download" /></td>
<td>Downloads the current project as an “archive.prj” archive file to the currently connected controller.</td>
</tr>
<tr>
<td><img src="image" alt="Edit offline" /></td>
<td>Enables the editing of a currently connected application. The changes can be downloaded later to the controller via “Download Changes.”</td>
</tr>
<tr>
<td><img src="image" alt="Changes download" /></td>
<td>Compiles changes to the active application carried out offline and downloads them to the controller.</td>
</tr>
</tbody>
</table>

**“Source code” Group**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Build" /></td>
<td>Checks the source code of the active application for errors</td>
</tr>
</tbody>
</table>
Table 11: “PROGRAM” Tab Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Button Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rebuild</td>
<td>Checks the source code again</td>
</tr>
<tr>
<td></td>
<td>Generate Code</td>
<td>Compiles the source code of the active application. The code is not loaded onto the PLC.</td>
</tr>
<tr>
<td></td>
<td>Clean</td>
<td>This cleans the compiler information for the currently active application on the development system. The compiler information was created the last time the application was downloaded to the target system and saved in a “*.compileinfo” file on the development system.</td>
</tr>
</tbody>
</table>

**Note**

For online change after cleaning, reload the application!

Reload the program to the controller after cleaning if you wish to use Online Change, otherwise it is not possible to execute Online Change.

<table>
<thead>
<tr>
<th>Button</th>
<th>Button Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Clean All</td>
<td>Deletes the compilation information for all applications.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>This was created the last time the applications were downloaded to the target systems and saved in a “*.compileinfo” file. After cleaning, an online change is no longer possible for the affected application. In this case, the program must be completely re-loaded onto the controllers. Internal information on persistent variables is also deleted, so this must be re-installed after a download.</td>
</tr>
</tbody>
</table>

“Search” Group

<table>
<thead>
<tr>
<th>Button</th>
<th>Button Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Search</td>
<td>Opens the “Search” dialog</td>
</tr>
<tr>
<td></td>
<td></td>
<td><img src="image" alt="Figure 19: “Search” Dialog" /></td>
</tr>
<tr>
<td></td>
<td>Find what:</td>
<td>Searches for the expression entered or selected from the drop-down list</td>
</tr>
<tr>
<td></td>
<td>Match case</td>
<td>Distinguishes between upper/lower case</td>
</tr>
<tr>
<td></td>
<td>Search up</td>
<td>When selected, searches for previous occurrences of the expression</td>
</tr>
<tr>
<td></td>
<td>Match whole word</td>
<td>When selected, searches for an exact term without additional characters</td>
</tr>
</tbody>
</table>


Table 11: “PROGRAM” Tab Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use regular expressions</strong></td>
<td>When selected, allows the use of syntax rules for evaluating the expression, e.g., characters “*,” “+” or “?”</td>
</tr>
<tr>
<td><strong>[Find Next]</strong></td>
<td>Searches for the next expression</td>
</tr>
<tr>
<td><strong>[Find All]</strong></td>
<td>Shows all occurrences of the expression in the message window Double-clicking on this entry enables you to move in the editor to the appropriate position</td>
</tr>
</tbody>
</table>
| **[Replace >>]**   | Opens a “Replace” dialog This dialog contains the following additional options:  
**[Replace]** replaces the next instance of the expression with the expression entered in the “Replace with” field. **[Replace All]** replaces all occurrences of the expression searched for. |
| **Search:**        | Selects the location to be searched:  
- Active editor  
- All opened editors  
- Entire project  
 […] opens a dialog with the additional option to search in objects and store this search as a search scheme. |
| **[Close]**        | Closes the “Search” dialog.                                                                                                                  |
|                     | Opens the quick access list for the search.                                                                                                  |
| **Search**          | Opens the “Search” dialog (Ctrl + F)                                                                                                           |
| **Replace**         | Opens the “Replace” dialog (Ctrl + H)                                                                                                          |
| **Next**            | Shows the next occurrence of the search expression (F3)                                                                                       |
| **Next (selection)** | Displays the next occurrence of the search expression (Ctrl + F3) selected in the editor                                                      |
| **Search previous** | Displays the previous occurrence of the search expression (Shift + F3)                                                                            |
| **Search previous (selection)** | Displays the previous occurrence of the search expression (Ctrl + Shift + F3) selected in the editor                                        |
| **Cross reference list** | Opens the cross reference list with the references of a variable in the project (see Section “Programming Workspace” > “Cross Reference List”) |
Table 11: “PROGRAM” Tab Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Mark declaration" /></td>
<td>“Source text navigation”; jumps to the relevant definition/declaration of the identifier that the cursor is positioned on. Function block, program or variable is opened in the appropriate editor. If the definition is located in a “compiled” library, the corresponding function block is opened in the Library Manager.</td>
</tr>
<tr>
<td><img src="image" alt="Import" /></td>
<td>Imports program elements of an export file</td>
</tr>
<tr>
<td><img src="image" alt="Export" /></td>
<td>Save program elements in an export</td>
</tr>
</tbody>
</table>
8.2.2.6 "DEBUG" Tab

The “DEBUG” tab contains functions for executing and checking the source code.

Figure 20: “DEBUG” Tab

The “DEBUG” tab contains the following operating elements:

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection Group</td>
<td></td>
</tr>
<tr>
<td>![Connect Icon]</td>
<td>Connect</td>
</tr>
<tr>
<td>![Disconnect Icon]</td>
<td>Disconnect</td>
</tr>
<tr>
<td>Execute Group</td>
<td></td>
</tr>
<tr>
<td>![Simulate Application Icon]</td>
<td>Simulate application</td>
</tr>
</tbody>
</table>
### Table 12: “DEBUG” Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Close simulation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Closes the simulation mode; the simulation is also closed if the [Connect] button is clicked.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Start</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Starts the application on the controller or in the simulation</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Stop</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Stops the application on the controller or in the simulation</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Single cycle</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Performs one cycle of the active application</td>
<td></td>
</tr>
</tbody>
</table>

**“Breakpoints” Group**

<table>
<thead>
<tr>
<th>Button</th>
<th>New</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Sets a breakpoint at the current position within the active code editor (see Section “Programming Workspace” &gt; “Breakpoints”)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Toggle</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Toggles between the active and inactive state of a breakpoint. This also causes the setting of a new breakpoint (if not yet set at the current breakpoint position). Active breakpoints are deactivated after online mode is closed or after logging in again (see Section “Programming Workspace &gt; “Breakpoints”).</td>
<td></td>
</tr>
</tbody>
</table>

**“Increment” Group**

<table>
<thead>
<tr>
<th>Button</th>
<th>Step Over</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Executes a program in online mode in defined steps, e.g., for debugging. With instructions on one level, this corresponds to “stepping” with the “Step Into” command (see next command). When a function block call is reached, the entire function block is performed in a single step. A complete action is processed in a sequential function chart.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Button</th>
<th>Single Into</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Button Icon]</td>
<td>Executes a program in online mode in defined steps, e.g., for debugging. A single step is performed. If necessary, the program switches to another program organization unit (POU). If the current position is a function call or function block call, the program stops in front of the first instruction of the called function block. In all other situations, the command has the same function as the “Step Over” command. Possible stop positions during step-by-step processing depend on the type of editor. The current position is highlighted in yellow.</td>
<td></td>
</tr>
</tbody>
</table>
### Table 12: “DEBUG” Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Button Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Step Out" /></td>
<td><strong>Step Out</strong></td>
<td>Steps a program backward. In the case of a nested program code, the backward step is made through the calls. This causes a return to the start of the program if the program does not contain any calls. It also causes a return to the calling instruction if the program had jumped previously to a called function block.</td>
</tr>
<tr>
<td><img src="image" alt="Run to Cursor" /></td>
<td><strong>Run to Cursor</strong></td>
<td>Runs the program from the current position to a temporarily definable position. For this, the cursor is placed at the required stop position before performing the command.</td>
</tr>
<tr>
<td><img src="image" alt="Set Next Statement" /></td>
<td><strong>Set Next Statement</strong></td>
<td>Sets the statement to be executed next; to do this, the cursor is placed in this instruction before executing the command.</td>
</tr>
<tr>
<td><img src="image" alt="Show Current Statement" /></td>
<td><strong>Show Current Statement</strong></td>
<td>This returns the cursor in online mode to the current execution position if the stepwise processing had moved the cursor to another position in the user interface.</td>
</tr>
<tr>
<td><img src="image" alt="“Values” Group" /></td>
<td><strong>Monitoring</strong></td>
<td>Opens a selection field with the commands “Monitoring 1, 2, 3, 4” or “View all forced values”, which display the relevant watch list. “View all forced values” is a special view of all currently forced values. It enables program organization units (POUs – program organization units of type program, function or function block) and a user-defined list of expressions/variables to be monitored.</td>
</tr>
<tr>
<td><img src="image" alt="Write" /></td>
<td><strong>Write</strong></td>
<td>Sets the corresponding variable in the controller at the start of the next cycle once to the value defined for it in the programming software. The command is applied to all variables of the active application that are prepared for writing.</td>
</tr>
<tr>
<td><img src="image" alt="Force" /></td>
<td><strong>Force</strong></td>
<td>If the controller is connected (online), the Force Values command causes one or several expressions/variables of the active application to be set permanently to defined values, i.e., they are not overwritten while the program is running. This command is available in online mode. [F7]</td>
</tr>
<tr>
<td><img src="image" alt="Unforced values" /></td>
<td><strong>Unforced values</strong></td>
<td>Removes the forcing of all variables of the active application in online mode. Forcing is also removed by disconnecting the controller from the programming system. The variables are then assigned their current value from the controller.</td>
</tr>
</tbody>
</table>
### Table 12: “DEBUG” Operating Elements

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Display Mode" /></td>
<td><strong>Display Mode</strong>&lt;br&gt; Sets the number format of variable values. The following can be selected: decimal, hexadecimal, binary.</td>
</tr>
<tr>
<td><img src="image" alt="Flow Control" /></td>
<td><strong>Flow Control</strong>&lt;br&gt; Switches the flow control in online mode on or off for the language editors ST, FBD, LD, IL and CFC; an activated flow control enables program processing to be followed in the editor window. Current variables, and thus the results of function calls and operations in the current cycle, are highlighted at the relevant processing position and time. The flow control functions in all currently visible sections of the currently opened editor window. Values cannot be written or forced in the flow control. This can only be performed in the declaration section.</td>
</tr>
</tbody>
</table>

### Note

*Function increases run time!*
Activating the flow control increases the application runtime.

### WARNING

*Be aware of dangers resulting from writing or forcing values!*
Abnormally changing a value in an application running on the controller may cause undesired behavior of the controlled system. Evaluate any possible dangers resulting from writing or forcing variable values and take appropriate safety precautions. Depending on the system controlled, there is a risk of damage to the system and equipment or danger to people.*
### 8.2.2.7 “I/O CHECK” Tab

This tab enables a systematic input/output test to be performed. It verifies that a controller is wired correctly. Both individual modules and entire nodes can be tested. The “I/O CHECK” tab is displayed if a device is selected or the Device Detail view of a device is opened. If several devices are selected, the “I/O CHECK” tab is hidden.

![Image 1](image1.png)

**Figure 21: “I/O CHECK” Tab**

The “I/O CHECK” tab contains the following operating elements: Active functions are highlighted in blue.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image2.png" alt="Image 2" /></td>
<td>Build</td>
<td>Tests the connection (systematic hardware function test)</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image 3" /></td>
<td>Result</td>
<td>Switches the view of the “Status” and “Test Result” columns on/off</td>
</tr>
<tr>
<td><img src="image4.png" alt="Image 4" /></td>
<td>Connect</td>
<td>Opens an online connection to one or several selected devices. The CODESYS application is connected with the target system. An ongoing online simulation is closed. The status of the button depends on the devices selected: “Connect” if no device is connected, “Disconnect” if all selected devices are connected. If the selected devices have different connection states, all devices are disconnected on the first click and connected with a second click. Connection requirements: - error-free program compilation - correctly configured communication device settings Connection to the selected device (login) is performed for all applications.</td>
</tr>
<tr>
<td><img src="image5.png" alt="Image 5" /></td>
<td>Disconnect</td>
<td>Disconnects the development and target systems (controller or simulated device) and returns to offline mode</td>
</tr>
</tbody>
</table>
## “Control mode” Group

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Control mode" /></td>
<td>Activates/deactivates the Control mode: Enables the modules for changes (write). If the control mode is not active and there is an online connection to the device, the monitor mode is automatically activated, i.e., actual values of the I/O module are displayed (read only).</td>
</tr>
<tr>
<td><img src="image" alt="Direct" /></td>
<td>Activates/deactivates the mode for direct writing to the I/O module</td>
</tr>
<tr>
<td><img src="image" alt="Collect (Number of prepared changes)" /></td>
<td>Writes prepared changes (marked with an asterisk) to the module</td>
</tr>
<tr>
<td><img src="image" alt="Buttons" /></td>
<td>Activates/deactivates the push-button mode for digital inputs/outputs</td>
</tr>
</tbody>
</table>

## “Collection values” Group

This tab group is displayed if the “Collect” mode is active (blue background).

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Write Now" /></td>
<td>Write collected values to the module. Changed values are indicated with an asterisk.</td>
</tr>
<tr>
<td><img src="image" alt="Reset" /></td>
<td>Reset collected values. Changed values are indicated with an asterisk.</td>
</tr>
</tbody>
</table>

Besides this systematic input/output test, a simple test can also be performed for digital input/digital output modules via the Device Detail view (see Section “Workspace” > … > “Device Detail view”).
8.2.2.8 Other Tabs

Depending on the devices used, other tabs may be displayed that have specific content relating to the particular device driver. The setting options can be found in the relevant product documentation of the device.
8.2.3 Workspace

The workspace consists of the window area underneath the menu ribbon. This is divided into the main workspace and panels that can be arranged freely; these elements can be shown or hidden via the “VIEW” tab to suit the particular application or specific operation. Panels are used to display and/or edit project data or other data.

*e!COCKPIT* offers two predefined workspaces that are already adapted for typical applications such as network and device configuration, parameterization and programming. The appearance of the user interface changes according to the workspace selected.

User-defined workspaces can be created in addition to the predefined workspaces. The position and size of selected views, tabs and panels are saved here to customize the workspace. These settings are stored separately for each Windows user.

![Figure 22: Views and Workspaces](image)

After opening a newly created project in *e!COCKPIT*, the standard view shows the “Device Structure” panel on the left, the Network view in the middle and “Product Catalog” and “Settings” panels on the right.

The configuration, parameterization and programming are performed in the central window area.
8.2.3.1 General Operating Elements

The main workspace and panels have the following operating elements (some of these functions depend on the panel):

Table 14: General Operating Elements

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Search Icon]</td>
<td>Search</td>
<td>Shows or hides the search field. (free text search, no wildcards necessary) The search field is permanently displayed in some panels.</td>
</tr>
<tr>
<td>![Dock Icon]</td>
<td>Dock</td>
<td>Hides the panel; Panel appears as a tab on the right of the window or at bottom left.</td>
</tr>
<tr>
<td>![Close Icon]</td>
<td>Close</td>
<td>Closes the panel</td>
</tr>
<tr>
<td>![Zoom Icon]</td>
<td>Zoom</td>
<td>Increases/decreases magnification of the view (Zoom) (Network/Device Detail view)</td>
</tr>
<tr>
<td>![Adjust Icon]</td>
<td>Adjust</td>
<td>Adjusts the view to the optimum size or to the window (Network/Device Detail view)</td>
</tr>
<tr>
<td>![Expand Icon]</td>
<td>Expand</td>
<td>Opens all elements of the tree structure</td>
</tr>
<tr>
<td>![Collapse Icon]</td>
<td>Collapse</td>
<td>Closes all elements of the tree structure</td>
</tr>
</tbody>
</table>

8.2.3.2 General Status Display for Connections

A device's connection status is shown as a four-section status display in operating fields and workspaces.

![General Status Display for Connections Diagram]

Figure 23: General Status Display for Connections

Table 15: Key for Figure “General Status Display for Connections”

<table>
<thead>
<tr>
<th>Item</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Connection status</td>
</tr>
<tr>
<td>2</td>
<td>PLC status</td>
</tr>
<tr>
<td>3</td>
<td>Status of the internal bus (e.g., internal data bus, 750 Series)</td>
</tr>
<tr>
<td>4</td>
<td>Fieldbus status</td>
</tr>
</tbody>
</table>
The status information is likewise displayed in the status bar. The following table provides an overview of the different symbols:

### Table 16: General Status Display for Connections

<table>
<thead>
<tr>
<th>Symbol in Status Bar</th>
<th>Symbol in Workspace</th>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td><strong>Connection</strong></td>
<td></td>
</tr>
<tr>
<td>![not_connected]</td>
<td>![not_connected]</td>
<td>Not connected</td>
<td>The selected device is not connected.</td>
</tr>
<tr>
<td>![connected]</td>
<td>![connected]</td>
<td>Connected</td>
<td>The selected device is connected.</td>
</tr>
<tr>
<td>![limited_connection]</td>
<td>![limited_connection]</td>
<td>Limited Connection</td>
<td>The selected device has a limited connection. This means that the application on the controller is different than the application in the project within e!COCKPIT.</td>
</tr>
<tr>
<td>![control_mode_active]</td>
<td>![control_mode_active]</td>
<td>Control Mode Active</td>
<td>The Control mode is active. Write access to modules is possible. No online connection to the device.</td>
</tr>
<tr>
<td>![simulation_mode_active]</td>
<td>![simulation_mode_active]</td>
<td>Simulation Mode Active</td>
<td>The Simulation mode is active.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>PLC</strong></td>
<td></td>
</tr>
<tr>
<td>![stop]</td>
<td>![stop]</td>
<td>Stop</td>
<td>The controller is not running (PLC stopped).</td>
</tr>
<tr>
<td>![running]</td>
<td>![running]</td>
<td>Running</td>
<td>The controller is running.</td>
</tr>
<tr>
<td>![error]</td>
<td>![error]</td>
<td>Error</td>
<td>Error is present. Note the error messages in the message window.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Internal Bus</strong></td>
<td></td>
</tr>
<tr>
<td>![not_running]</td>
<td>![not_running]</td>
<td>Not running</td>
<td>No communication via the internal bus.</td>
</tr>
<tr>
<td>![running]</td>
<td>![running]</td>
<td>Running</td>
<td>Internal bus ready for communication.</td>
</tr>
<tr>
<td>![error]</td>
<td>![error]</td>
<td>Error</td>
<td>Error is present. Note the error messages in the message window.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Fieldbus</strong></td>
<td></td>
</tr>
<tr>
<td>![not_running]</td>
<td>![not_running]</td>
<td>Not running</td>
<td>No communication via fieldbus.</td>
</tr>
<tr>
<td>![running]</td>
<td>![running]</td>
<td>Running</td>
<td>Fieldbus ready for communication.</td>
</tr>
<tr>
<td>![error]</td>
<td>![error]</td>
<td>Error</td>
<td>Error is present. Note the error messages in the message window.</td>
</tr>
</tbody>
</table>

If a device is connected online, connection status is also displayed in a pop-up display as soon as the mouse is placed over the device. The status information is displayed both for fieldbus couplers/controllers as well as for connected modules.
Figure 24: Status Display for Devices (Online)
8.2.3.3  “Network/Devices” Workspace

The “Network/Devices” workspace is used to configure and parameterize the network and devices. The workspace contains the following operating elements:

![Figure 25: “Network/Devices” Workspace](image)

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tab</td>
</tr>
<tr>
<td>2</td>
<td>Breadcrumb Navigation</td>
</tr>
<tr>
<td>3</td>
<td>Graphic/Table Toggle Button</td>
</tr>
<tr>
<td>4</td>
<td>Zoom Function</td>
</tr>
<tr>
<td>5</td>
<td>Search field</td>
</tr>
<tr>
<td>6</td>
<td>Active Tabs</td>
</tr>
</tbody>
</table>
### 8.2.3.3.1 Network View

Network view displays configured network connections between devices. It contains the following elements:

**Figure 26: Network View**

**Table 18: Key for Figure “Network View”**

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tile</td>
</tr>
<tr>
<td></td>
<td>Displays the included device.</td>
</tr>
<tr>
<td></td>
<td>To change a name, click the displayed device name to enter a new name.</td>
</tr>
<tr>
<td>2</td>
<td>Device Location</td>
</tr>
<tr>
<td></td>
<td>Empty tile as location for devices.</td>
</tr>
<tr>
<td>3</td>
<td>Network/Bus Connections</td>
</tr>
<tr>
<td></td>
<td>Physical/logical connections between devices are displayed as horizontal/vertical lines.</td>
</tr>
<tr>
<td></td>
<td>Colors are uniquely assigned to a particular network/bus type.</td>
</tr>
<tr>
<td>4</td>
<td>Number of Modules</td>
</tr>
<tr>
<td></td>
<td>This shows the number of modules connected with this head station and which of these are included in the Device Detail view.</td>
</tr>
<tr>
<td>5</td>
<td>Connectors</td>
</tr>
<tr>
<td></td>
<td>The interfaces via which connections to other devices can be configured.</td>
</tr>
</tbody>
</table>
8.2.3.3.1.1 Tiles

The Network view’s workspace is divided into tiles. Tiles are placeholders for devices that are added from the Product Catalog. The device's status is indicated by its frame:

Table 19: Device Status

<table>
<thead>
<tr>
<th>Frame Color</th>
<th>Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray</td>
<td>Standard frame of an included device.</td>
<td></td>
</tr>
<tr>
<td>Light Blue</td>
<td>Device has been selected/highlighted.</td>
<td></td>
</tr>
<tr>
<td>Green</td>
<td>The device is connected online.</td>
<td></td>
</tr>
<tr>
<td>Light green</td>
<td>The device has a limited connection (controller application is different to the project application in e!COCKPIT).</td>
<td></td>
</tr>
<tr>
<td>Orange</td>
<td>Control mode (“I/O-CHECK” tab) is active.</td>
<td></td>
</tr>
<tr>
<td>Blue</td>
<td>Simulation mode is active (“PROGRAM”/“DEBUG” tab).</td>
<td></td>
</tr>
</tbody>
</table>

The device’s status — with regard to the connection, PLC, internal bus and fieldbus — is indicated by a four-section status display in the top right corner of the tile. The individual symbols are explained in the Section “Workspace” > “General Status Display for Connections”.

Right-clicking the tile displays the context menu’s different buttons:

![Figure 27: Context Menu of a Device](image-url)
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Copy]</td>
<td>Copies selected devices</td>
</tr>
<tr>
<td>![Insert]</td>
<td>Inserts copied devices</td>
</tr>
<tr>
<td>![Delete]</td>
<td>Deletes the selected device from the Network view. When deleting the device, the associated program elements are also deleted.</td>
</tr>
<tr>
<td>![Connect]</td>
<td>Connects the development and target systems (controller or simulated device) and returns to offline mode.</td>
</tr>
<tr>
<td>![Disconnect]</td>
<td>Disconnects the connection between the development and target systems (controller or simulated device) and returns to offline mode.</td>
</tr>
<tr>
<td>![Start]</td>
<td>Starts the application on the controller or in the simulation</td>
</tr>
<tr>
<td>![Stop]</td>
<td>Stops the application on the controller or in the simulation</td>
</tr>
<tr>
<td>![Scan]</td>
<td>Scans the node for the presence of modules and displays results in a list</td>
</tr>
<tr>
<td>![Upload]</td>
<td>Reads parameters from the selected device</td>
</tr>
<tr>
<td>![Download]</td>
<td>Writes parameters to the selected device</td>
</tr>
</tbody>
</table>
| ![I/O-Check] | Opens the “I/O CHECK” tab (see Section “I/O CHECK Tab”)
| ![Device details] | Opens the Device Detail view and displays the I/O mapping |
| ![Settings] | Displays device settings in the corresponding panel (see Section “Panels” > “Settings”) |
8.2.3.1.2 Connectors

Each device supports specific interfaces for communicating with other devices. The interfaces are shown as connection points (connectors) in different colors on the edge of a tile:

![Figure 28: Color Coding of Connectors](image)

Table 21: Color Coding of Connectors

<table>
<thead>
<tr>
<th>Connector Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray</td>
<td>Connector for ETHERNET/MODBUS connections</td>
</tr>
<tr>
<td>Black</td>
<td>Connector for serial connections/COM interfaces</td>
</tr>
<tr>
<td>Blue</td>
<td>Connector for CANopen connections</td>
</tr>
<tr>
<td>Purple</td>
<td>Connector for PROFIBUS connections</td>
</tr>
</tbody>
</table>

Hovering the mouse over a connector displays additional information, such as interface type, the device's IP address or connections to other devices, if present.

![Figure 29: Context Menu of a Connector](image)
Table 22: Operating Elements in the Context Menu of a Connector

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Symbol" /></td>
<td>Pictogram for the type of logical connection</td>
</tr>
<tr>
<td><img src="image2" alt="Symbol" /></td>
<td>Selects the connected device and centers the view</td>
</tr>
</tbody>
</table>
| ![Symbol](image3) | Updates the configuration  
The button is available when the configuration of connected devices has been changed.  
Example: The configuration of a master device is adapted to the configuration of the connected slave. This is indicated by a warning symbol next to the interface: ![Warning Symbol] |
| ![Symbol](image4) | Opens the connection settings (see Section “Fieldbus-Specific Configurators”). |
| ![Symbol](image5) | Removes the logical connection |
| ![Symbol](image6) | Displays device settings in the corresponding panel (see Section “Panels” > “Settings”) |
| ![Symbol](image7) | Deletes all connections for this interface to any number of devices |
8.2.3.3.1.3 Network/Bus Connections

The devices can be interconnected in Network view. Connectors are dragged with the mouse between two similar device connectors. The color of a connection is the same as the relevant connector color (see previous Section).

![Figure 30: Connection between Two Devices](image)

Connections are either physical or logical connections.

**Physical connections** correspond to cables between devices that are compatible with their hardware, i.e., their terminations. The direction in which connection lines are dragged with the mouse is not important to physical connections.

**Logical connections** are only possible where physically connected devices can exchange information with the appropriate protocols. Logical connections are dragged over an already displayed physical connection.

Unlike physical connections, the direction in which devices are connected is relevant for logical connections as the displayed options (e.g., “Modbus TCP master”) refer to the source device.

Example: When selecting “Modbus TCP Master” the source device becomes the master, the target device becomes the slave for the new connection.

With connections via CANopen and PROFIBUS, the physical connection is the same as the logical connection. Only one connection line must be dragged between the devices. Whether it is a master or slave device depends solely on the configuration of the device concerned.

To control data exchange, each logical connection can be further configured via the buttons in the context menu of the connector (see previous Section).
8.2.3.3.2 Fieldbus-Specific Configurators

Fieldbus-specific configurators are used to configure MODBUS, CANopen and PROFIBUS connections. These are accessed via the context menu of a connector and displayed in the main workspace.

The connected devices, as well as the device type, are displayed in the upper area.

![Fieldbus-Specific Configurators](image)

Figure 31: Fieldbus-Specific Configurators

The lower area is structured according to the specific fieldbus, see the following Section.
8.2.3.3.2.1 MODBUS Configurator

*e!COCKPIT* integrates two MODBUS configurators for configuring the connection between MODBUS master and MODBUS slave.

The data points/variables to be exchanged in a real master-slave connection is set in the master configurator.

The slave configurator sets which data points/variables are available for all masters.

**MODBUS Master Configurator**

The settings are made in two tabs.

The “MODBUS Master-Variables” tab configures data exchange between the master and slave. The left-hand column here displays the variables used by the slave. The right-hand column displays the variables available on the master.

![Figure 32: “MODBUS Master Variables” Tab](image)

Figure 32: “MODBUS Master Variables” Tab
### Table 23: Key for Figure “MODBUS Master Variables” Tab

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variables (Slave, left-hand column)</strong></td>
<td></td>
</tr>
</tbody>
</table>
| Variable | Variable name and path on the slave  
Variants are normally either manual or automatic.  
Automatic: A variable is created according to the variable path on the master.  
Manual: The variable is entered by the user. This can be an existing variable or one created by CODESYS. |
| Mapping | Existing (arrow symbol) variables and variables generated by CODESYS (asterisk) |
| Variable (slave) | Variable name and path on the master |
| **T [ms]** | Cycle time when accessing the variables |
| **Access** | Read and write access |
| **Trigger** | Triggering of access. Selection depends on the access setting.  
Read: Cyclic  
Write: Cyclic/on value change |
| **Variables (Master, right-hand column)** | |
| Variable | Name of variable |
| **Data type** | Variable data type |
| **Comment** | Comment of the variable |
| **Variable settings** | |
| **Designation** | Name of variable |
| **Data type** | Variable data type |
| **Array** | Change data type to array (to read or write access several MODBUS tabs at once).  
The size of the variables is set via the number of elements in the array. |
| **Access** | Access via MODBUS (RO/RW) |
| MODBUS address | MODBUS address of the variable |

**Context menu of the slave variables (left-hand column):**

![Context menu](image)

**Figure 33: Slave Variables Context Menu**
Table 24: Key to Figure “Slave Variables Context Menu”

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delete variable</td>
<td>Delete variable of slave</td>
</tr>
<tr>
<td></td>
<td>Reset</td>
<td>Reset variable to “automatic value”</td>
</tr>
<tr>
<td></td>
<td>Transfer</td>
<td>Accept value of the cell for all selected rows (T [ms], trigger, access)</td>
</tr>
</tbody>
</table>

Context menu of the master variables (right-hand column):

![Application, PLC_PRO]  ![Create, Publish, Delete]  ![WAGO MODBUS special register]  ![WatchdogCommand]  ![WORD]  

Figure 34: Master Variables Context Menu

Table 25: Key to Figure “Master Variables Context Menu”

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Publish</td>
<td>Publish variable via MODBUS and make it available in the slave</td>
</tr>
<tr>
<td></td>
<td>Delete</td>
<td>Delete variable of master</td>
</tr>
<tr>
<td></td>
<td>Create</td>
<td>Create new variable on the master</td>
</tr>
</tbody>
</table>

In the “Connection Settings” tab, all general settings for the MODBUS connection are performed, such as the response timeout of requests and request intervals. Settings for “TCP Keepalive” specifically apply to MODBUS/TCP connections; “Type Of Service (IP)” applies to MODBUS-UDP and TCP connections.

![Communication settings]  ![Response timeout (ms)]  ![Request interval (ms)]  ![Keepalive (TCP)]  ![KeepAliveTime [s]]  ![KeepAliveInterval [s]]  ![KeepAliveProbes [Number]]  ![Type of Service (IP)]  ![Low Delay]  ![High Throughput]  ![High Reliability]  

Figure 35: “Connection Settings” Tab
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Communication Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Response timeout [ms]</td>
<td>Wait time in ms until a request is repeated</td>
</tr>
<tr>
<td>Request interval [ms]</td>
<td>Time in ms between two successive requests</td>
</tr>
<tr>
<td><strong>Keepalive (TCP)</strong></td>
<td></td>
</tr>
<tr>
<td>KeepAliveTime [s]</td>
<td>The time span in seconds (once) between the last data packet and the first KeepAlive packet (probe)</td>
</tr>
<tr>
<td>KeepAliveInterval [s]</td>
<td>Interval in seconds between two successive Keepalive packets (Probes)</td>
</tr>
<tr>
<td>KeepAliveProbes [Number]</td>
<td>Number of Keepalive packets until the connection is considered lost/removed</td>
</tr>
<tr>
<td><strong>Type of Service (IP)</strong></td>
<td></td>
</tr>
<tr>
<td>Low Delay</td>
<td>Sets the “Low Delay” attribute in the IP header</td>
</tr>
<tr>
<td>High Throughput</td>
<td>Sets the “High Throughput” attribute in the IP header</td>
</tr>
<tr>
<td>High Reliability</td>
<td>Sets the “High Reliability” attribute in the IP header</td>
</tr>
</tbody>
</table>
MODBUS Slave Configurator

The MODBUS Slave Configurator is used to configure the MODBUS-accessible variables.

The left-hand column contains a tabular overview of all variables published via MODBUS. The variables present in the slave application are mapped in the right-hand column.

Figure 36: MODBUS Slave Configurator

The function of the right-hand column is identical to the function in the MODBUS master configurator. A variable can be deleted in the left-hand column, otherwise the column is only for informational purposes.
8.2.3.3.2.2 CANopen Configurator

“CANopen Remote Device” Tab

This tab sets the parameters in order to configure the sending and receiving of messages of other CANopen devices, such as the cyclic polling or monitoring of devices.

![CANopen Remote Device Tab]

Figure 37: “CANopen Remote Device” Tab (view with “Expert Settings” checkbox enabled)
Table 27: Key for Figure “CANopen Remote Device Tab”

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>Node ID</td>
<td>Selects the node ID or station address (1 … 126) that the PFC uses to communicate with the slave in the CAN network.</td>
</tr>
<tr>
<td><img src="image" alt="SDO Channels" /></td>
<td>Opens the “SDO Channel Configuration” window for configuring the service data objects (SDO). With some devices, this can be used to configure the communication parameters for the SDO transfer.</td>
</tr>
</tbody>
</table>

| ![Figure 38: SD Channel Configuration](image) |

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Channel</strong></td>
<td>Channel number</td>
</tr>
<tr>
<td><strong>Activate</strong></td>
<td>Activate channel</td>
</tr>
<tr>
<td><strong>NodeID</strong></td>
<td>Additional node ID for SDO transfers (e.g., a channel for communication with the master and a channel for the communication with another slave)</td>
</tr>
<tr>
<td><strong>Client COB-ID</strong></td>
<td>Resulting “Communication Object ID” of the client</td>
</tr>
<tr>
<td><strong>Server COB-ID</strong></td>
<td>Resulting “Communication Object ID” of the server</td>
</tr>
</tbody>
</table>

Enable Expert Settings When enabled, the following expert functions described are displayed

Generate all SDOs When enabled, SDOs are generated for all objects from the EDS file. In addition, the corresponding options must be activated. If writing nodeguarding objects is required, the checkbox for the “Nodeguarding” option must be enabled. When disabled, SDOs are only generated for the objects with default values that deviate from the EDS file.

Enable Sync Producing When enabled, synchronization messages are generated for synchronizing the process images between devices via this slave.

Optional Device When enabled, the CAN network is also “OPERATIONAL” (ready for communication via PDOs), if the device is not available.
### Table 27: Key for Figure “CANopen Remote Device Tab”

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Initialization</td>
<td>When enabled, no parameterization of the slave is performed by the master.</td>
</tr>
<tr>
<td>Reset Node</td>
<td>When enabled, the slave is reset by the PFC before the configuration is sent to the slave.</td>
</tr>
<tr>
<td><strong>Nodeguarding</strong></td>
<td></td>
</tr>
<tr>
<td>Enable Nodeguarding</td>
<td>When enabled, the slave monitors the PFC for possible interruptions during fieldbus communication.</td>
</tr>
<tr>
<td>Guard Time (ms)</td>
<td>Selects the interval in which the PFC expects “Confirmation” from the slave.</td>
</tr>
<tr>
<td>Life Time Factor</td>
<td>Sets the multiplication value for the “Guard Time.” If the time resulting from “Guard Time” x “Life Time Factor” (“Node Life Time”) has elapsed, the slave is switched to the predefined state.</td>
</tr>
<tr>
<td><strong>Heartbeat</strong></td>
<td></td>
</tr>
<tr>
<td>Enable Heartbeat Production</td>
<td>When enabled, the CAN device produces heartbeats at the ms intervals entered at “Producer Time.”</td>
</tr>
<tr>
<td>Producer Time (ms)</td>
<td></td>
</tr>
<tr>
<td><strong>[Heartbeat Consuming]</strong></td>
<td>Opens the “Heartbeat Consuming Properties” dialog.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Emergency</td>
<td>When enabled, the slave sends error and status messages (Emergency messages). When disabled, the SDO 0x1014 is not transferred to the slave. The standard setting of the slave therefore continues to be valid.</td>
</tr>
<tr>
<td>COB-ID</td>
<td>CAN identifier, preset: Node-ID + 0x80</td>
</tr>
</tbody>
</table>
Table 27: Key for Figure “CANopen Remote Device Tab”

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TIME</strong></td>
<td></td>
</tr>
<tr>
<td>Enable TIME Production</td>
<td>When enabled, the slave sends “Time” messages for synchronizing network devices.</td>
</tr>
<tr>
<td>COB-ID (Hex)</td>
<td>CAN identifier, preset: Node-ID + 0x80</td>
</tr>
<tr>
<td>Enable TIME Consumption</td>
<td>The slave receives “Time” messages.</td>
</tr>
<tr>
<td><strong>Checks at Startup</strong></td>
<td></td>
</tr>
<tr>
<td>Check Vendor ID</td>
<td>Manufacturer ID of the CAN device</td>
</tr>
<tr>
<td>Check Product Number</td>
<td>Product number of the CAN device</td>
</tr>
<tr>
<td>Check Revision Number</td>
<td>Revision number of the CAN device</td>
</tr>
</tbody>
</table>

“PDO Mapping” Tab

Process data objects (PDO) transport data. These are divided into send and receive objects (TPDO and RPDO). This tab displays and performs the mapping between process data objects (PDO).

The receive PDO is selected in the left-hand area, and the send PDO (from the point of view of the controller) in the right-hand area.

Figure 40: “PDO Mapping Tab”
Table 28: Key for Figure “PDO Mapping Tab”

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of PDO</td>
</tr>
<tr>
<td>Index</td>
<td>Index of PDO</td>
</tr>
<tr>
<td>SubIndex</td>
<td>Subindex of PDO</td>
</tr>
<tr>
<td>Bit length</td>
<td>Bit length of the PDO</td>
</tr>
<tr>
<td>Force explicit</td>
<td>When enabled, the displayed PDOs are no longer</td>
</tr>
<tr>
<td>deactivating of</td>
<td>used for communication.</td>
</tr>
<tr>
<td>PDOs</td>
<td></td>
</tr>
</tbody>
</table>

“Receive PDO Mapping”/“Send PDO Mapping” Tab

These tabs are displayed if the “Expert Settings” option is enabled in the “CANopen Remote Device” tab.

They display the receive and send PDOs.

“Service Data Object” Tab

This tab displays the Service Data Objects (SDO). Additional SDOs can also be created for parameterization. SDOs transfer parameters to the device configuration.

On initialization of the CAN bus, the set values will be transmitted to the slaves as SDOs.

![Figure 41: “Service Data Object Tab”](image)

Table 29: Key to Figure “Service Data Object Tab”

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line</td>
<td>Line number</td>
</tr>
<tr>
<td>Index:Subindex</td>
<td>Index and subindex of the SDO</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the service data object</td>
</tr>
<tr>
<td>Value</td>
<td>Value of the service data object</td>
</tr>
</tbody>
</table>
Table 29: Key to Figure “Service Data Object Tab”

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit length</td>
<td>Bit length of the service data object</td>
</tr>
<tr>
<td>Abort if error</td>
<td>Not used</td>
</tr>
<tr>
<td>Jump to line if error</td>
<td>Not used</td>
</tr>
<tr>
<td>Next line</td>
<td>Not used</td>
</tr>
<tr>
<td>Comment</td>
<td>Displays the comment, can be changed by selecting and overwriting the cell</td>
</tr>
<tr>
<td>[Move up]</td>
<td>Moves the entry up</td>
</tr>
<tr>
<td>[Move down]</td>
<td>Moves the entry down</td>
</tr>
<tr>
<td>[New…]</td>
<td>Opens a dialog window for adding an existing service data object from the list</td>
</tr>
<tr>
<td></td>
<td>Additional new service data objects can be created or selected service data objects changed in the lower area.</td>
</tr>
<tr>
<td></td>
<td>Adding/changing is performed by double-clicking an entry or via the [OK] button.</td>
</tr>
</tbody>
</table>

Figure 42: Adding Service Data Objects

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index:Subindex</td>
<td>Index and subindex of the SDO</td>
</tr>
<tr>
<td>Name</td>
<td>Name of SDO</td>
</tr>
<tr>
<td>Access Type</td>
<td>Displays which objects can be changed (RW)</td>
</tr>
<tr>
<td>Data type</td>
<td>Data type of the SDO</td>
</tr>
<tr>
<td>Default</td>
<td>Value from the EDS file</td>
</tr>
<tr>
<td>[Delete…]</td>
<td>Deletes one or several selected service data objects</td>
</tr>
<tr>
<td>[Edit…]</td>
<td>Opens a dialog for editing a selected service data object</td>
</tr>
<tr>
<td>SDO Timeout (ms)</td>
<td>Not used</td>
</tr>
</tbody>
</table>
“CANopen I/O Mapping” Tab

The tab displays the mapping of CODESYS IEC objects to CAN variables.

![Diagram of CANopen I/O Mapping Tab]

Figure 43: “CANopen I/O Mapping” Tab

Table 30: Key to Figure “CANopen I/O Mapping” Tab

<table>
<thead>
<tr>
<th>Column/Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Project variable</td>
</tr>
<tr>
<td>Mapping</td>
<td>Displays which variable is new and which variable already exists: The variable is not yet available, is newly created and can then be used in the entire project. Uses an already existing variable for mapping</td>
</tr>
<tr>
<td>Channel</td>
<td>Channel name</td>
</tr>
<tr>
<td>Address</td>
<td>Variable address</td>
</tr>
<tr>
<td>Type</td>
<td>Variable data type</td>
</tr>
<tr>
<td>Unit</td>
<td>Unit of the variable</td>
</tr>
<tr>
<td>Description</td>
<td>Variable description</td>
</tr>
</tbody>
</table>
Configuring the CANopen Slave:

The Configurator is opened after a slave is selected using the appropriate button in the “Settings” panel.

General Configuration

![Figure 44: CANopen Slave Configuration](image)

Table 31: General Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>Node ID</td>
<td>Selects the node ID or station address (1 … 126) that the PFC uses to communicate with the slave in the CAN network.</td>
</tr>
<tr>
<td>[Edit I/O area…]</td>
<td>Opens the dialog for creating I/O areas or process data objects and the corresponding object directory indices (slaves). The possible number and size of the I/O areas depends on the number of unused PDOs. The actual number of <strong>Used TxPDOs</strong> and <strong>Used RxPDOs</strong> is displayed below the table (max. 16 PDOs).</td>
</tr>
</tbody>
</table>
Table 31: General Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range Name</td>
<td>Displays the unique name of the range This is automatically entered but can be changed.</td>
</tr>
<tr>
<td>Count</td>
<td>Displays the number of inputs/outputs. The possible values depend on the data type and the actual number of PDOs.</td>
</tr>
<tr>
<td>Datatype</td>
<td>Displays the data type of the process data object: USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL</td>
</tr>
<tr>
<td>Index</td>
<td>Displays the index in the object dictionary of the slave that is assigned to the new object</td>
</tr>
<tr>
<td>Force new PDO</td>
<td>If this option is enabled (default), each new I/O area is mapped to a new PDO — even if the existing PDO mapping is not yet full. Otherwise, the actual PDO mapping is completely filled in before a new PDO is created.</td>
</tr>
<tr>
<td>[Add area…]</td>
<td>Opens a dialog for defining a range</td>
</tr>
<tr>
<td>[Delete area…]</td>
<td>Deletes ranges and associated data (e.g., mapping variables, PDOs, PDO mapping) of the selected entries</td>
</tr>
<tr>
<td>Used TxPDOs/Used RxPDOs</td>
<td>Displays the current number or receive/send process data objects The possible number and size of the I/O areas depends on the number of unused PDOs.</td>
</tr>
</tbody>
</table>
### Table 31: General Configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[OK]</td>
<td>Activates the new I/O range and displays it in the I/O overview table. Input/output channels and mapping variables are created. In the device's object dictionary, each I/O range receives its own index (output range: 0x3000 … 0x30FF, input range: 0x3800 … 0x38FF) and each individual input/output within the input range of a subindex. When an I/O range is created, PDOs are automatically created (RxPDOs start with 0x1400, TxPDOs with 0x1800). If the “Force New PDO” option is enabled (see above), the inputs/outputs are automatically mapped sequentially to these new PDOs.</td>
</tr>
<tr>
<td>[Cancel]</td>
<td>Closes the dialog and discards any changes</td>
</tr>
<tr>
<td>[Edit SDO Parameter area...]</td>
<td>Opens a dialog for creating configuration data objects. The description of the dialog is the same as that of “[Edit I/O area...].”</td>
</tr>
</tbody>
</table>

![Figure 47: Edit Parameter Range](image)

### EDS File

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor Name</td>
<td>Name of the device manufacturer</td>
</tr>
<tr>
<td>Vendor Number</td>
<td>Number of the device manufacturer</td>
</tr>
<tr>
<td>Product Name</td>
<td>Name of the product</td>
</tr>
<tr>
<td>Product Number</td>
<td>Number of the product</td>
</tr>
<tr>
<td>Revision Number</td>
<td>Revision number of the product</td>
</tr>
<tr>
<td>[Generate EDS File...]</td>
<td>This function enables an EDS file to be generated from the slave configuration.</td>
</tr>
</tbody>
</table>
CANbus Slave I/O Mapping

The window displays the I/O mapping of the slave side.

![CANbus Slave I/O Mapping](image)

Figure 48: CANbus Slave I/O Mapping

<table>
<thead>
<tr>
<th>Column/Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Project variable</td>
</tr>
<tr>
<td>Mapping</td>
<td>Displays which variable is new and which variable already exists:</td>
</tr>
<tr>
<td></td>
<td>![Icon] The variable is not yet available, is newly created and can then be used in the entire project.</td>
</tr>
<tr>
<td></td>
<td>![Icon] Uses a pre-existing variable for mapping.</td>
</tr>
<tr>
<td>Type</td>
<td>Variable data type</td>
</tr>
</tbody>
</table>
Configuring the CANopen Master

The CANopen master configurator opens when the master device is selected and the appropriate button in the “Settings” panel is clicked. It contains two tabs: “CANopen Manager” and “CANopen I/O Mapping.”

**CANopen Manager**

![CANopen Manager](image)

**Table 33: CANopen Remote Device**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Node ID</strong></td>
<td>Selects the node ID or station address (1 … 126) that the PFC uses to communicate with the slave in the CAN network</td>
</tr>
<tr>
<td>[Check and fix configuration...]</td>
<td>Opens a dialog for editing conflicts and for automatic repair</td>
</tr>
<tr>
<td><strong>Autostart CANopenManager</strong></td>
<td>When enabled, starts the CANopen Manager automatically (OPERATIONAL) as soon as all obligatory slaves are ready. Otherwise, the CANopen Manager must be started by the application using the “CiA405 NMT” function block.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Start Slaves</strong></td>
<td>When enabled, the CANopen Manager is responsible for starting the slaves. Otherwise the slaves must be started by the application using the “CiA405 NMT” function block.</td>
</tr>
<tr>
<td><strong>Polling of optional slaves</strong></td>
<td>When enabled, a slave that does not return a response during the boot sequence is polled regularly by the CANopen Manager until it returns a response. This option is currently enabled by default.</td>
</tr>
<tr>
<td><strong>NMT Error Behavior</strong></td>
<td>The NWT error behavior is not currently configurable.</td>
</tr>
<tr>
<td><strong>NMT Start All (if possible)</strong></td>
<td>The “NWT Start All” option cannot be selected at present.</td>
</tr>
<tr>
<td><strong>Sync</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Enable Sync Production</strong></td>
<td>When enabled, (disabled by default), the CANopen Manager transmits synchronization telegrams</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td><strong>Observe note on “Sync Production!”</strong></td>
</tr>
<tr>
<td></td>
<td>If “Sync Production” is enabled in the CANopen Manager, this function is disabled for all other bus stations.</td>
</tr>
<tr>
<td><strong>COB-ID (Hex)</strong></td>
<td>CAN ID of the SYNC telegram Possible value range: [1, 2047]</td>
</tr>
<tr>
<td><strong>Cycle Period (µs)</strong></td>
<td>Interval length (in microseconds) after which the synchronization telegram is sent</td>
</tr>
<tr>
<td><strong>Window Length (µs)</strong></td>
<td>Not configurable</td>
</tr>
<tr>
<td><strong>Enable Sync Consumption</strong></td>
<td>When enabled, (disabled by default), the receipt of synchronization telegrams of other devices by the CANopen Manager is displayed</td>
</tr>
<tr>
<td><strong>Heartbeat</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Enable Heartbeat Production</strong></td>
<td>When enabled, the CAN device produces heartbeats at the ms intervals entered at “Producer Time.”</td>
</tr>
<tr>
<td><strong>Node-ID</strong></td>
<td>Unique identification (1–27) of the Heartbeat producer on the bus.</td>
</tr>
<tr>
<td><strong>Producer Time (ms)</strong></td>
<td>ms intervals in which heartbeats are sent</td>
</tr>
</tbody>
</table>
Table 33: CANopen Remote Device

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIME</td>
<td></td>
</tr>
<tr>
<td>Enable TIME Production</td>
<td>When enabled, the slave sends a time stamp</td>
</tr>
<tr>
<td>COB-ID (Hex)</td>
<td>“Communication Object Identifier” which identifies the message's time stamp Default values: [0, 2047], default 100</td>
</tr>
<tr>
<td>Producer Time (ms)</td>
<td>Interval in milliseconds in which the time stamp is sent This value must be a multiple of the task cycle time.</td>
</tr>
</tbody>
</table>

CANopen I/O Mapping:

The window displays the master's I/O mapping.

![CANopen I/O Mapping](image)

Figure 51: CANopen I/O Mapping
8.2.3.3.2.3 PROFIBUS Configurator

The PROFIBUS Slave Configurator creates data points for communication between the IEC program and PROFIBUS. The following functions are provided:

- Creation of data points
- Deletion of data points
- Mapping of data points to existing IEC variables (CODESYS configuration user interface)
- Creation of data points as new CODESYS variables (CODESYS configuration user interface); these can be addressed directly in the program editor.
- Export of the configuration in the form of a device description file (*.gsd or *.gsg).

The configuration user interface consists of two main areas: “Data Points” tab and “I/O-Mapping” tab, which are arranged beneath the device images.

The “Data Points” tab lists defined data points and creates or deletes new ones. The data points are shown in tabular form. Each column represents a property of the data point. After a data point is created, properties (interdependent) can be changed. The individual table cells serve as entry or selection windows.

![Figure 52: PROFIBUS Connection Configurator](image)
Table 34: PROFIBUS Connection Configurator

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| Variable      | Displays the data point name  
The name is always identical to the name of the IEC variable |
| Data type     | Displays the data type of the data point, as well as the variable per IEC  
The following data types are supported:  
• BOOL  
• BYTE  
• WORD  
• DWORD  
• SINT  
• INT  
• DINT  
• USINT  
• UINT  
• UDINT  
• REAL  
• ARRAY OF BOOL  
• ARRAY OF BYTE  
• ARRAY OF WORD  
• ARRAY OF DWORD  
• ARRAY OF SINT  
• ARRAY OF INT  
• ARRAY OF DINT  
• ARRAY OF USINT  
• ARRAY OF UINT  
• ARRAY OF UDINT  
• ARRAY OF REAL |
| Direction     | Indicates whether it is input data or output data |
| Length (Array)| Displays the number of elements  
With simple data types (no arrays), the value is always set to 1.  
With arrays, the value can be changed. |
| Slot          | Displays the slot's number  
Each data point is automatically assigned a slot. This cannot be changed.  
The data of a slot is always transmitted in a PROFIBUS telegram. This guarantees the consistency of the transferred data. |
| Size          | Displays the size of the data point in bytes |
| Module designation | Displays the module name  
Each data point can be assigned a module name. If a user-defined name is not assigned, the name is generated from the data type and the number of elements. This is automatically updated with changes. A default name is also generated for deleted module names. |

Right-clicking inside the table opens the context menu for adding (New), removing (Delete) or rearranging (Up / Down) data points. These commands are
also provided in the menu ribbon. The menu ribbon also provides commands for creating and duplicating several data points.

![Figure 53: Context Menu of a Data Point](image)

The lower area of the PROFIBUS Configurator displays a gray status line. Clicking the text of the status line toggles between two views: “Used” and “Remaining.” The first view shows the slots, input bytes and output bytes already used. The second view shows the remaining bytes.

![Figure 54: Information in the Status Line of the PROFIBUS Configurator](image)

The “I/O Mapping” tab performs mapping to the variables (if required).

![Figure 55: PROFIBUS Connection Configurator](image)
Table 35: PROFIBUS Configurator

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>Displays the name of the assigned or created variable. This name is visible in the IEC program.</td>
</tr>
<tr>
<td>Mapping</td>
<td>Displays which variable is new and which variable already exists: The variable is not yet available, is newly created and can then be used in the entire project.</td>
</tr>
<tr>
<td>Channel</td>
<td>Displays the data point's name</td>
</tr>
<tr>
<td>Address</td>
<td>Displays the data point's IEC address</td>
</tr>
<tr>
<td>Type</td>
<td>Displays the variable's data type</td>
</tr>
<tr>
<td>Unit</td>
<td>Not used in this context</td>
</tr>
<tr>
<td>Description</td>
<td>Displays the description</td>
</tr>
<tr>
<td>[Reset mapping]</td>
<td>Deletes all mapping to existing variables</td>
</tr>
<tr>
<td>Always update variables</td>
<td>Defines whether the I/O variables are updated in the bus cycle task. The default value is defined in the device description.</td>
</tr>
</tbody>
</table>

**Use parent device settings**

- Updates the settings according to the settings of the parent device

**Activated 1 (use bus cycle task if not used in a task)**

- Updates the I/O variables in the bus cycle task if they are used in no other task

**Activated 2 (always in the bus cycle task)**

- Updates all variables in each cycle of the bus cycle task, regardless of whether they are used or whether they are mapped to an input or output channel
### 8.2.3.3.3 Device Detail View

The Device Detail view is opened by double-clicking a device in the Network view. The Device Detail view is designed for each specific product series. It adds modules to a head station (via Drag & Drop from the Product Catalog), for simple cable tests on digital input/output modules (lamp test by clicking LEDs), as well as an entry point for configuring displayed devices.

![Device View](image)

Figure 56: Device View (Example: Fieldbus Controller with Connected Modules)
Table 36: Elements of the Device View

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1   | **Head Station/Controller**  
Detailed view of a head station/controller in the project |
| 2   | **Arranged Modules**  
Detailed view of arranged modules  
To check the wiring, the LEDs of digital input/output modules can be switched on and off with a mouse click (if the device is connected online and control mode is active). |
| 3   | **Positioning Line for Modules**  
Position where new modules are added. Displayed if a module is dragged to the Device Detail view with the mouse. |
| 4   | **End Module**  
Display of the end module that terminates a node. End modules cannot be removed. |
| 5   | **I/O Mapping**  
Mapping and assignment of input, output, and memory addresses of the controller to project variables that are used by the application. |

Right-clicking a device displays the different buttons of the context menu:

![Figure 57: I/O Module Context Menu](image-url)
Table 37: Context Menu of a Device

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Copy" /></td>
<td>Copy</td>
<td>Copies selected modules</td>
</tr>
<tr>
<td><img src="image2" alt="Insert" /></td>
<td>Insert</td>
<td>Inserts copied modules</td>
</tr>
<tr>
<td><img src="image3" alt="Delete" /></td>
<td>Delete</td>
<td>Deletes selected modules from the node</td>
</tr>
<tr>
<td><img src="image4" alt="Connect" /></td>
<td>Connect</td>
<td>Opens an online connection to the node</td>
</tr>
<tr>
<td><img src="image5" alt="Disconnect" /></td>
<td>Disconnect</td>
<td>Removes the connection between the development and target systems</td>
</tr>
<tr>
<td><img src="image6" alt="Start" /></td>
<td>Start</td>
<td>Starts the application on the controller or in the simulation</td>
</tr>
<tr>
<td><img src="image7" alt="Stop" /></td>
<td>Stop</td>
<td>Stops the application on the controller or in the simulation</td>
</tr>
<tr>
<td><img src="image8" alt="Scan" /></td>
<td>Scan</td>
<td>Scans the node for the presence of modules and displays results in a list</td>
</tr>
<tr>
<td><img src="image9" alt="Upload" /></td>
<td>Upload</td>
<td>Reads parameters from the displayed devices</td>
</tr>
<tr>
<td><img src="image10" alt="Download" /></td>
<td>Download</td>
<td>Writes parameters to the displayed device</td>
</tr>
<tr>
<td><img src="image11" alt="I/O-Check" /></td>
<td>I/O-Check</td>
<td>Opens the “I/O CHECK” tab (see Section “I/O CHECK Tab”)</td>
</tr>
<tr>
<td><img src="image12" alt="Settings" /></td>
<td>Settings</td>
<td>Displays device settings in the corresponding panel (see Section “Panels” &gt; “Settings”)</td>
</tr>
</tbody>
</table>

The I/O mapping is displayed below the Device Detail view.
8.2.3.3.1 I/O Mapping

I/O mapping is the assignment of input, output, and memory addresses of the controller to project variables that are used by the application.

The display refers to the selected I/O module in the Device Detail view. If several I/O modules are selected, the I/O mapping of the last selected I/O module is displayed.

Figure 58: I/O Mapping

Table 38: I/O Mapping

<table>
<thead>
<tr>
<th>Column</th>
<th>Value (example)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variable</td>
<td>“Input01”</td>
<td>Input field for the name of the variable to be used for mapping the channel in the CODESYS application. Double-clicking the input field displays the [...] button for opening the Input Assistant.</td>
</tr>
<tr>
<td>Mapping</td>
<td></td>
<td>Symbol which displays a new or existing variable. The variable is not yet available, is newly created and can then be used in the entire project. Uses an already existing variable for mapping</td>
</tr>
<tr>
<td>Channel</td>
<td>“Input channel 1”</td>
<td>Displays the symbolic name of the channel.</td>
</tr>
<tr>
<td>Address</td>
<td>“%IB3”</td>
<td>Displays the address of the channel in the following format:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>%&lt;Memory area&gt;&lt;Data type&gt;&lt;Number(s)&gt;</td>
</tr>
<tr>
<td>Memory area</td>
<td></td>
<td>I Input/input memory area</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Q Output/output memory area</td>
</tr>
<tr>
<td></td>
<td></td>
<td>M Marker memory area</td>
</tr>
<tr>
<td>Data type</td>
<td></td>
<td>X Single bit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>B Byte (8 Bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>W Word (16 Bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>D Double word (32 Bit)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>For additional information on I/O mapping and addressing, see the CODESYS Online Help, “I/O Mapping.”</td>
</tr>
</tbody>
</table>
Table 38: I/O Mapping

<table>
<thead>
<tr>
<th>Column</th>
<th>Value (example)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>“BYTE”</td>
<td>Displays the channel data type</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The table cell remains empty if non-IEC data types are involved.</td>
</tr>
<tr>
<td>Unit</td>
<td>“ms”</td>
<td>Displays the unit of the parameter value</td>
</tr>
<tr>
<td>Description</td>
<td>“Process data input channel 1”</td>
<td>Writes the parameter</td>
</tr>
</tbody>
</table>
8.2.3.4 “Programming” Workspace

The “Programming” workspace is similar to the CODESYS 3.0 programming environment.

Different editors are displayed in the middle of the workspace, and are opened by double-clicking on objects in the Project Structure or via the “Edit” button in the menu ribbon. The tab carries the name of the selected object.

The “Programming” workspace also contains its own panels, which can be used for programming (breakpoints, call stack, cross reference list).
8.2.3.4.1 Program Editor

The program editor is displayed in a tab if a program (e.g., PLC_PRG) is double-clicked in the Project Structure or the “Edit” button of the menu ribbon is clicked. For additional information, see the CODESYS online help.

![Figure 59: Elements of the Program Editor](image)

Table 39: Elements of the Program Editor

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tabs</td>
</tr>
</tbody>
</table>
|     | Program editors, library managers, visualizations are displayed based on the selection in the Program Structure (double-clicking). Right-clicking a tab opens a context menu that contains the following functions: “Close,” “Close all windows apart from this one,” “Undock/Dock”, “New horizontal/vertical tab group” and “Move to the next/previous tab group”.
| 2   | Declaration Editor |
|     | This area normally appears above the program code. It contains the program name, the scope of validity and variable declarations (see CODESYS online help: “Options, Declaration Editor”). |
| 3   | Active Tabs |
|     | Shows active tabs in a selection list. |
| 4   | Text/Table Toggle Button |
|     | Toggles between textual and tabular view (see CODESYS online help: “Options, Declaration Editor”) |
| 5   | Language Editor |
|     | The program code appears here when using language editors such as for “Structured Text” (ST) or “Continuous Function Chart” (CFC). |
8.2.3.4.2 Library Manager

Libraries contain collections of reusable objects such as functions, function blocks or variables that are incorporated into projects. The contents of these libraries can be used here without any new declaration. The library manager is used to manage the libraries of a project.

The following libraries are supported:

- **User-Related Libraries**, which contain ready-to-use software solutions for manufacturing, building and process automation
- **Function-Related Libraries**, which contain technology functions and simple applications, such as for opening a communication connection
- **WAGO and CODESYS System Libraries** which enable complete system access

### Note

**Only use system libraries from WAGO!**

In order to use functions adapted to WAGO devices, only use WAGO's proprietary system libraries. You can recognize these libraries by the prefix “Wago” in the library name.

### Information

**Additional Information**

Further notes on the libraries are provided in the manual “Libraries for e!COCKPIT (Overview and Migration Instructions)”. The manual provides an overview of libraries that can be used in e!COCKPIT. It also provides instructions on importing software projects created in the WAGO-I/O-PRO, CODESYS-2 or CODESYS-3 programming environments. The appendix of this manual provides an overview of corresponding functions in libraries for WAGO-I/O-PRO and e!COCKPIT.

Additional information on converting CODESYS 2 and CODESYS 3 projects is provided in the CODESYS online help: (“Converting”).

### Note

**Permanently install libraries!**

Note that all libraries must be incorporated permanently. If you simply replace the library files in your local library directory, the libraries are not available in e!COCKPIT. They must be installed from the repository.
Table 40: Library Manager

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Add library]</td>
<td>Opens a dialog for library selection. The library manager adds to the project a reference to the library already installed in the repository.</td>
</tr>
<tr>
<td>[Delete library]</td>
<td>Deletes the reference to the selected library</td>
</tr>
<tr>
<td>[Properties]</td>
<td>Displays general properties of the selected library</td>
</tr>
<tr>
<td>[Details]</td>
<td>Displays details of the selected library</td>
</tr>
<tr>
<td>[Placeholders]</td>
<td>Displays placeholders used and adds target system-independent libraries. Depending on the device, the appropriate library is used; the standard library is used if no devices are available.</td>
</tr>
<tr>
<td>[Library repository]</td>
<td>Allows the addition and deletion of libraries from the Library Repository. Libraries are permanently installed in e!COCKPIT.</td>
</tr>
</tbody>
</table>
8.2.3.4.3 Task Configuration

A task is a temporal sequence and control unit in an IEC program. It is defined by name, priority, type and a start condition. Either temporal conditions or an internal or external event can be defined for starting the task.

If a device is added to a project, a task configuration with a main task is automatically added for this device in the Program Structure (see the following Section). This calls one or several POUs. Call sequence is defined by the combination of priority and condition.

Time monitoring for each task can also be configured (watchdog). The setting options depend on the target system.

The task configuration is opened by double-clicking the appropriate entry in the Program Structure. The “Properties” tab shows the basic settings, such as maximum values for tasks and watchdog parameters.

The “Monitor” tab displays the status and current statistics for cycle times in online mode.

For additional information, see CODESYS online help: “Task Configuration Editor > Dialog, Properties.”
8.2.3.4.4 “PLC_Task” Main Task

Configured tasks are displayed below the task configuration in the Program Structure. A “PLC_Task” main task is automatically created first for each task configuration. Tasks can be edited or deleted. The number behind the task indicates how many program calls are associated with the task. The task can be double-clicked to change the name. Other tasks can be extended by right-clicking them in the task configuration and choosing [Add New Element] > Task in the context menu.

For additional information, see CODESYS online help: “Task Configuration Editor > Configuration Dialog.”

Figure 62: “PLC_Task” Main Task

Table 41: “PLC_Task” Main Task

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority (1..15)</td>
<td>Indicates the task priority on a scale between 1 to 15</td>
</tr>
<tr>
<td></td>
<td>1 = highest priority</td>
</tr>
<tr>
<td></td>
<td>15 = lowest priority</td>
</tr>
<tr>
<td>“Cyclic” Type</td>
<td>Defines the time after which the task will be restarted. The cycle time</td>
</tr>
<tr>
<td></td>
<td>(e.g., “200”) corresponds to the task cycle time entered in the “Interval”</td>
</tr>
<tr>
<td></td>
<td>field in milliseconds “ms” or microseconds “µs.”</td>
</tr>
</tbody>
</table>

Note

Task priority can affect the runtime behavior of the controller!

Note that a set task priority in the critical range (1 .. 3) can considerably affect the runtime behavior of the controller.
Table 41: “PLC_Task” Main Task

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Note** | **Observe requirements for other bus systems!**  
When entering the task cycle time, consider system requirements. For example, in CAN bus systems, this must fit the currently set baud rate and the number of frames used on the bus.  
To prevent the loss of CAN frames, also enter multiples of the task cycle time for Heartbeat, Nodeguarding and Sync. |
| **Event** Type | Starts the processing of the task as soon as a rising edge occurs on the variable defined in the “Event” text field.  
The [...] button opens the Input Assistant with a list of all available global variables. |
| **External** Type | Starts task processing as soon as the event defined in the “Event” field occurs (events dependent on the target system). |
| **Freewheeling** Type | Begins task processing at the start of the program and restarts automatically in a continuous loop at the end of the task (no defined cycle time). |
| **Status** Type | Starts task processing if the variable of the “Event” field has the value TRUE. |
| Watchdog | Defines watchdog parameters for a task.  
Depending on the device used and definitions in the respective device description, setting options (upper/lower limits, default values, units, etc.) may be different than the settings shown. The watchdog can be switched off for specific cycles. |
Table 41: “PLC_Task” Main Task

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable</td>
<td>If this option is activated, the watchdog is active. The task ends with an error status if the entered cycle time incl. set tolerance (sensitivity) is exceeded. <strong>Several successive timeouts:</strong> A sensitivity of 0 or 1 generates the exception error in cycle 1, a sensitivity of 2 to n generates the exception error in cycle 2 to n. <strong>With Single Timeout:</strong> Exception error if the cycle time of the current cycle is longer than (time * sensitivity) Example: Cycle time 10 ms and sensitivity 5 → Exception error as soon as the task (once) is longer than 50 ms. If this option is deactivated, the watchdog is deactivated. There is no task watchdog.</td>
</tr>
</tbody>
</table>

[Add call] Opens a dialog for adding a program or POU call The dialog can likewise be opened in the Program Structure (right-click on a task > Program Call).

[Remove Call] Removes the selected program call

[Change Call] Opens the Input Assistant for selecting another program or POU

[Move up] Moves the selected program or POU up by one position in the processing sequence

[Move down] Moves the selected program or POU down by one position in the processing sequence

[Open POU] Opens the selected program in the programming editor

For additional information, see CODESYS online help: “Program Call”
8.2.3.4.5 Breakpoints

Breakpoints are used for debugging an application program. Breakpoints enable the program to be stopped at a defined position or executed in defined steps.

The breakpoint list is displayed in a window. This is opened via the menu ribbon, “VIEW” tab, [Breakpoints] button.

The “DEBUG” tab enables breakpoints to be set at the actual position in the Programming Editor ([New] button). It is also possible to toggle between the active and inactive state of a breakpoint ([Toggle] button).

Program symbols:
- Breakpoint active
- Breakpoint inactive
- Stop at the breakpoint in online mode

Figure 64: Breakpoints

Further information on creating and editing breakpoints can be found in CODESYS online help: (“Breakpoints”).

8.2.3.4.6 Call Stack

During the step-by-step execution of a program, this always shows the currently reached position in the call stack with a complete call path.

The “Call Stack” panel is opened as a panel via the “View” tab of the menu ribbon.

Figure 65: Call Stack

Further information on creating and editing call stacks can be found in the CODESYS online help: (“Call Stack”).
8.2.3.4.7 Cross Reference List

The cross reference list shows all occurrences of a variable in the project or the active editor.

The “Cross Reference List” panel is opened as a panel via the “View” tab of the menu ribbon.

Figure 66: Cross Reference List

Further information on creating and editing cross reference lists can be found in CODESYS online help: (“Cross Reference List”).
8.2.3.5 User-Defined Workspaces

*e!COCKPIT*s graphical user interface can be configured as needed. Predefined workspaces are provided for two main tasks in a window layout (“Network/Devices” and “Programming”).

These workspaces can be customized with user-defined settings, additional panels, window divisions, etc. and saved as a user-defined workspace. Use the [Manage] button within the “START” tab for this. After saving, a new button is displayed in the “START” tab through which the new workspace can be accessed.

![Figure 67: Creating and Managing User-Defined Workspaces](image)

The information for user-defined workspaces is stored for each user in the local Windows properties.

If the default “Network/Devices” or “Programming” workspaces were changed and overwritten with [Save the current workspace], the default settings can be restored at any time in the Backstage view, [Options] > [Standard workspaces] > [Reset standard workspaces].
### 8.2.3.6 Panels

The workspace in e!COCKPIT consists of different windows; within these windows, the panels can be arranged as required.

![Panels](image1)

Figure 68: Panels

The “Network/Devices” workspace displays the “Device Structure,” “Product Catalog” and “Settings” panels by default. The “Programming” workspace displays the “Program Structure” panel. This and other panels, such as “Messages” or the panels related to programming such as “Cross Reference Lists,” can be displayed via the “VIEW” tab.

![Displaying Panels](image2)

Figure 69: Displaying Panels

Panels can be docked, floated and grouped.

The selection panel synchronizes the selection in other windows or areas of the workspace. The commands and functions of individual tabs depend on the particular elements selected in the panels.
8.2.3.6.1 Device Structure

The Device Structure panel is used for display and providing navigation between the different devices integrated in a project. The panel is part of the “Network/Devices” workspace and is opened via the [Network/Devices] button in the “VIEW” tab.

The devices are organized in groups that are based on product segments of a Product Catalog, e.g., “Controllers,” or “I/O Systems.”

Device types are represented with different pictograms. A four-section display behind the device pictogram indicates a device’s status for the connection, PLC, internal bus and fieldbus (see Section “Workspaces” > “General Status Display for Connections”). This is followed by the device name based on the device description file. The name can be double-clicked in order to change it.

It is possible to toggle between Device Structure and Program Structure by clicking the tabs in the panel.

Other functions can be accessed via the context menu of a device. These functions are explained in the Section “Tiles.”
8.2.3.6.2 Program Structure

The “Program Structure” is used for display and providing navigation between the different devices integrated in a project. The panel is part of the “Programming” workspace and is opened via the [Programming] button in the “VIEW” tab.

The program elements are displayed by device group ("Application ([Device Name])").

A four-section status display provided in front of each application indicates the status with regard to connection, PLC, internal bus and fieldbus (see Section “General Status Display for Connections”).

A running program is indicated by a green circle icon displayed in front of it. Possible error states (such as compiler errors) are indicated with a red underline just as they are in CODESYS. Programs excluded from compiling are indicated in gray.

Right-clicking an element in the list opens a context menu that contains additional functions.

Figure 71: “Program Structure” Panel
Figure 72: Context Menu of an Application
### Table 42: Context Menu of an Application

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Copy" /></td>
<td>Copy</td>
<td>Copies the selected element</td>
</tr>
<tr>
<td><img src="image" alt="Insert" /></td>
<td>Insert</td>
<td>Inserts the copied element</td>
</tr>
<tr>
<td><img src="image" alt="Delete" /></td>
<td>Delete</td>
<td>Deletes the selected element from the Program Structure</td>
</tr>
<tr>
<td><img src="image" alt="Connect" /></td>
<td>Connect</td>
<td>Opens an online connection to one or several selected applications</td>
</tr>
<tr>
<td><img src="image" alt="Disconnect" /></td>
<td>Disconnect</td>
<td>Removes the connection between the development and target systems (controller or simulated device) and returns to offline mode</td>
</tr>
<tr>
<td><img src="image" alt="Start" /></td>
<td>Start</td>
<td>Starts the application on the controller or in the simulation</td>
</tr>
<tr>
<td><img src="image" alt="Stop" /></td>
<td>Stop</td>
<td>Stops the application on the controller or in the simulation</td>
</tr>
<tr>
<td><img src="image" alt="Editor" /></td>
<td>Editor</td>
<td>Opens the programming editor for entering program code</td>
</tr>
<tr>
<td><img src="image" alt="Settings" /></td>
<td>Settings</td>
<td>Opens a dialog in which program properties can be edited (see CODESYS online help: “Properties, General”)</td>
</tr>
</tbody>
</table>

Depending on the element that is right-clicked, other CODESYS elements can be appended to the structure, e.g., other program organization units (POUs), visualizations or variable lists. For additional information, see the CODESYS online help: “Adding Objects.”
8.2.3.6.3 Product Catalog

The Product Catalog displays all devices that are available in the system and have device drivers that are installed (Backstage view > [Product Catalog] > [Add Devices]). The devices can be sorted by product segment and protocol type.

**Note**

If the Network view is activated, only controllers are visible – not modules!

If Network view is active, no modules are shown in the Product Catalog. These are only displayed when Device Detail view is active. No internal WAGO modules (e.g., the internal bus) or end modules are shown.

![Figure 73: “Product Catalog” Panel](image)

Table 43: Buttons of the “Product Catalog” Panel

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
</table>
| ![Sort by product segment](image) | Displays the products by segment:  
• Operation and monitoring  
• Controllers  
• I/O systems  
• Accessories  
• Modules  
• Basic network functions  
• Other communication partners |
| ![Sort by communication protocol](image) | Sorts the devices by protocol type, such as:  
• MODBUS  
• PROFIBUS  
• CANopen  
• Serial interface |
| ![Move to the product catalog management](image) | Moves to the “Product Catalog” page of the Backstage view for incorporating new devices ([Add devices] button) |
**Table 43: Buttons of the “Product Catalog” Panel**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Display the available versions]</td>
<td>Displays in the tree structure all available device type versions of a device. This function can be switched on/off for each tab in the Product Catalog.</td>
</tr>
<tr>
<td>![Expand]</td>
<td>Opens all elements of the tree structure (up to device level)</td>
</tr>
<tr>
<td>![Collapse]</td>
<td>Closes all elements of the tree structure</td>
</tr>
</tbody>
</table>

Functions of the Product Catalog:

- To add a device to a project, individual devices are dragged from the Product Catalog and dropped onto a tile in the Network view.

  ![Figure 75: Dragging Devices to the Network View](attachment:image1.png)

- Hovering the mouse over a device will display a tooltip showing the product view, device name, type, order number of the device.

  ![Figure 76: Displaying Product Information](attachment:image2.png)

- A green “+” symbol is also displayed. Clicking this symbol adds a device to the next free position in the project.
Figure 77: Adding a Device to a Project
8.2.3.6.4 Settings

Device parameters are selected in the “Settings” panel. The parameters can be entered within defined limits. Entries in the “Settings” panel depend on the device selection and are based on the relevant device description of the selected device.

Complex modules or devices with more extensive settings, e.g., those involving CANopen or PROFIBUS parameters, can be set using integrated configurators. The configurators are opened from this panel via the relevant buttons. The setting options for these device configurators, as well as an explanation of individual parameters, can be found in the manual of the respective device.

![Figure 78: “Settings” Panel with the Button to a Device Configurator (CANopen)](image)

If a head station is selected in Device Detail view or a device in Network view, the “Settings” panel also serves as an entry point for opening network configurators for MODBUS, CANopen and PROFIBUS connections. Network configurators are described in Section “Fieldbus-Specific Configurators”.

Other PLC settings can be carried out via the second tab.
Figure 79: Additional Plc Settings
8.2.3.6.5 Message Bar and Message Window

*e!COCKPIT* features a message bar below the menu ribbon that displays any warnings, messages or errors during configuration, programming or validation. Up to three messages from different sources can be output. The info symbol on each line shows the meaning of the message: Error, warning or info.

![Message Bar](image)

Table 44: Message Bar Elements
<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Message type “Errors”</td>
</tr>
<tr>
<td>1</td>
<td>Name of the affected element, e.g., the program “PLC_PRG”</td>
</tr>
<tr>
<td>2</td>
<td>Short message description</td>
</tr>
<tr>
<td>3</td>
<td>Button for opening the message window</td>
</tr>
<tr>
<td>4</td>
<td>Close current message and display next message</td>
</tr>
</tbody>
</table>

Error messages can refer, for example, to validations of the Device Detail or Network view. CODESYS error messages, configuration errors (incompatible I/O module), compiler errors, warnings, etc. can be displayed. The test is performed continuously.

The complete message list is displayed in a message window. This is opened via the [Display Message Window] button of the yellow message bar or alternatively via the menu ribbon, “VIEW” tab, [Messages] button.

![Message Window](image)

Table 45: Elements of the Message Window
<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Description: Description of the message</td>
</tr>
<tr>
<td>2</td>
<td>Filter Messages/Warning/Errors filter are checked (blue background) or unchecked. [Delete messages] removes all messages from the list.</td>
</tr>
<tr>
<td>3</td>
<td>Object: Name of the project element</td>
</tr>
<tr>
<td>4</td>
<td>Position: Line in the source code</td>
</tr>
</tbody>
</table>
8.2.3.6.6 Other Panels

Besides the panels described, other device-specific panels may appear that are not included in this description. The contents of these tabs result from the particular device description file (DTP) used.

The “Call stack,” “Breakpoints” and “Cross reference list” panels are described in greater detail in the context of programming, see Section “Programming Workspace”.

8.2.3.6.7 Status Bar

The status bar displays all relevant connection information of the currently selected device with regard to the connection, PLC, internal bus and the fieldbus.

Figure 82: Status Bar

This information applies to one device. No status is displayed if several devices are selected. However, the connection status is displayed in the status bar if several applications of the same device with the same connection properties are selected.

The individual symbols are explained in the Section “Workspace” > “General Status Display for Connections”.

8.3 Backstage View (Files and Settings)

The project file itself, as well as other general basic settings of **e!COCKPIT**, are handled and managed via Backstage View, keeping these tasks separate from the processing of actual project content.

The buttons of the navigation bar in the left-hand window display the relevant pages in the right-hand window area or in a separate dialog. The individual pages are described in the following sections.

Open the Backstage view from the Start view via **[Other Projects]** or via the “FILE” tab of the main view.

![Backstage View](image)

**Figure 83: Backstage View**

**Table 46: Legend for Figure Backstage View**

<table>
<thead>
<tr>
<th>No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Back</strong></td>
</tr>
<tr>
<td></td>
<td>Returns to the previous view</td>
</tr>
<tr>
<td>2</td>
<td><strong>Pages</strong></td>
</tr>
<tr>
<td></td>
<td>Shows the content for handling and managing files, as well as basic settings for <strong>e!COCKPIT</strong> (see following Section)</td>
</tr>
<tr>
<td>3</td>
<td><strong>Navigation Bar</strong></td>
</tr>
<tr>
<td></td>
<td>Navigates via buttons to different pages that open in the right-hand window area</td>
</tr>
</tbody>
</table>
8.3.1 “New” Page

The “New” page creates a new project. Project templates provide special presets for creating a project.

Some templates in the right-hand window offer additional setting options, such as for creating a project with the “750 Series” template. This template creates a project with a fieldbus controller and an initial application. Create the project by clicking the [Create Project] button. The projects of templates that do not require any additional settings are created immediately when these templates are clicked.

Figure 84: Backstage View, “New” Page
8.3.2 “Open” Page

The “Open” page finds and opens documents.

![Open Page Screenshot]

Figure 85: Backstage View, “Open” Page

Table 47: Backstage View Operating Elements, “Open” Page

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Automatically saved]</td>
<td>Displays (if present) automatically saved projects</td>
</tr>
<tr>
<td></td>
<td>Besides manual saving (“Save”/“Save as” page), the project is also saved</td>
</tr>
<tr>
<td></td>
<td>automatically by default every 10 minutes. The activation/deactivation of</td>
</tr>
<tr>
<td></td>
<td>the automatic saving and setting of the save interval are made on the</td>
</tr>
<tr>
<td></td>
<td>“Options” page.</td>
</tr>
<tr>
<td>[Last used]</td>
<td>Displays the last projects used</td>
</tr>
<tr>
<td></td>
<td>Hovering the mouse over one of these projects will cause a gray pin to</td>
</tr>
<tr>
<td></td>
<td>appear on the right. Clicking the pin will permanently identify the</td>
</tr>
<tr>
<td></td>
<td>corresponding project in the list as a favorite.</td>
</tr>
<tr>
<td>[Computer]</td>
<td>Shows the last folder selected on the PC.</td>
</tr>
<tr>
<td>[Browse]</td>
<td>Opens files in other locations.</td>
</tr>
</tbody>
</table>
8.3.3 “Save” and “Save As” Page

“Save” stores the project under the currently used file name and memory location. The “Save as” page selects either the last folder used or another location via [Browse].

See “Options” for automatic saving of projects.

Figure 86: Backstage View, “Save As” Page
8.3.4  “Print” Page

The “Print” page generates a print version of the entire project content or selected objects.

![Print page screenshot]

Figure 87: Backstage View, “Print” Page

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Document project]</td>
<td>Opens a dialog for selecting contents of the current project that can be combined and printed</td>
</tr>
</tbody>
</table>
8.3.5  “Import/Export” Page

The “Import/Export” page enables projects from CODESYS 2 and 3 to be imported, as well as e!COCKPIT archives generated and opened.

Functions are provided for compiling and saving libraries ([IEC library] button) if a library is selected on the “New” page as a project template.

Figure 88: Backstage View, “Import/Export” Page
### Table 49: Backstage View Operating Elements, “Import/Export” Page

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Program elements</strong></td>
<td></td>
</tr>
<tr>
<td>Import</td>
<td>Imports program elements of an export file</td>
</tr>
<tr>
<td>Export</td>
<td>Save program elements in an export</td>
</tr>
<tr>
<td><strong>CODESYS project</strong></td>
<td></td>
</tr>
<tr>
<td>[Open CODESYS 2 project]</td>
<td>Imports CODESYS 2 projects (file suffix “.pro”)</td>
</tr>
<tr>
<td>[Open CODESYS 3 project]</td>
<td>Imports CODESYS 3 projects (file suffix “.project”)</td>
</tr>
<tr>
<td><strong>e!COCKPIT archive</strong></td>
<td></td>
</tr>
<tr>
<td>[Save archive]</td>
<td>Saves the current project on the controller as an archive file. The archive combines project-related files so that they can be forwarded to third parties. An archive combines all the files contained and referenced in the currently opened project into a central “*.projectarchive” archive file.</td>
</tr>
<tr>
<td>[Open archive]</td>
<td>Opens a project present on the controller as an archive file.</td>
</tr>
<tr>
<td><strong>IEC library</strong></td>
<td></td>
</tr>
<tr>
<td>[Compile and save]</td>
<td>Compiles and saves the library with the file suffix “compiled-library”</td>
</tr>
<tr>
<td>[Save in library repository]</td>
<td>Saves the library in the system repository</td>
</tr>
<tr>
<td><strong>smartDESIGNER</strong></td>
<td></td>
</tr>
<tr>
<td>[Import]</td>
<td><strong>[Import]</strong> loads all devices and modules of a project saved as a new project in e!COCKPIT from smartDESIGNER via CAE export.</td>
</tr>
<tr>
<td>[Export]</td>
<td><strong>[Export]</strong> saves all devices and modules in a file that can be read in from smartDESIGNER via the CAE import and further processed. The <strong>[Export]</strong> button is inactive if a project is not present or a library project is selected.</td>
</tr>
</tbody>
</table>
8.3.6 “Project Settings” Page

The “Project Settings” page is used to view and modify CODESYS-specific project settings. For additional information, see the CODESYS Help.

![Project Settings Page](image)

**Table 50: Backstage View Operating Elements, “Project Settings” Page**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Project information]</td>
<td>Opens the “Project Information” dialog to view or enter file information/settings, statistics and license settings (see CODESYS online help: “File, Project information”)</td>
</tr>
<tr>
<td>[Project settings]</td>
<td>Opens the “Project Settings” dialog for setting tasks such as compiler options and visualization. Also opens a dialog for security settings including project encryption, user and access rights administration, version handling and page settings for printing. The availability of individual categories partly depends on the installation scope (see CODESYS online help: “File, Project settings”).</td>
</tr>
<tr>
<td>[Project environment]</td>
<td>Opens the “Project Settings” dialog for managing versions of embedded libraries, compilers, devices, visualization profiles and styles (see CODESYS online help: “File, Project environment”)</td>
</tr>
</tbody>
</table>
8.3.7  “Product Catalog” Page

The “Product Catalog” page adds and deletes devices, as well as manages installed device description files.

Figure 90: Backstage View, “Product Catalog” Page

The Product Catalog content is shown on the left-hand side of the page (see also Section “Panels” > “Product Catalog”).

When a device is selected, the name, version and path of the installed device description file are shown on the right-hand page (“Content of the device description”).

The lower area lists all devices that belong to this description file and version (“Supported Devices”).

Table 51: Backstage View Operating Elements, “Product Catalog” Page

<table>
<thead>
<tr>
<th>Symbol/Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Add devices]</td>
<td>Imports a device description file. A window opens to select the description files.</td>
</tr>
<tr>
<td>[Delete devices]</td>
<td>Deletes the device description file for the device selected in the Product Catalog, as well as for all devices listed under “Supported devices” using the same device description</td>
</tr>
</tbody>
</table>
### Table 51: Backstage View Operating Elements, “Product Catalog” Page

<table>
<thead>
<tr>
<th>Symbol/Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Check projects on opening for new device descriptions</strong></td>
<td>When a project is opened, this automatically checks for new versions of device description files available for the included devices in the e!COCKPIT Product Catalog. If updates are needed, the versions replaced are indicated. The update is performed after confirmation. When disabled, there is no check for new device descriptions. The check can be performed manually and individually for each device (“Replace Device Description” function).</td>
</tr>
</tbody>
</table>
8.3.8 “Options” Page

The “Options” page contains general settings such as language selection or resetting of the workspaces, as well as CODESYS-specific settings. For additional information, open CODESYS Help.

![Backstage View, “Options” Page]

**Table 52: Backstage View Operating Elements, “Options” Page**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>[Language setting]</strong></td>
<td>Sets the language in which e!COCKPIT is started (German, English). The setting is activated when the software is restarted.</td>
</tr>
<tr>
<td><strong>[CODESYS options]</strong></td>
<td><strong>[Options]</strong> opens a window with general options that control the appearance of the CODESYS user interfaces embedded in e!COCKPIT (see CODESYS online help: “Options dialog”).</td>
</tr>
<tr>
<td></td>
<td><strong>[Library/visualization elements/visualization styles repository]</strong> opens library repositories, visualization elements and styles that are installed and used in CODESYS projects.</td>
</tr>
<tr>
<td></td>
<td>The repositories stored there are installed from the specified location permanently in e!COCKPIT. When changing to other libraries, for example, these must not be replaced in the local system folder but must be imported from the repository in e!COCKPIT and installed.</td>
</tr>
<tr>
<td><strong>[Restore]</strong></td>
<td><strong>[Restore]</strong> saves the current project at regular intervals if changes were carried out. The save interval can be set in minutes (default activated, 10 min).</td>
</tr>
</tbody>
</table>
Table 52: Backstage View Operating Elements, “Options” Page

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Standard workspaces]</td>
<td>[Reset standard workspaces] resets changes to the “Network/Devices” and “Programming” standard workspaces back to the installation settings.</td>
</tr>
</tbody>
</table>
8.3.9 “Help” Page

The “Help” page contains product and license information, product web pages, support addresses and update options.

![Backstage View, “Help” Page](image)

Table 53: Backstage View Operating Elements, “Help” Page

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Product information]</td>
<td>Displays the e!COCKPIT version.</td>
</tr>
<tr>
<td>[SOFTWARE LICENSE AGREEMENT]</td>
<td>shows e!COCKPIT conditions of use.</td>
</tr>
<tr>
<td>[Display information]</td>
<td>shows the integrated CODESYS version.</td>
</tr>
<tr>
<td>[Licensing]</td>
<td>Displays information on the used product license</td>
</tr>
<tr>
<td>[Enter licenses]</td>
<td>opens a dialog for entering access data.</td>
</tr>
<tr>
<td>[Updates]</td>
<td>Shows (can be filtered via “Updates”/“New”/“Installed”) all available updates, as well as previously installed versions of the e!COCKPIT software, device description files, communication drivers and IEC libraries</td>
</tr>
<tr>
<td>[Search for updates]</td>
<td>opens a connection with the Internet and searches for available updates.</td>
</tr>
<tr>
<td>[Install updates]</td>
<td>installs selected components.</td>
</tr>
<tr>
<td>“Check for updates and license information on starting e!COCKPIT”</td>
<td>checks at program start for available updates and displays them.</td>
</tr>
<tr>
<td>I accept the WAGO SOFTWARE LICENSE AGREEMENT</td>
<td>checks license information when installing updates. The license agreement must therefore be accepted to complete the installation.</td>
</tr>
<tr>
<td>Button</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>[Support]</td>
<td>Displays Support contact information Clicking the displayed email address causes a locally installed mail client (if present) to open an email window. Clicking the [Generate Email with Log Files] attaches all e!COCKPIT log files to your email. Click the [Save log files] button to create and save a log file.</td>
</tr>
<tr>
<td>[Website]</td>
<td>Displays the link to the e!COCKPIT software website This provides you with general information and additional links.</td>
</tr>
</tbody>
</table>
9 Operating

This section describes software operation through its stages and typical workflows.

9.1 Working with the User Interface

In this section, familiarize yourself first of all with the general settings of the user interface and the interaction between the workspaces and panels.

9.1.1 Using Online Help

The online help feature in e!COCKPIT contains the description of the e!COCKPIT user interface (1) and help for the integrated CODESYS 3.5.(2) software.

![Figure 93: Using Online Help](image)

Please note that some settings, windows, etc. in CODESYS online help may be named differently and that the workflows described may differ slightly from those in e!COCKPIT.
9.1.1.1 Calling Up Online Help

The e!COCKPIT online help function is context-sensitive – the help pages displayed relate directly to the currently active panel.

- To display all sections of the Online Help, click [?] at the top right of the window.
- To display the online help of a selected/active panel, window or element of the user interface, press the [F1] key.
- Help information on devices are provided in DTP files which are installed during the device installation. You can access the device help files via the Windows Start menu > All Programs > WAGO Software > e!COCKPIT > Device Help Files
- Press [F1] if you are in a program editor and require help on elements from libraries.
- In the Library Manager, context-sensitive information on selected elements of WAGO libraries is displayed in the “Documentation” tab. In the Library Manager click a library and an element (e.g., a module) in order to display more information.

Besides online help providing general information on entire windows or dialogs, tooltips are provided for individual buttons.

- To open a tooltip containing a short description of the button function, hover the mouse over a button.

9.1.1.2 Search

Users can perform searches in the online help function to narrow down the results displayed.

1. Click the [Search] button in the online help window.
2. Enter a search term in the text field.
3. Start the search by pressing the [Enter] key or by using the [Search] button.
9.1.2 Using Panels

9.1.2.1 Opening and Closing Panels

The workspace in e!COCKPIT consists of different windows; within these windows, the panels can be arranged as required.

1. Open the “VIEW” tab.

2. Click the appropriate button to display the Device Structure, Program Structure, Product Catalog, Messages or Settings panels. The “Settings” panel can be selected as soon as a project contains a device.

![Figure 94: Displaying Panels](image)
The buttons of visible panels are highlighted in blue.

3. To close a panel, click the ‘x’ on the panel itself.

![Figure 95: Closing a Panel](image)

9.1.2.2 Docking, Floating and Grouping Panels

To change the layout of the panels:

1. Click the title bar of the appropriate panel.

2. Hold down the mouse button and drag the panel to the required position.

The panel appears as a transparent blue object. Potential destinations are shown for the panel.
3. **Docking:**
To dock the panel at this position, release the mouse button precisely on the appropriate location icon.

**Floating:**
To float the panel on the workspace, release the mouse button at any other position.

**Grouping:**
Drag the panel over, underneath or between other panels so that a cross appears. Release the mouse button on the appropriate side of the cross to connect it with the desired panel.

To separate panels from each other, drag the particular panel to another area.
9.1.2.3 Docking and Floating the Main View

The tabs of the main view cannot be docked, however they can be floated on the workspace.

1. Click the tab of the main view, hold down the mouse button and drag the view to another position.

2. To centrally dock the main view again, move the view to the docking point at the center of the workspace and release the mouse button.

![Figure 98: Docking and Floating the Main View](image)

9.1.2.4 Using Multiple Selection

*e!COCKPIT* supports the multiple selection of elements in order, for example, to move several devices to the Network view or to easily change the settings of multiple devices.

You can always jointly select those elements that belong to the same class. For this reason, I/O modules and head stations cannot be selected together.

Some functions/buttons are deactivated if they cannot be applied to multiple elements simultaneously.

**Selecting Multiple Elements with [Ctrl] + Mouse:**

1. Click the elements to be selected while holding the Control key [Ctrl] down.

The selection appears in blue.
Selecting Multiple Contiguous Elements with [Shift] + Mouse:

1. To select a contiguous series of elements, select the first element with the mouse and click the last element while holding down the [Shift] key.

This selection is also only possible for elements of the same class. Other devices are skipped.

In Network view, click the first and last device to be selected from left to right and from top to bottom with the [Shift] key pressed to add the entire group of devices to the selection. This selection will also include devices that are not visible in the current view on the Network view workspace.

Selecting Elements with Frames (Network View):

1. Click the mouse and drag a frame around or across the device to be selected.

The devices touched by the frame are selected.
2. Move the selected devices to free tiles.

![Figure 101: Moving Multiple Selection](image)

**Deselecting Individual Elements with the [Ctrl] Key:**

1. In order to deselect individual elements from a selection, click them while pressing the [Ctrl] key.

**Canceling the Entire Selection:**

1. To deactivate the entire selection, press the [Esc] key.

**Invert Selection:**

1. To invert the selection click the [Invert selection] button in the “START” tab of the menu ribbon.
9.1.3 Using Workspaces

*e!COCKPIT*’s graphical user interface can be configured as needed. Predefined workspaces are provided for two main tasks in a window layout.

The predefined workspaces are provided in the “START” tab or the “VIEW” tab via the menu ribbon. The appropriate window layout is displayed in the workspace after selecting a workspace.

![Predefined Workspaces](image)

Figure 102: Predefined Workspaces

9.1.3.1 Opening the “Network/Devices” Workspace

1. To open the workspace for configuring the network and setting device parameters, click the [Network/Devices] button in the “START” or “VIEW” tab.

![Opening the “Network/Devices” Workspace](image)

Figure 103: Opening the “Network/Devices” Workspace

Alternatively, click the [Network/Devices] button in the Program Structure.

![Opening the “Network/Devices” Workspace](image)

Figure 104: Opening the “Network/Devices” Workspace

The [Network/Devices] button accesses two workspaces: the Network view and the Device Detail view. Network view is displayed first by default. If the Device Detail view is displayed, click **Network** in the Breadcrumb Navigation.
The “NETWORK” tab provides functions for configuring the network.

2. Move to Device Detail view to display a single selected device. To do this, double-click the device in the Network view.

Alternatively, click [Device details] in the “DEVICE” tab.

You can also open the Device Detail view by double-clicking a device in the Device Structure.

The particular device is displayed.
The “DEVICE” tab in the menu ribbon displays functions for configuring the device.

3. Click “Network” in the Breadcrumb navigation to return to Network view.

Alternatively, click the [Network] button in the “NETWORK” tab.

Moving between Network view and Device Detail view is possible by double-clicking the entries of the Device Structure:

4. To open Network view, providing an overview of all controllers, double-click “Controllers.”

5. Double-click the required device to open the Device Detail view.
You can choose between a graphical or tabular display both in Network view and in Device Detail view.

6. Click the appropriate button in order to move to the graphical or tabular view.

9.1.3.2 Opening the “Programming” Workspace

1. To open the “Programming” workspace for programming the controller, click the [Programming] button in the “START” or “VIEW” tab.

Alternatively, click the [Programming] button in the Device Structure.

The “Programming” workspace is displayed.
This opens the “PROGRAM” and “DEBUG” tabs in the ribbon, which contain additional functions for programming and debugging.

9.1.3.3 Using User-Defined Workspaces

You can save and manage your own window layouts in the “Workspaces” group in the “START” tab. A saved workspace is stored as a user-defined button in the menu ribbon. You can restore modified default workspaces at any time. The information for user-defined workspaces is stored for each user in the local Windows properties.

9.1.3.3.1 Creating a Workspace

1. Open the “START” tab.
2. Click [Manage] in the “Workspaces” group.
3. Click [Create new workspace].
4. Assign a name for the workspace.
   If required, select any pictogram of your choice (JPG, PNG or BMP) for the button via [Change image]. The image is automatically scaled to a size of 32 x 32 pixels.
5. To save the workspace, confirm your setting with [Accept].
6. The new button is displayed in the menu ribbon.
The active workspace is displayed on a blue background.

### 9.1.3.3.2 Editing a Workspace

**Changing Name and Image:**

1. Click [Manage] in the “Workspaces” group of the “START” tab.
2. Select the workspace to be edited under “User-Defined Workspaces.”
3. Click [Edit workspace].
4. Change the name and/or image of the workspace.
5. Save the changes by clicking [Accept].

**Changing the Window Layout:**

1. To make future modifications to the window layout, create the layout first: Rearrange the panels or open others.
2. Click the [Save the current workspace] button in the “START” tab.

The previously created workspace is overwritten with the new settings.

Please note that changes are only possible within one application area, i.e., the “Network/Devices” workspace cannot be changed to a “Programming” workspace. Changes refer only to panels and their visibility, position and size.

### 9.1.3.3 Deleting a Workspace

1. Click [Manage] in the “Workspaces” group of the “START” tab.
2. Select the workspace to be deleted under “User-Defined Workspaces.”
3. Click [Delete workspace].
9.1.3.3.4 Restoring a Workspace

The two predefined standard workspaces “Network/Devices,” and “Programming” can be changed with user-defined settings. To restore the default settings:

1. Move to Backstage view (“FILE” tab).
2. Select “Options.”
3. Click [Standard workspaces].
4. Click [Reset standard workspaces].
9.1.4 Using Key Combinations

Use key combinations to quickly access frequently required functions.

1. To enter key combinations, press the keys stated in the following list. Keys to be pressed simultaneously are marked with a plus sign (+).

9.1.4.1 General

Table 54: General Key Combinations

<table>
<thead>
<tr>
<th>Function</th>
<th>Key Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Save</td>
<td>[Ctrl] + [S]</td>
</tr>
<tr>
<td>Select All</td>
<td>[Ctrl] + [A]</td>
</tr>
<tr>
<td>Print</td>
<td>[Ctrl] + [P]</td>
</tr>
<tr>
<td>Remove Multiple Selections</td>
<td>[Esc]</td>
</tr>
<tr>
<td>Open Input Assistant</td>
<td>[F2]</td>
</tr>
</tbody>
</table>

9.1.4.2 “START” Tab

Table 55: Key Combinations, “START” Tab

<table>
<thead>
<tr>
<th>Function</th>
<th>Key Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Copy</td>
<td>[Ctrl] + [C]</td>
</tr>
<tr>
<td>Cut</td>
<td>[Ctrl] + [X]</td>
</tr>
<tr>
<td>Paste</td>
<td>[Ctrl] + [V]</td>
</tr>
<tr>
<td>Edit</td>
<td>[Ctrl] + [E]</td>
</tr>
<tr>
<td>Delete</td>
<td>[Del]</td>
</tr>
</tbody>
</table>

9.1.4.3 “PROGRAM” Tab

Table 56: Key Combinations, “PROGRAM” Tab

<table>
<thead>
<tr>
<th>Function</th>
<th>Key Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search</td>
<td>[Ctrl] + [F]</td>
</tr>
<tr>
<td>Replace</td>
<td>[Ctrl] + [H]</td>
</tr>
<tr>
<td>Next</td>
<td>[F3]</td>
</tr>
<tr>
<td>Next (selection)</td>
<td>[Ctrl] + [F3]</td>
</tr>
<tr>
<td>Previous Search</td>
<td>[Shift] + [F3]</td>
</tr>
<tr>
<td>Previous Search (selection)</td>
<td>[Ctrl] + Shift + [F3]</td>
</tr>
<tr>
<td>Build</td>
<td>[F11]</td>
</tr>
<tr>
<td>Connect</td>
<td>[Alt] + [F8]</td>
</tr>
<tr>
<td>Disconnect</td>
<td>[Ctrl] + [F8]</td>
</tr>
</tbody>
</table>
## 9.1.4.4 “DEBUG” Tab

Table 57: Key Combinations, “PROGRAM” Tab

<table>
<thead>
<tr>
<th>Function</th>
<th>Key Combination</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>[F5]</td>
</tr>
<tr>
<td>Stop</td>
<td>[Shift] + [F8]</td>
</tr>
<tr>
<td>Single cycle</td>
<td>[Ctrl] + [F5]</td>
</tr>
<tr>
<td>Toggle</td>
<td>[F9]</td>
</tr>
<tr>
<td>Step Over</td>
<td>[F10]</td>
</tr>
<tr>
<td>Single Step</td>
<td>[F8]</td>
</tr>
<tr>
<td>Step Out</td>
<td>[Shift] + [F10]</td>
</tr>
<tr>
<td>Write Values</td>
<td>[Ctrl] + [F7]</td>
</tr>
<tr>
<td>Force Values</td>
<td>[F7]</td>
</tr>
<tr>
<td>Unforce Values</td>
<td>[Alt] + [F7]</td>
</tr>
</tbody>
</table>
9.2 Using Projects

A project contains configured networks, as well as their entered devices and programming.

9.2.1 Creating a New Project

1. Launch e!COCKPIT.

2. Select a template in the Start view, e.g., “Empty Project.”

![Selecting a Template](image)

This creates a project. The main view is opened.

Additional Information

The “Empty Project” template can be used, for example, for online configuration in multi-controller projects.

The “750 Product Series” template can be used, for example, for offline configuration in single-controller projects.

9.2.2 Opening an Existing Project

1. Launch e!COCKPIT.

Recently edited projects are displayed in the Start view under “Last used.”

2. To open a project, click one of the displayed project entries.

   Alternatively click [Browse] to open projects of other sources ([Open] button) via the Start view.


**Dock the position of the project file!**
To keep frequently used projects in the “Last used” list and place them at the top of the list, click the gray pin symbol behind project name. The pin is shown in green.

![Docking the Position of the Project File](image)

Click the pin again to float the project once more from the position. The pin is then shown in gray.

### 9.2.2.1 Opening the Project with an Obsolete Device Description

When opening a project containing devices for which new device descriptions are available, this is shown in a dialog window. New descriptions may require a firmware update for your devices.

![Replacing Device Descriptions](image)

1. To display devices descriptions compatible with the current firmware, place a tick next to the checkbox “Only display device descriptions field that are compatible with the device firmware used”. Remove the tick to display all device descriptions. In this case the device description files that are not compatible with the current device firmware are also displayed.
I/O modules and devices that are not programmable are always displayed in the list and always updated irrespective of the set tick in this dialog.

### Note

**New device descriptions may require a firmware update!**  
New device descriptions may require an update of the firmware of your device. Update your firmware to continue using the device in your project.

2. Save the setting whether you wish to display only suitable device descriptions for the firmware of your device or all available device descriptions via “Do not display when opening this project.”  
   If you remove the tick, this dialog is then no longer displayed if new device descriptions exist.  
   If the tick is set, this dialog window continues to be displayed if new versions of device descriptions are available.

3. To update the displayed device descriptions click the [Replace device descriptions] button.  
   To close the dialog window without updating the device descriptions, click the [Retain device descriptions] button.

4. If you do not wish to update one device description but the device descriptions of all devices used in the project, click the [Replace all] button.

5. Replace the device descriptions as described in the Section “Opening project with obsolete device description”.

![Figure 124: Replacing All Device Descriptions](image-url)
9.2.3  Saving a Project

1. Create your project.

2. Open the Backstage view by clicking the “FILE” tab.

3. To save the project, click [Save] or [Save as].

4. Select a location. Click [Desktop], [Own Documents] or choose another memory location via [Browse].

5. Enter the file name and click [Save].

9.2.3.1  Saving Projects Automatically

Projects are saved automatically every 10 minutes if changes are carried out. Proceed as follows to activate/deactivate this function or change the save interval:

1. Open Backstage view.

2. Click [Options] in the navigation bar.

3. Click [Restore].

4. To activate the function tick “Restore − Save information”
   To deactivate the function, untick the option.

5. To change the save interval, enter the corresponding number of minutes in the input field.

Figure 125: Saving a Project Automatically
9.2.3.2 Restoring Projects after a Software/Hardware Crash

A dialog opens when the software is restarted after a software/hardware crash. You are notified of any restored project files.

1. Click [Yes] to display the project files.

The “Open” page of the Backstage view is opened. This page lists all projects that can be restored (in descending order by modification date).

If several instances of e!COCKPIT were opened before the crash, the projects of all instances are offered for restoring.

2. Delete any restored projects that are not required by clicking the cross next to the corresponding file.

3. Double-click one of the displayed files to open it.

Another dialog will open. You are asked whether you wish to open the restored file or the original document. The restoration of the original file discards the restoration file.

4. Select accordingly [Move to restoration view] or [Open original document].

5. To save the project click the “FILE” tab and then [Save].
9.3 Configuration and Parameterization

9.3.1 Offline Configuration

In offline configuration mode, you configure the devices and network first without connecting to real devices. The configuration is saved and transferred at a later time.

9.3.1.1 Adding Devices in the Project

If there are imported devices in the Product Catalog (see Section “Setting and Managing” >...> “Adding Devices”), they can be used as follows:

1. To add devices to a project, drag the individual devices (in this example, a PFC200) from the Product Catalog and drop them onto a free tile in Network view.
   To place devices on a free tile automatically, double-click the device name or click the green “+” symbol after the device name.

   ![Figure 127: Dragging Devices to the Network View](image)

2. Open the Device Detail view by double-clicking the device.

3. Drag the modules from the product catalog to any position after the head station.
This will display a [+] next to the mouse pointer. Positioning lines between existing modules indicate at what point the module will be placed when you release the mouse button. Alternatively, the module can also be added by double-clicking it from the Product Catalog behind the currently selected module or at the last location.

The Device detail view only allows the configuration of one head station with connected modules.

9.3.2 Online Configuration

In online configuration mode, you are physically connected to your devices in the network via e!COCKPIT. The configuration can be transferred directly. Click the [Connect] button to connect to available devices directly or perform a network scan first.

9.3.2.1 Scanning the Network and Devices

To display devices present in the network, scan the network first:

1. Click [Network/Devices] in the device/program structure.
2. Open the “NETWORK” tab.
3. If the Network view does not appear in the workspace, click the [Network] button in the menu ribbon.
4. First check the communication settings by selecting from the selection field in the “Scan Settings” group the medium in which the search will be performed, such as ETHERNET.
5. Click [Settings] to make other settings.

6. For instance, restrict the scan range for the ETHERNET settings.

7. Click [Accept].

8. To search for devices within the specified range, click the [Scan] button in the menu ribbon.

After the scan has finished, the search results of the configuration are displayed:

- Green: Configured and scanned device match.
- Red: Configured and scanned device are different.
- Blue: New device detected (if the device cannot be selected, this is an unknown device).
A red exclamation mark on a device indicates that it is not in the Product Catalog. In this case import the corresponding device description file for this device by importing it (“FILE” tab > [Product Catalog] > [Add Devices]).

9. Use the mouse to select the devices you wish to transfer to your project and click [Accept selection]. Click [Accept all] to transfer all devices.

10. With the mouse, select a device in the “Configuration” list. Click the device again and change the device name.

11. To switch from the scan results back to the graphical view (Network view), click the symbol for the graphical Network view.

12. To identify the modules of the scanned head station, double-click the device to enter Device Detail view.

13. Click [Scan modules] in the open “DEVICE” tab.
If the device is not connected, it is first automatically connected. The modules are then detected. The workspace shows the configuration and scan results. In the right-hand area, all modules which are connected with the device and which have their device description files loaded in e!COCKPIT are displayed.

A device description file can represent several modules if these have the same process data structure. A detected module may also be represented by several device description files. For this reason, there can be selection boxes for certain modules.

14. Select the used module in the selection box.

If there are several modules with the same scan result, the “Use for all suitable modules” checkbox is displayed in the selection field. Activate this checkbox to accept the selection for all suitable modules at the same time. The corresponding modules are marked for the transfer.
15. Tick the checkboxes in front of the modules that you wish to add to your project and click **[Accept selection]**. To transfer all modules, click **[Accept all]**.

The Device Detail view appears. You see the head station with the scanned modules.
Information

Additional Information
The number of modules detected is displayed in the Network view at the bottom of the device image.

![Network view](image)

Figure 140: Display of Applied Modules

9.3.2.2 Scanning the Modules of Several Devices

If you have scanned several devices (head stations), you can scan arranged modules for these devices together.

1. Open the Network view.
2. Use the mouse to select several devices.
3. Right-click one of these devices and click [Scan] button in the context menu.

![Scanning modules](image)

Figure 141: Scanning for Modules of Several Devices

e!COCKPIT scans the modules of the devices and shows these in the “Scan result” dialog.
4. Click [Accept] to add the displayed modules to the project.

9.3.2.3 Connecting and Disconnecting Devices

There are several ways to connect and disconnect devices. This function can also always be executed simultaneously for multiple devices (multiple selection).

1. Right-click the device to which you wish to establish a connection and click [Connect] in the context menu.

Alternatively perform the connection via the context menu of the device in the Device Structure.

The “Connect” function is also integrated in the menu ribbon. You can find the corresponding button in the “NETWORK”, “DEVICE”, “I/O CHECK”, “PROGRAM” and “DEBUG” tabs.
Additional Information

The [Connect] button is active if at least one of the selected devices has all the necessary information for opening a connection (e.g., an interface via which the connection can be made).

5. Click the [Connect] button.

The connection to the device or the devices is established. An automatic login is carried out in e!RUNTIME. The [Connect] button changes to a [Disconnect] button.

If two or more devices have different states (connected/restricted connection/not connected), these devices are disconnected on the first click of the button and are all connected with the second click.

If parameters were changed, this is displayed in a dialog.

6. If the displayed parameters are to be downloaded to the device, confirm this with [OK].

7. If the connection of the selected devices is to be removed, click the [Disconnect] button.

The status of the button changes back to “Connect.”

9.3.3 Parameterizing Devices

1. In the workspace of the Network/Device Detail view or in the device structure right-click the device to be parameterized.

2. Click the [Settings] button in the context menu.

The [Settings] button is also provided in the menu ribbon in the “VIEW” tab.

The “Settings” panel is displayed (see following Section).
9.3.3.1 Parameterizing Devices via the “Settings” Panel

Tabs, entry screens and selection fields in the panel depend on the particular device. Displayed setting options depend on the content of the particular device description file.

Figure 40: Opening the “Settings” Panel

1. Move between different tabs which contain settings groups by clicking the appropriate tab.

Figure 147: “Settings” Panel (Tab Navigation)

2. Tabs can contain subgroups. Open these by clicking the black arrow symbol.

The group is extended and marked with a green arrow symbol.

Figure 148: “Settings” Panel (Groups in Tabs)

3. Close the subgroups by clicking on the green arrow symbol.
4. Change the parameters and settings via the specific selection and entry options.

**Information**

**Additional Information**
The possible parameters, values and effects of parameter changes are described in the manual of the particular device.

### 9.3.3.2 Using Device-Specific Configurators

Simple modules can be parameterized via the “Settings” panel. Devices with more complex setting options, such as I/O modules that have mailbox functions, are assigned parameters in their own configurators. The configurators are opened from this panel via the relevant “Settings” buttons.

1. Click the device to be configured.
2. Click the appropriate button in the “Settings” panel to assign additional parameters.

![Figure 149: Opening Device-Specific Configurators](image)

**Information**

**Additional Information**
The possible parameters, values and effects of parameter changes are described in the manual of the particular device.
9.3.3.3 Uploading and Downloading Parameters

1. Right-click the device with the set parameters and click the **[Download]** button in the context menu.

![Figure 150: Writing Parameters to the Device](image)

The parameters are transferred to the device.

2. To read parameters from the selected device with the set parameters, click the **[Upload]** button in the context menu of the device.

![Figure 151: Reading Parameters from the Device](image)

The parameters are loaded from the device to the software and displayed in the “Settings” panel.

The **[Upload]** and **[Download]** buttons are accessed via the context menu of a device in the Device Structure or via the corresponding buttons in the “DEVICE” tab of the menu ribbon.
9.3.4 Configuring Networks

Besides the parameterization of individual devices you can also design complete networks via e!COCKPIT. Choose devices from the Product Catalog, assign roles (master/slave), drag connections between the connectors of the devices with the mouse and select communication protocols and communication parameters.

9.3.4.1 Creating a Simple MODBUS Network

1. Select two devices from the Product Catalog and drag them with the mouse into the Network view.

![Dragging Devices to the Network View](image)

2. To change the settings or parameters of the devices added, click the respective device, open the “Settings” panel in the context menu and make your changes there.

![Opening the “Settings” Panel](image)

3. Now connect the devices. Click the gray connector (ETHERNET/MODBUS) for one of the devices in the Network view and hold down the mouse button.

4. To connect the devices to each other, drag the connection line to the same type connector of the second device. Release the mouse button as soon as a green plus sign appears.

The line initially symbolizes the physical ETHERNET connection.
5. Click a device connector to open the interface.

6. Click the green plus sign.

Possible connections are displayed for each device. The selection depends on the device used.

7. Select the required connection. In this example, select “MODBUS via (TCP) Master”. You have thus defined the role and protocol used for this connection of the first device (role: client/master, protocol: MODBUS TCP).
8. A broken line is displayed. Similar to the first line, drag this to the connector of the partner device to establish a logical connection for the existing physical connection.

You have also defined the role and protocol of the second device for this logical connection (role: slave/server, protocol: MODBUS TCP).

The connection to the partner device(s) is displayed.

9. Move the mouse over the logical connection in the list, so that other buttons are visible.

10. Click the button to open the connection settings.
The MODBUS Configurator opens matching the role of the device in this connection. In this example, the connection configurator for the MODBUS slave/server is opened (open the configurator for the MODBUS master/client in the same way as for the first configured device).

Use the MODBUS Configurator to define which variables of the slave (server) are available on the master (client).

The figure below shows the MODBUS Configurator for the slave. No variables for communication between master and slave have been defined yet.

The configurator is divided into two columns:

The variables on the master are displayed on the left. The variables available on the slave and used in the slave application are displayed on the right. The tree displays them in a hierarchical view according to the program structure.

Create variables:
1. Right-click “PLC_PRG.”

The context menu opens.

2. Click the button to add a data point to create a variable in your program “PLC_PRG.”

![Figure 160: Creating New Variables on the Slave](image)

Click the new variable to display the settings for MODBUS access to the new variable “newVAR” in the bottom half of the Slave view (“Variable Settings”).

3. Enter settings in this area if you want to configure access to this variable via MODBUS (Description, Data Type, Array, Access, MODBUS Address).

The variable is created in the Slave application.

![Figure 161: New Variable in the Slave Application](image)

A variable is created in the master/client application by default (corresponding to the path (namespace) in the slave application). The variable is available on the master directly.

4. Click the master connector in the Network view.

5. Open the master configurator by clicking the button to open the connection settings.
The MODBUS Configurator for the master opens. The variable created on the master (1), the variable on the slave (2), the cycle time for updating the variables via MODBUS and MODBUS access to the variables (4) are displayed.

ReadOnly (RO): Master reads
WriteOnly (WO): Master writes
ReadWrite (RW): Master reads and writes

Open the “Programming” workspace and open the main program of the master.

In the master application, access to the “newVar” variable as shown in the following figure. Tip: Press the [F2] key to open the Input Assistant for easy variable selection.
Figure 165: Accessing the “newVar” Variable in the Master Application

In this way, a variable (“newVar”) has been declared in the slave application, made accessible via MODBUS and read in the master application.
### 9.3.4.2 Creating a Simple CANopen Network

In the following description you connect two CANopen devices and create data points for joint communication.

1. Select two devices from the Product Catalog (e.g., 750-8204) and drag them with the mouse into the Network view.

![Dragging Devices to the Network View](image1)

2. Select the first device and click the “DEVICE” tab in the menu ribbon.

3. Click the [Address Management] button.

4. Enter the IP address of the device in the entry field and click [Ok].

![Assigning an IP Address](image2)

5. Proceed in the same way with the second device. The IP addresses of both devices must be different.

6. Open the “Settings” panel by right-clicking the first device and clicking the [Settings] button in the context menu.
7. Assign the master role to the first device by selecting “Master” in the drop-down menu.

Figure 168: Opening the “Settings” Panel

8. Configure the second device (if not preset) in the same way, however, as a slave.

9. Create a logical connection between the devices by dragging a connection from the CANopen connector of the master to the CANopen connector of the slave.

Figure 169: Assigning the Role of the Device

Preparing the Slave:

10. Select the CANopen slave with the mouse.

11. Change if required the “Node ID” (e.g., “1”) and the baud rate (e.g., 250000) in the “Settings” panel.
    Note that “Node IDs” within a CANopen network must always be different.
Create a data point for the slave (UINT):

12. To display the logical connection to the master click the CANopen connector of the slave and then the button for opening the connection settings.

13. To add a data point, click first the [Edit I/O area...] button in the “General Configuration” tab.

14. Click the [Add area...] button in the following dialog.

15. Enter the I/O type, name, number and data type of your data point and confirm via [OK].
   Create several data types if required.

The created data points and your channels within the I/O range are displayed here (if the display is incomplete, close the dialog and reopen it.).

**Preparing the Master:**

17. Select the CANopen master with the mouse.

18. Change the “Node ID” (e.g., “2”) and the baud rate (e.g., 250000) in the “Settings” panel.

   Note that “Node IDs” within a CANopen network must always be different.

The configuration of the slave was changed. This is indicated as a warning symbol (yellow triangle with an exclamation mark). The configuration must be updated in order for all created slave data points to be available in the master.

19. To do this click the connector of the master and click the button for updating the configuration.

20. In the “Choose Update Behavior” dialog activate the “Create New” checkbox.
The configuration is updated. The warning symbol disappears.

21. Open the CANopen connection configurator for the master. To do this click the connector of the CANopen master and click the button for opening the connection settings.

22. Open the “CANopen I/O Mapping” tab.

23. Assign variables to the channels created in the slave by clicking on the appropriate variable field and entering a variable.

Alternatively, use the Input Assistant by clicking the [...] button.

The configuration is then complete. The variables can be used in e!RUNTIME programs of both devices.
9.3.4.3 Creating a Simple PROFIBUS Network

To create a PROFIBUS network, you need at least one PROFIBUS master and one slave. The PROFIBUS slave function is provided with a PFC200 (order number 750-8206). The configuration of the PROFIBUS slave requires a connection to a generic PROFIBUS master.

1. Place the generic PROFIBUS master and the PROFIBUS slave in the Network view.

![Figure 179: Placing the PROFIBUS Master and Slave in the Network View](image)

2. Create a connection between both devices by connecting up the PROFIBUS connectors (purple) with each other.

![Figure 180: Connecting PROFIBUS Devices](image)

In a PROFIBUS network, each device is assigned a unique address. The PROFIBUS address is set in the “Settings” panel.

3. If the “Settings” panel is not yet opened, open it by right-clicking the slave device and clicking the [Settings] button.
4. Click the tab in the “Settings” panel in which the PROFIBUS address is set.

5. Enter a unique address in the “PROFIBUS Address” entry field.

The communication in the PROFIBUS network is performed using data points. These are set in the configurator of the PROFIBUS connection for the PROFIBUS slave.

6. Open the configuration interface (PROFIBUS interface) via the context menu of the PROFIBUS connector.

The user interface of the configurator is opened.
The configuration user interface consists of two main areas, the “Variables” tab and “I/O Mapping” tab, which are arranged beneath the device images.

In the “Variables” tab, defined variables are listed and new ones are created. The “I/O Mapping” tab is used to perform (if required) the mapping to the variables.

7. Create a data point via the context menu in the “Variables” tab or by clicking the appropriate [New] button in the menu ribbon “Configurator: PROFIBUS-DP-V1.”

The variable is created. The “Name”, “Data type,” “Direction,” Slot,” “Size” and “Module Designation” are provided with default values. These values are assigned for each new variable created.

Create additional variables and adjust the values as required.
To create multiple variables at once or to duplicate existing data points, use the corresponding [Create Multiple Variables] or [Duplicate Variable] buttons in the menu ribbon. In both cases these open a dialog in which multiple values can be set for the newly created variables.

Place variables at any location!
If you have not selected a variable from the list, the new variable is placed at the last position. If you have selected one or multiple variables, the new variable is placed behind the last selected variable. This enables you to insert the new variable at any position.

You can delete the variable if required in the same way (context menu of the corresponding variable or [Delete Variable] button in the menu ribbon).

After a variable name is assigned, an IEC variable with the same name (global variable list) is created automatically in the background. You can use this variable in your IEC program.
8. If you have already created IEC variables and wish to map them to data points, click the “I/O Mapping” tab to open the corresponding configuration user interface.

There are two ways of mapping a data point to an existing IEC variable:

9. Double-click the corresponding variable name and enter the name of the existing IEC variable manually.

10. Or click the [...] button to open the Input Assistant.

11. In the Input Assistant select the IEC variable and confirm with [OK].
The mapping is performed.

After the variables are created and optional mapping, you can save the configuration in the form of a device description file.

12. To do this, click the [Export device description (GSD)] button.

13. Enter a name and confirm with the [OK] button.

Two files are saved. The two files contain identical configurations. Only the description language of the parameters is different. The file with the suffix "*.gsg" contains German text. The file with the suffix "*.gsd" contains English text.

The exported device description can be imported by other engineering tools.
9.4 Testing

9.4.1 Performing an Input/Output Test for Digital Input/Output Modules

To verify the wiring of digital input/output modules, switch an LED on or off with e!COCKPIT.

The input/output test requires the device to be connected online and Control mode to be active (“I/O-CHECK” tab).

1. Open the Device Detail view.
2. Click the LED of the relevant module.

![Figure 194: Switching an LED On/Off with the Mouse](image)

3. Verify if the device’s LED mirrors its on-screen behavior in e!COCKPIT.

If this is the case, the module is connected correctly.

The display is reset as soon as the Device Detail view is closed.

9.4.2 Performing a Systematic Input/Output Test (I/O CHECK)

Check for correct wiring of your node using the systematic input/output test. You can test both individual modules and entire nodes.

Testing Individual Modules:

1. Open the Device Detail view.
2. Click a module.
3. Click the [Build] button in the “I/O-CHECK” tab of the menu ribbon.

A tab is opened in the workspace, in which the selected module is displayed.
4. To add additional modules to this test, repeat steps 1–3.

Testing an Entire Node:

1. Open the Network view.

2. Click a device.

3. Click the [Build] button in the “IO-CHECK” tab of the menu ribbon.

An overview of input/output channels of the corresponding modules is displayed in both cases.

4. Click the [Connect] button in the menu ribbon.
The monitoring function is active first. The current values are displayed. If required, the “Result” column can be switched on/off via the button of the same name in the menu ribbon.

5. Compare the displayed values with your hardware, e.g., whether the green LED is actually lit and for each entry click either [Ok] or [Error].

![Figure 197: Performing a Test](image)

Depending on your entry, the status is shown in the “Result” column.

![Figure 198: Status Display for Inputs and Outputs](image)

Besides value monitoring, you can also write or read them.

6. Switch to control mode by clicking [Control mode] in the menu ribbon.

Change the individual values. For this use, the slide adjuster and entry fields for analog output terminals; for digital output terminals click, the LED symbol.

![Figure 199: Changing Analog and Digital Values](image)
Changed values are indicated with an asterisk (*) and the number of prepared changes shown as a number in the menu ribbon (here: 3) if “Collect” is active (blue background).

7. To load all changed values to the I/O module click the **Write now** button in the menu ribbon. The write operation is performed once.

If the **Direct** button is clicked, the values are continuously written as they are entered.
9.5 Programming

*e!RUNTIME* is integrated in *e!COCKPIT* user interface. This is the 3S hardware-independent CODESYS IEC 61131-3 programming system (version 3.5). *e!COCKPIT* can assign elements for program development to (fieldbus controllers), thus enabling for example the implementation of fieldbus and I/O configurations, visualizations, motion control and other tasks.

1. Open the “Programming” workspace.

![Program structure](image)

Figure 201: Opening the “Programming” Workspace

A CODESYS application with a library manager, “PLC_PRG” program and a task configuration is created for each head station incorporated. Other applications can be created on the same level or below.

A “project library” is created for POU,s, which applies to all applications.

You can incorporate additional CODESYS elements via the context menu in the Program Structure, such as additional POU,s, files, variable lists or visualizations (see CODESYS online help, “Adding an Object”).

Right-click the project library and choose [Copy] to copy, for example the entire content of this directory, to a directory of an appropriate project application. Copying and inserting is also possible between different instances of *e!COCKPIT*. 
“POUs” are program organization units/objects that form a controller program. When you create a POU, the workspace is adapted accordingly. Depending on the selected programming language, other tabs are added to the menu ribbon and new panels are shown (e.g., “Tools”).

Figure 202: Context Menu in the Program Structure
The use of individual panels is described in the CODESYS online help.

The context menu of an application also enables you to copy, insert or delete the appropriate application, connect the device, start and stop the application, open the programming editor or view/change the settings of the application.
9.5.1 Creating Libraries

1. Select a library template in the Start or Backstage view.

![Selecting a Template for Libraries](image)

Figure 204: Selecting a Template for Libraries

The main view is opened.

2. Open the “Programming” workspace.

3. Create your library.

![Creating a Library](image)

Figure 205: Creating a Library

4. To check the created library, click the [Syntax check project library] button in the “PROGRAM” tab of the menu ribbon.

This checks the written library according to IEC 61131-3.

5. After a successful check move to the Backstage view (“FILE” tab).
6. Click [Import/Export].

7. Click the [IEC library] button.

8. Saving and installing a library:

   - To save the library as a compiled library with the “compiled-library” suffix, click [Compile and save].
   - To save the library in the system repository, click [Save in library repository].

![Figure 206: Saving and Installing a Library](image)

### 9.5.2 Adding Libraries

Libraries contain collections of reusable objects such as functions, function blocks or variables that are incorporated into projects. The contents of these libraries can be used here without any new declaration. The library manager is used to manage the libraries in a project. This displays both application-related and function-related libraries as well as CODESYS and WAGO system libraries.

WAGO system libraries are already supplied. Add libraries that are not present as follows.

1. Open the “Programming” workspace.

2. Right-click the “Project Library (POUs)” entry and click **Library Manager** in the context menu.
3. Click [Add] in the dialog.

The library manager is opened in the workspace.

4. To add a new library that has not been installed on the system, click [Add library].

5. Select the relevant library in the dialog.

6. Confirm your selection with [OK].

The library manager incorporates a reference to a pre-installed library into the project.

To add a library that is not yet installed on the local system, click the [Library repository] button to install it.

To add target-system neutral libraries, use the [Placeholders] button instead. Depending on the device used and its device description, the appropriate library is used. Alternatively, the standard library is used if no device is present, so that the program can be compiled error-free.

**Note**

**Install libraries in e!COCKPIT!**

Note that all libraries must be incorporated permanently. If you simply replace the library files in your local library directory, the libraries are not available in e!COCKPIT. They must be installed from the repository.
9.5.3 Creating Programs

The “PLC_PRG” main program is created automatically. You can create additional programs.

1. Right-click the corresponding application and select [POU] in the context menu.

2. In the dialog assign a name and select the type and the implementation language per IEC 61131-3.

   ![Image of POU dialog](image)

   Figure 208: Creating a Program

   The programs are automatically assigned to a task. If the task was deleted, you can create this again by right-clicking the application and selecting [Task Configuration] in the context menu.

3. To assign a program to the task, click the created task and add a **Program call**.

   ![Image of adding a program call](image)

   Figure 209: Adding a Program Call

   A “Program call” dialog is opened.

4. In the “POU to Call” field, enter the program name.
Alternatively, use the Input Assistant by clicking the […] button, select the program in the new window and confirm with [OK].

![Figure 210: Entering the Program Name via the Input Assistant](image)

The program name is entered in the “Program call” dialog.

5. To add the program call, click the [Add] button.

6. To open and edit the added program double-click the program in the Project Structure.

Depending on the implementation language you have selected under point 2, the corresponding programming editor is displayed.

![Figure 211: Programming Editor (in this example: Structured Text (ST))](image)

7. Write your program according to IEC 61131-3.

9.5.4 Performing I/O Mapping

By I/O mapping is meant the mapping and assignment of input, output, and memory addresses of the controller to project variables that are used by the application. The I/O mapping is displayed in the Device Detail view below the device image.

1. Open the Device Detail view.

2. Select the I/O module with the mouse.

3. Click the “Variables” column in the corresponding fields and enter the variables/data points.

![Figure 212: Entering Variables/Data Points](image)

The variable or data point is created and is available in the entire project.
You can use the Input Assistant to assign variables that are already known in the project.

4. To do this, double-click a field.

5. To open the Input Assistant, click the [...] button.

![Figure 213: Entering Variables/Data Points via the Input Assistant](image)

6. Select the corresponding variable or the data point and confirm with [OK].

The mapping is accepted for the project.

![Figure 214: I/O Mapping](image)
9.5.5 Compiling an Application

1. To compile the program click the [Build] button in the “PROGRAM” tab of the menu ribbon.

   The program is compiled and a message bar shows the errors.

   Figure 215: Compiling an Application

2. To display details of the validation, click [Display message window]. Alternatively, open the message window via the [Messages] button in the lower left area of the workspace.

3. All errors, warnings and messages are displayed first. To filter the view, click the appropriate button in the message window. Active displays are highlighted in blue.

   Figure 216: Message Window

4. Click an entry in the list to jump directly to the error location.

5. To clean the list, click the [Delete messages] button.
9.5.6 Excluding Programs for Compiling

If you wish to connect and compile an application, you can exclude individual programs that must not be compiled.

1. Right-click in the Program Structure the program that must not be compiled, e.g., “PLC_PRG.”

2. Click Settings in the context menu.

3. In the dialog click the “Build” tab and activate the “Exclude from compiling” check box.

4. Click [OK] to confirm.

The program is displayed in gray and in italics.
9.5.7 Debugging an Application

The debug function examines programming errors. This requires a connection to the device.

1. Move to the “DEBUG” tab in the menu ribbon.

2. To start the debug process, click [Start].

3. To create a watch window, click the [Monitoring] button in the menu ribbon and select one of the five entries.

This will open the “Watch” panel.

4. Double-click the “Expression” field and then the […] button displayed there for the Input Assistant.

5. Select the value or values to be monitored.

6. Transfer the value or values by clicking [OK].

7. Click the [New] button in the “Breakpoints” group to add breakpoints.

**Breakpoints** can be set at specific positions in the program to stop execution. Specific conditions can be defined for the stop (see CODESYS online help, “Breakpoints”).

8. Then click [Toggle] to activate the breakpoint.

Step-by-step processing via **Single Step** (menu ribbon, “DEBUG” tab) enables the program to be run in controlled steps. The particular variable values can be examined at each stop. A **Call stack** (menu ribbon, “VIEW” tab) can be displayed to determine the current position.

The flow control can be activated to track previous sections of the application program. Unlike standard monitoring, which only displays the value of a variable between two cycles, the flow control supplies the value for each processing step precisely at the time of processing.
9.5.8 Simulating an Application

Simulation mode enables you to debug and test the behavior of an application without connecting to a real target system. No communication settings are required for the simulation.

1. Open the “PROGRAM” tab in the menu ribbon.

2. Click [Simulate application].

The button in the menu ribbon changes to [Close simulation]. In the status bar, the status display in front of the corresponding device and in the tooltip of the device (“Program structure” panel) the connection is shown in blue. Also in the Network view the tile of the corresponding device is outlined in blue.

3. Use the commands of the “DEBUG” tab to test the application.
9.5.9 Creating Visualizations

*e!COCKPIT* integrates the visualization functions of CODESYS. These enable the use of predefined visualization elements and commands for simulating, controlling and monitoring machine and plant systems.

A web server connection (CODESYS-Web-Visu) also enables a visualization to be operated with a web browser or displayed on an external device/display (CODESYS-HMI).

---

**Note**

Ensure compatibility with web browsers!
The view of the web visualization has been tested and optimized for the following web browsers:
- Firefox 30.0
- Chrome 34.0, 35.0
- Internet Explorer 11
- Opera 12.17
- Android 4.0
- Safari iPad2

---

**Note**

If the connection establishment fails, reduce the number of visualization tabs!
If many tabs containing the visualizations are opened in the programming workspace, it may not be possible to make a connection to a device. Use the “Close all windows apart from this one” function in the context menu of the tab to retain only the current tab and close all the others. Try again to connect the application.

1. First open the “Programming” workspace.
2. Add a visualization by right-clicking an application and selecting [Visualization] as a new element.
The visualization editor is opened in the workspace. This will display the “ToolBox” and “Properties” panels.

3. Use the elements from the “ToolBox” panel, such as the bar display, by placing them in the visualization editor via drag and drop.

Assign a value to the display:

4. Click the “Value” field and press [F2] to open the Input Assistant.

5. Via the Input Assistant select a value which will be used for the visualization. This example uses the value “i” from the “PLC_PRG.”

The value is accepted:
6. In the “DEBUG” tab of the menu ribbon, click [Simulate application].

7. To start the visualization click [Start].

This will start the visualization.
9.5.10 Connecting

To perform online actions such as a program download, program execution etc. for this application, connect the application with the target system.

1. **“Network/Devices” Workspace:** In the context menu of a device or via the menu ribbon (e.g., “DEVICE”, “NETWORK” tabs) click [Connect]. The device is switched to online mode. All applications of the device are active/connected online.

   **“Programming” workspace:** In the context menu of one or several selected applications or in the menu ribbon (“PROGRAM”, “DEBUG” tabs) click [Connect]. The selected applications of one or several devices are active/connected online.

2. If an application was not previously stored on the controller, a prompt will ask whether an application should be created or downloaded. Confirm this with [Yes].
9.5.11 Downloading Applications to the Device

1. To download the selected application to the controller, click [Program Download] in the “PROGRAM” menu ribbon.

To load multiple applications/parameters on the device and set other options (e.g., Online Change), click [Multiple download] in the “PROGRAM” menu ribbon.

![Multiple download](image)

Figure 224: Multiple Download

2. All devices with associated applications and parameters are preselected by default. Use the mouse to untick components that are not to be loaded as required.
9.5.12 Importing CODESYS 2 and CODESYS 3 Projects

To load projects that you have created in CODESYS 2 (file suffix “.pro”) or CODESYS 3 (file suffix “.project”):

1. Open Backstage view (“FILE” tab).
2. Click the [Import/Export] buttons and then [CODESYS project].

![Figure 225: Loading a CODESYS Project](image)

3. Click [Open CODESYS 2 project] or [Open CODESYS 3 project].
4. Select the project file and click [Open].

This starts the project import.

*e!RUNTIME* uses new libraries which have different names and content to the libraries of previous CODESYS projects.

5. If the following window is displayed for converting a library reference, first select the following setting: “Ignore the device. All application specific objects will not be available in the new project.”

![Figure 226: Converting a Library Reference](image)

The window is displayed for each new library not found. You require a new version for these libraries.
6. Click [OK].

This allows you to only load programs and devices first of all. You can update new libraries later.

---

**Information**

**Additional Information**

Further information on the libraries are provided in the manual “Libraries for e!COCKPIT (Overview and Migration Instructions)”.

The manual provides an overview of libraries which can be used in e!COCKPIT. Instructions are provided for migrating software projects which were created in the WAGO-I/O-PRO, CODESYS 2 or CODESYS 3 programming environments. The appendix of this manual provides an overview of corresponding functions in libraries for WAGO-I/O-PRO and e!COCKPIT.

Further instructions on converting between CODESYS 2 and CODESYS 3 projects are provided in the CODESYS online help (“Converting”).

If a device of the imported project is unknown, a suggestion list of available and compatible devices is displayed.

---

**Note**

Applications may be discarded when using compatible devices!

If the compatible device does not support all existing applications, a warning message will be shown before you replace the device.

Unsupported applications are discarded.

Select a corresponding device.

7. To accept the device click [Replace device].

The main view is opened after the CODESYS project is loaded.

8. Open the “Programming” workspace.

Load the “Standard.lib” library first:

9. Double-click the library manager of the imported application.

10. Click [Add library] and select the “Standard” library.
11. Open your program/s and click [Build] in the “PROGRAM” tab.

12. If the message bar is opened, click [Display message window].

13. Adjust your program according to the messages.

14. Load missing libraries (see point 5) via the library manager. Both libraries for your devices are available at [http://www.wago.com](http://www.wago.com).
9.5.13 Importing/Exporting Program Elements

In order to import or export program elements, use the appropriate button [Import] or [Export] in the menu ribbon (“PROGRAM” tab). Alternatively access these functions via the Backstage view, “Import/Export” page > [Program Elements].

**Exporting**

1. Click the [Export] button.

The “Export” dialog window is opened and the Program Structure is displayed.

![Export dialog window](image)

Figure 228: Exporting Program Elements

2. Tick the elements to be exported.

3. Click [Export].

4. Select the memory location in the file selection dialog and enter a file name.

5. Export the program elements by clicking [OK].

A file with the suffix “*.export” is saved.

**Importing**

1. Click the [Import] button.

This opens a file selection dialog box.

2. Select the “*.export” file to be imported and click [Open].

The “Import” dialog is displayed.
The Program Structure is displayed in the left-hand area of the dialog. This is used to select the position where the imported elements are inserted.

The content of the import file is displayed in the right-hand area. The content is context-sensitive according to the selection in the Program Structure. If the content to be imported cannot be inserted at the current position in the Program Structure, the relevant objects are grayed out.

3. To insert the selected objects at the selected insert position click [Insert Elements].

The dialog does not close automatically. You can select a succession of elements and insert them at different positions. Multiple selection is also possible. This only allows the simultaneous selection of objects of the same class (e.g., programs). The selected objects are inserted at several positions at the same time.

4. Click [Close] to finish the import.

---

**Figure 229: Importing Program Elements**

The Program Structure is displayed in the left-hand area of the dialog. This is used to select the position where the imported elements are inserted.

The content of the import file is displayed in the right-hand area. The content is context-sensitive according to the selection in the Program Structure. If the content to be imported cannot be inserted at the current position in the Program Structure, the relevant objects are grayed out.

3. To insert the selected objects at the selected insert position click [Insert Elements].

The dialog does not close automatically. You can select a succession of elements and insert them at different positions. Multiple selection is also possible. This only allows the simultaneous selection of objects of the same class (e.g., programs). The selected objects are inserted at several positions at the same time.

4. Click [Close] to finish the import.
9.6 Setting and Managing

9.6.1 Documenting and Printing a Project

Overviews of all components contained in a project can be printed for project documentation.

1. Open Backstage view.
2. Click [Print].
3. Click [Document project...] and [Document].

A dialog will open, which displays the project components. Components which are printed are indicated with a green tick.

![Figure 230: Documenting a Project](image)

4. Untick components that are not to be printed as required.
5. Tick the appropriate checkboxes to create a **Title Page** or **Table of contents**.
6. Click [OK] to print the project.

An additional print function is provided for active editors in the “Programming” workspace.

1. Open the “Programming” workspace.
2. Open an element by clicking it in the Program Structure, such as a “PLC_PRG” program.
3. Move to the “START” tab.

4. Click [Print].

The content of the active editor is printed.

**9.6.2 Managing Devices**

To operate devices in *e!COCKPIT*, drivers containing device-specific description files are imported and interpreted by *e!COCKPIT*. These files contain information such as, device properties, interfaces and supported protocols.

The following device description files are supported:

- “Device Type Package” (DTP) for ETHERNET/MODBUS devices
- “Electronic Data Sheet” (EDS) for CANopen devices
- “Device master files” (GSD) for PROFIBUS devices
- “Device Configuration File” (DCF)

Device manufacturers provide these description files for their devices.

A device description file can apply to several similar devices. It is also possible to install several versions and use these in the same or different projects.

Devices and installed device description files are managed on the “Product Catalog” page of Backstage view.

1. To move to the Product Catalog management in the Backstage view, click the corresponding button in the Product Catalog panel. Alternatively click the “FILE” tab and the [Product Catalog] button.

![Figure 231: Moving to the Product Catalog Management](image)

**9.6.2.1 Adding Devices**

When *e!COCKPIT* is installed, description files are also installed for standard WAGO devices so they can be displayed and used directly in the Product Catalog.

If you require WAGO or other devices, proceed as follows to import the description files for these devices on the “Product Catalog” page in the Backstage view:

1. Click the [Add devices] button.

2. In the “Open” dialog select the file type and one or several device description files.
3. Click [Open].

Figure 232: Selecting a Device Description

This will import the description files. The devices are available in the Product Catalog.

9.6.2.2 Deleting Devices

Please note that when deleting a device description file for a particular device, it may be required for several similar devices. You may therefore be removing several devices from your project.

1. In the “Product Catalog” page in Backstage view select the device that you wish to remove.

Figure 233: Selecting a Device

Information on the relevant device description file and a list of devices that belong to this description file and version are displayed on the right.

2. Click the [Delete device...] button.

The “Included Devices” area highlights in yellow the devices related to this device description file that will also be deleted.

3. Click [Ok] to actually delete the device types.
4. If the currently opened project contains devices that are affected by this deletion, save and close the project before deletion and repeat the step.

9.6.2.3 Displaying Device Type Versions/Device Descriptions of a Device

A device may be present in several device type versions, i.e., may be represented in different versions of device description files. Any of these versions can be used in a project. Versions of device descriptions are displayed on the “Product Catalog” page in the Backstage view and in the Product Catalog.

1. To display available versions of a device, first select a device or module in the Network or Device Detail view.

2. Click the [Version information] button in the “DEVICE” tab in the menu ribbon.

9.6.2.4 Showing Device Type Versions/Device Descriptions of Several Devices

1. To display the available versions of several or all devices, enable the Display the Available Versions in the Product Catalog so that a green tick is displayed.
   This activation does not apply to all tabs of the Product Catalog, therefore switch the function on or off according to the tab.
Other levels are displayed with the available device type versions in the tree structure of the Product Catalog.

When using the Search function, only the versions of devices are displayed that are present in the Search result.

### 9.6.2.5 Replacing Device Type Versions/Device Descriptions

Devices are integrated in e!COCKPIT via device descriptions. Every device description has a version. Different versions can be installed for a device, although versions can also be replaced by others. When returning to a previous version, a check for any functional restrictions is made via the DTP description. Incompatible versions are not displayed. Replacing is not possible if the device is connected online.

1. Double-click the required version of the device description to operate the used device in this version, for example, to maintain compatibility with other devices.

Alternatively, select a device in the Network or Device Detail view and then click the [Replace] button.
2. In the selection field, choose the required version and click [Replace].

3. To replace the device descriptions of all Devices used in the project click [Replace all].

4. Replace the device description as described in section “Opening the Project with an Obsolete Device Description”.

Abbildung 239: Replace All Device Descriptions
9.6.2.6 Replacing Older Versions Automatically

1. Open the “Product Catalog” page in the Backstage view.

2. To search for new device descriptions when opening a project and to replace older versions automatically, tick the “Check projects on opening for new device descriptions” checkbox.

Older versions are only replaced if the new versions are compatible with the existing data and settings.

Figure 240: Replacing Older Versions Automatically
9.6.3 Performing Updates

Software updates can be started both automatically and manually on demand. These contain updates for e!COCKPIT itself, device description files for new WAGO devices and communication drivers.

1. Open Backstage view (“FILE” tab).
2. Click [Help] in the navigation bar.
3. Click [Updates].

The right-hand page shows information on installed components. When checking for updates, found updates and new components available are identified and also displayed in this area.

4. Use the “All,” “Updates,” “New” or “Installed” tabs to modify the display as required.

<table>
<thead>
<tr>
<th></th>
<th>Installed Version</th>
<th>Version Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>e!COCKPIT</td>
<td>0.1.0.475</td>
<td>n.a.</td>
</tr>
<tr>
<td>Device Descriptions:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>750Mod64</td>
<td>0.1.0.113</td>
<td>n.a.</td>
</tr>
<tr>
<td>750Mod632</td>
<td>0.1.0.113</td>
<td>n.a.</td>
</tr>
</tbody>
</table>

Figure 241: Performing Updates

Searching for updates automatically:

5. Tick the “Check for updates and license information on starting e!COCKPIT” checkbox (Internet connection required during program start).

Note

Automatic restart after update!
Updates are activated after the next restart. If you are installing e!COCKPIT updates or new communication drivers, the software is restarted automatically. A restart is not required for importing new device descriptions.
Searching for updates manually:

6. To search for updates manually, click the [Search for Updates] button. The updates found are then displayed in the categories e!COCKPIT, device descriptions and communication drivers, and highlighted in color. All updates are selected by default.

7. If required, exclude the individual updates by clicking on the relevant checkboxes and deactivating these updates.

The update installation process checks your license information with WAGO Kontakttechnik GmbH & Co. KG.

8. Agree to this check by activating the “I accept the WAGO SOFTWARE LICENSE AGREEMENT” checkbox.

9. To update all the selected components click [Install updates].
9.6.4 Archiving

Saves the current project with all its associated data, such as devices used, device descriptions, libraries on the controller as an archive file. The archive combines project-related files so that they can be forwarded to third parties. An archive combines all the files contained and referenced in the currently opened project into a central "*.eca" archive file. The "Archive" function can be selected when a project is active.

9.6.4.1 Creating an Archive

1. Open Backstage view ("FILE" tab).
2. Click [Import/Export].
3. Click [e!COCKPIT archive].
4. Click the [Save archive] button.

A dialog will open, which displays the elements of the currently opened project. The archiving always includes all devices and applications. Libraries can be excluded from archiving by unticking the checkbox for the appropriate library.

![Figure 242: Saving Archives](image)

5. Click [Save].
6. Select a location for the archive.
7. Click [Save] to finish the archiving.
9.6.4.2 Opening an Existing Archive

1. Open Backstage view (“FILE” tab).
2. Click [Import/Export].
3. Click [e!COCKPIT archive].
4. Click the [Open archive] button.
5. Select a saved “.eca” archive file.
6. Click [Open].

The archive file is extracted. Libraries are installed and loaded. The project is opened.
9.6.5 Deinstalling

Back up your project files before deinstalling in case you wish to reinstall e!COCKPIT in future. For this use the archive function (see Section “Operation” > “Setting and Managing” > “Archiving”).

1. Call up the Control Panel.
2. Click [Uninstall Programs] (such as with Windows 7).
3. Select “WAGO e!COCKPIT”.
4. Click [Uninstall].

The software with all integrated components is removed from your PC. Separately installed CODESYS versions are not affected by this.
9.7 Interfaces

9.7.1 smartDESIGNER

The smartDESIGNER from WAGO Kontakttechnik GmbH & Co. KG is an online configurator which enables terminal blocks, I/O modules, multi-terminal connectors, PCB terminal blocks and WINSTA® cables to be designed and projects to be validated. Labels can be created and printed for the components used. The graphical display of components on a DIN rail is the same as in e!COCKPIT. The smartDESIGNER can also be used to generate mounting plans, parts lists and 3D views. The tool features several CAE interfaces including EPLAN, a system for designing electrical installations. This interface supports export/import to/from e!COCKPIT, so that modifications made in one system can be transferred to the other.

Figure 243: WAGO smartDESIGNER
9.7.1.1 Importing a Project from smartDESIGNER

1. To import projects created with smartDESIGNER, click [Import/Export] in the Backstage view.
2. Click [smartDESIGNER].
3. Click [Import].
4. Select a valid file (CAE file, *.xml) from the dialog that opens.

The content of the file is opened as a new project. Network view is opened automatically.

If a project was opened and modified beforehand, you will be asked to save it.

The “Import” dialog is opened. This contains all imported (green) and unimported (red) devices.

![Import](image)

Figure 244: Overview of the Imported Devices

9.7.1.2 Exporting a Project for smartDESIGNER

1. To export projects you wish to open in smartDESIGNER, click [Import/Export] in Backstage view.
2. Click [smartDESIGNER].
3. Click [Export].

Select a valid file in the dialog.

All devices with modules are created in the export file.

The Backstage view stays open.

A warning message appears if the project contains a device for which a DTP is not installed. If this is a coupler, the modules are exported on a DIN rail without this coupler.

A warning message also appears if the project contains a device that does not have a valid order number, as this is possibly a generic item.
List of Figures

Figure 1: Display of the Start Screen during Loading ........................................... 18
Figure 2: Display of the License Status ................................................................. 18
Figure 3: Entering the License Key ................................................................. 19
Figure 4: Entering Licenses ....................................................................... 20
Figure 5: Activating Automatic Updates ............................................................... 21
Figure 6: Start View .............................................................................................. 22
Figure 7: Main View ............................................................................................. 24
Figure 8: Quick Access Toolbar (Standard View) ................................................ 25
Figure 9: Customizing the Quick Access Toolbar ............................................. 25
Figure 10: Menu Ribbon (Standard View) ............................................................ 26
Figure 11: Managing workspaces ......................................................................... 27
Figure 12: Creating a New Workspace .................................................................. 28
Figure 13: “VIEW” Tab ........................................................................................ 30
Figure 14: “NETWORK” Tab ........................................................................... 32
Figure 15: “DEVICE” Tab ................................................................................... 34
Figure 16: “Multiple download” Dialog ............................................................. 35
Figure 17: “PROGRAM” Tab ............................................................................ 36
Figure 18: Multiple Download ........................................................................... 38
Figure 19: “Search” Dialog ................................................................................... 39
Figure 20: “DEBUG” Tab ..................................................................................... 42
Figure 21: “I/O CHECK” Tab ............................................................................... 46
Figure 22: Views and Workspaces ........................................................................ 49
Figure 23: General Status Display for Connections .............................................. 50
Figure 24: Status Display for Devices (Online) .................................................... 52
Figure 25: “Network/Devices” Workspace ........................................................... 53
Figure 26: Network View ...................................................................................... 54
Figure 27: Context Menu of a Device ................................................................... 55
Figure 28: Color Coding of Connectors ............................................................... 57
Figure 29: Context Menu of a Connector .............................................................. 57
Figure 30: Connection between Two Devices ...................................................... 59
Figure 31: Fieldbus-Specific Configurators .......................................................... 60
Figure 32: “MODBUS Master Variables” Tab ..................................................... 61
Figure 33: Slave Variables Context Menu ............................................................ 62
Figure 34: Master Variables Context Menu .......................................................... 63
Figure 35: “Connection Settings” Tab .................................................................. 63
Figure 36: MODBUS Slave Configurator (view with “Expert Settings” checkbox enabled) ....................................................................................... 66
Figure 37: “CANopen Remote Device” Tab ....................................................... 66
Figure 38: SD Channel Configuration .................................................................. 67
Figure 39: Heartbeat Consuming Properties ..................................................... 68
Figure 40: “PDO Mapping Tab” .......................................................................... 69
Figure 41: “Service Data Object Tab” ................................................................. 70
Figure 42: Adding Service Data Objects ............................................................. 71
Figure 43: “CANopen I/O Mapping” Tab ............................................................. 72
Figure 44: CANopen Slave Configuration ............................................................ 73
Figure 45: Edit I/O Area ........................................................................................ 73
Figure 46: Add I/O Range ..................................................................................... 74
Figure 47: Edit Parameter Range .......................................................................... 75
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>CANbus Slave I/O Mapping</td>
<td>76</td>
</tr>
<tr>
<td>49</td>
<td>CANopen Manager</td>
<td>77</td>
</tr>
<tr>
<td>50</td>
<td>Check and Fix Configuration</td>
<td>77</td>
</tr>
<tr>
<td>51</td>
<td>CANopen I/O Mapping</td>
<td>79</td>
</tr>
<tr>
<td>52</td>
<td>PROFIBUS Connection Configurator</td>
<td>80</td>
</tr>
<tr>
<td>53</td>
<td>Context Menu of a Data Point</td>
<td>82</td>
</tr>
<tr>
<td>54</td>
<td>Information in the Status Line of the PROFIBUS Configurator</td>
<td>82</td>
</tr>
<tr>
<td>55</td>
<td>PROFIBUS Connection Configurator</td>
<td>82</td>
</tr>
<tr>
<td>56</td>
<td>Device View (Example: Fieldbus Controller with Connected Modules)</td>
<td>84</td>
</tr>
<tr>
<td>57</td>
<td>I/O Module Context Menu</td>
<td>85</td>
</tr>
<tr>
<td>58</td>
<td>I/O Mapping</td>
<td>87</td>
</tr>
<tr>
<td>59</td>
<td>Elements of the Program Editor</td>
<td>90</td>
</tr>
<tr>
<td>60</td>
<td>Library Manager</td>
<td>92</td>
</tr>
<tr>
<td>61</td>
<td>Task Configuration</td>
<td>93</td>
</tr>
<tr>
<td>62</td>
<td>“PLC_Task” Main Task</td>
<td>94</td>
</tr>
<tr>
<td>63</td>
<td>Selecting a Program via the Input Assistant</td>
<td>96</td>
</tr>
<tr>
<td>64</td>
<td>Breakpoints</td>
<td>97</td>
</tr>
<tr>
<td>65</td>
<td>Call Stack</td>
<td>97</td>
</tr>
<tr>
<td>66</td>
<td>Cross Reference List</td>
<td>98</td>
</tr>
<tr>
<td>67</td>
<td>Creating and Managing User-Defined Workspaces</td>
<td>99</td>
</tr>
<tr>
<td>68</td>
<td>Panels</td>
<td>100</td>
</tr>
<tr>
<td>69</td>
<td>Displaying Panels</td>
<td>100</td>
</tr>
<tr>
<td>70</td>
<td>“Device Structure” Panel</td>
<td>101</td>
</tr>
<tr>
<td>71</td>
<td>“Program Structure” Panel</td>
<td>102</td>
</tr>
<tr>
<td>72</td>
<td>Context Menu of an Application</td>
<td>103</td>
</tr>
<tr>
<td>73</td>
<td>“Product Catalog” Panel</td>
<td>105</td>
</tr>
<tr>
<td>74</td>
<td>Displaying Available Versions</td>
<td>106</td>
</tr>
<tr>
<td>75</td>
<td>Dragging Devices to the Network View</td>
<td>106</td>
</tr>
<tr>
<td>76</td>
<td>Displaying Product Information</td>
<td>106</td>
</tr>
<tr>
<td>77</td>
<td>Adding a Device to a Project</td>
<td>107</td>
</tr>
<tr>
<td>78</td>
<td>“Settings” Panel with the Button to a Device Configurator (CANopen)</td>
<td>108</td>
</tr>
<tr>
<td>79</td>
<td>Additional Plc Settings</td>
<td>109</td>
</tr>
<tr>
<td>80</td>
<td>Message Bar</td>
<td>110</td>
</tr>
<tr>
<td>81</td>
<td>Message Window</td>
<td>110</td>
</tr>
<tr>
<td>82</td>
<td>Status Bar</td>
<td>111</td>
</tr>
<tr>
<td>83</td>
<td>Backstage View</td>
<td>112</td>
</tr>
<tr>
<td>84</td>
<td>Backstage View, “New” Page</td>
<td>113</td>
</tr>
<tr>
<td>85</td>
<td>Backstage View, “Open” Page</td>
<td>114</td>
</tr>
<tr>
<td>86</td>
<td>Backstage View, “Save As” Page</td>
<td>115</td>
</tr>
<tr>
<td>87</td>
<td>Backstage View, “Print” Page</td>
<td>116</td>
</tr>
<tr>
<td>88</td>
<td>Backstage View, “Import/Export” Page</td>
<td>117</td>
</tr>
<tr>
<td>89</td>
<td>Backstage View, “Project Settings” Page</td>
<td>119</td>
</tr>
<tr>
<td>90</td>
<td>Backstage View, “Product Catalog” Page</td>
<td>120</td>
</tr>
<tr>
<td>91</td>
<td>Backstage View, “Options” Page</td>
<td>122</td>
</tr>
<tr>
<td>92</td>
<td>Backstage View, “Help” Page</td>
<td>124</td>
</tr>
<tr>
<td>93</td>
<td>Using Online Help</td>
<td>126</td>
</tr>
<tr>
<td>94</td>
<td>Displaying Panels</td>
<td>128</td>
</tr>
<tr>
<td>95</td>
<td>Closing a Panel</td>
<td>128</td>
</tr>
</tbody>
</table>
Figure 96: Docking Panels ................................................................. 129
Figure 97: Grouping Panels ............................................................. 129
Figure 98: Docking and Floating the Main View ................................ 130
Figure 99: Multiple Selection in Tree Structures or Network View .... 131
Figure 100: Multiple Selection with Frames ...................................... 131
Figure 101: Moving Multiple Selection ............................................. 132
Figure 102: Predefined Workspaces .................................................. 133
Figure 103: Opening the “Network/Devices” Workspace ................. 133
Figure 104: Opening the “Network/Devices” Workspace ................... 133
Figure 105: Opening Network View .................................................. 134
Figure 106: “NETWORK” Tab ........................................................... 134
Figure 107: Opening Device Detail View via Network View .............. 134
Figure 108: Opening Device Detail View via the “DEVICE” Tab ......... 134
Figure 109: Opening Device Detail View .......................................... 135
Figure 110: “DEVICE” Tab .............................................................. 135
Figure 111: Use the Breadcrumb navigation to move to the Network view 135
Figure 112: Use the “NETWORK” tab to move to the Network view .... 135
Figure 113: Moving between Network View and Device Detail View ..... 136
Figure 114: Toggling between Graphical and Tabular View .............. 136
Figure 115: Opening the “Programming” Workspace ....................... 136
Figure 116: Opening the “Programming” Workspace ....................... 136
Figure 117: Opening the “Programming” Workspace ....................... 137
Figure 118: Opening the “PROGRAM” and “DEBUG” Tabs ............... 137
Figure 119: Creating a New Workspace .......................................... 137
Figure 120: New Button in the Menu Ribbon ................................... 138
Figure 121: Selecting a Template ...................................................... 142
Figure 122: Docking the Position of the Project File ......................... 143
Figure 123: Replacing Device Descriptions ..................................... 143
Figure 124: Replacing All Device Descriptions ................................. 144
Figure 125: Saving a Project Automatically ....................................... 145
Figure 126: Automatically Saved Projects on the “Open” Page ........ 146
Figure 127: Selecting the Medium for Scanning ............................... 149
Figure 128: Restricting the Scan Range ............................................ 149
Figure 129: Starting the Scan Operation .......................................... 149
Figure 130: Displaying a Scan Result .............................................. 150
Figure 131: Opening Graphical Network View .................................. 150
Figure 132: Scanning Modules ....................................................... 151
Figure 133: Displaying Scanned Modules ........................................ 151
Figure 134: Selecting Modules ......................................................... 151
Figure 135: Accepting Modules with the Same Scan Result Together ... 152
Figure 136: Applying Modules ........................................................ 152
Figure 137: Displaying Scanned Modules ........................................ 152
Figure 138: Display of Applied Modules ......................................... 153
Figure 139: Scanning for Modules of Several Devices ..................... 153
Figure 140: Displaying the Scan Report .......................................... 154
Figure 141: Connecting Devices (Network View) ............................. 154
Figure 142: Connecting Devices (Device Structure) ......................... 154
Figure 143: Connecting Devices (Menu Ribbon) ............................. 155
Figure 144: Opening the “Settings” Panel (here: Device Detail View) .......... 155
Figure 145: “Settings” Panel (Tab Navigation) ...................................... 156
Figure 146: “Settings” Panel (Groups in Tabs) ...................................... 156
Figure 147: Opening Device-Specific Configurators ............................... 157
Figure 148: Writing Parameters to the Device ...................................... 158
Figure 149: Reading Parameters from the Device ................................. 158
Figure 150: Dragging Devices to the Network View ............................... 159
Figure 151: Opening the “Settings” Panel ............................................ 159
Figure 152: Establishing Network Connections .................................... 160
Figure 153: Opening the Interface ..................................................... 160
Figure 154: Selecting a Connection ..................................................... 161
Figure 155: Displaying Devices That Are Logically Connected .............. 161
Figure 156: Opening Connection Settings or MODBUS Slave Configurator 162
Figure 157: Variable Configuration MODBUS, View for the Master (Client) 162
Figure 158: Creating New Variables on the Slave ................................. 163
Figure 159: New Variable in the Slave Application ............................... 163
Figure 160: MODBUS Connection Settings ....................................... 164
Figure 161: New Variable on the Master ............................................. 164
Figure 162: Opening the Main Program of the Master ............................ 164
Figure 163: Accessing the “newVar” Variable in the Master Application ... 165
Figure 164: Dragging Devices to the Network View ............................. 166
Figure 165: Assigning an IP Address .................................................. 166
Figure 166: Opening the “Settings” Panel ............................................ 167
Figure 167: Assigning the Role of the Device ....................................... 167
Figure 168: Creating a Connection between CANopen Devices .............. 167
Figure 169: Adapting “Node ID” and Baud Rate .................................. 168
Figure 170: Opening the CANopen Connection Configurator .................. 168
Figure 171: CANopen Connection Configurator (Slave) ......................... 168
Figure 172: Adding a Data Point .......................................................... 169
Figure 173: Updating the Configuration .............................................. 169
Figure 174: Choosing the Update Behavior ......................................... 170
Figure 175: Opening the CANopen Connection Configurator ................. 170
Figure 176: Entering Variables ............................................................ 170
Figure 177: Placing the PROFIBUS Master and Slave in the Network View ... 171
Figure 178: Connecting PROFIBUS Devices ....................................... 171
Figure 179: Opening the “Settings” Panel ............................................ 172
Figure 180: Opening the Configurator of the PROFIBUS Connection ....... 172
Figure 181: PROFIBUS Connection Configurator .................................. 173
Figure 182: Creating a New Variable .................................................. 173
Figure 183: Configured Variables ...................................................... 174
Figure 184: Creating Multiple Variables ............................................. 174
Figure 185: Using Created Variables .................................................. 175
Figure 186: Opening the “I/O Mapping” Tab ....................................... 175
Figure 187: Manual Entry of the IEC Variable .................................... 175
Figure 188: Entering IEC Variables via the Input Assistant .................... 175
Figure 189: Selecting a Variable ........................................................ 176
Figure 190: Mapping Completed ....................................................... 176
Figure 191: Saving the Device Description File .................................... 176
Figure 192: Switching an LED On/Off with the Mouse ......................... 177
Figure 193: Display of an I/O Module for the Input/Output Test .............. 178
Figure 194: Display of a Node for the Input/Output Test ................................... 178
Figure 195: Performing a Test ............................................................................. 179
Figure 196: Status Display for Inputs and Outputs ............................................. 179
Figure 197: Changing Analog and Digital Values .............................................. 179
Figure 198: Write Collection Values ................................................................... 180
Figure 199: Opening the “Programming” Workspace ........................................... 181
Figure 200: Context Menu in the Program Structure ........................................... 182
Figure 201: Adding a POU .................................................................................. 183
Figure 202: Selecting a Template for Libraries..................................................... 184
Figure 203: Creating a Library ............................................................................ 184
Figure 204: Saving and Installing a Library ......................................................... 185
Figure 205: Adding the Library Manager ............................................................. 186
Figure 206: Creating a Program .......................................................................... 187
Figure 207: Adding a Program Call ..................................................................... 187
Figure 208: Entering the Program Name via the Input Assistant ......................... 188
Figure 209: Programming Editor (in this example: Structured Text (ST))............. 188
Figure 210: Entering Variables/Data Points ......................................................... 188
Figure 211: Entering Variables/Data Points via the Input Assistant ..................... 189
Figure 212: I/O Mapping ..................................................................................... 189
Figure 213: Compiling an Application .................................................................. 190
Figure 214: Message Window ............................................................................. 190
Figure 215: Excluding the Program from Compiling ........................................... 191
Figure 216: Program Excluded from Compiling .................................................. 191
Figure 217: Selecting Values for Monitoring ....................................................... 192
Figure 218: Watch List ....................................................................................... 192
Figure 219: Adding a Visualization ..................................................................... 195
Figure 220: Creating a Visualization .................................................................. 195
Figure 221: Assigning a Value ........................................................................... 196
Figure 222: Multiple Download ......................................................................... 198
Figure 223: Loading a CODESYS Project ........................................................... 199
Figure 224: Converting a Library Reference ....................................................... 199
Figure 225: Adding a Library ............................................................................. 201
Figure 226: Exporting Program Elements ............................................................ 202
Figure 227: Importing Program Elements ............................................................. 203
Figure 228: Documenting a Project ..................................................................... 204
Figure 229: Moving to the Product Catalog Management .................................... 205
Figure 230: Selecting a Device Description ........................................................... 206
Figure 231: Selecting a Device ........................................................................... 206
Figure 232: Deleting a Device Type ..................................................................... 207
Figure 233: Displaying Device Type Versions/Device Descriptions ..................... 207
Figure 234: Displaying Available Versions ........................................................... 208
Figure 235: Replacing Device Type Versions/Device Descriptions ..................... 208
Figure 236: Replacing Device Type Versions/Device Descriptions ..................... 209
Abbildung 239: Replace All Device Descriptions .............................................. 209
Figure 237: Replacing Older Versions Automatically ........................................... 210
Figure 238: Performing Updates ........................................................................ 211
Figure 239: Saving Archives .............................................................................. 213
Figure 240: WAGO smartDESIGNER ............................................................... 216
Figure 241: Overview of the Imported Devices ..................................................... 217
List of Tables

Table 1: Number Notation ................................................................. 10
Table 2: Font Conventions .............................................................. 10
Table 3: System Requirements ......................................................... 15
Table 4: Key for Figure “Start View” ................................................ 23
Table 5: Key for Figure “Main View” ................................................ 24
Table 6: Buttons on the Quick Access Toolbar ................................. 25
Table 7: “START” Tab Operating Elements ........................................ 26
Table 8: “VIEW” Tab Operating Elements .......................................... 30
Table 9: “NETWORK” Tab Operating Elements ................................... 32
Table 10: “DEVICE” Tab Operating Elements .................................... 34
Table 11: “PROGRAM” Tab Operating Elements ................................. 36
Table 12: “DEBUG” Operating Elements .......................................... 42
Table 13: “I/O CHECK” Tab Operating Elements ............................... 46
Table 14: General Operating Elements .............................................. 50
Table 15: Key for Figure “General Status Display for Connections” ....... 50
Table 16: General Status Display for Connections ............................... 51
Table 17: Elements of the Network View ........................................... 53
Table 18: Key for Figure “Network View” ......................................... 54
Table 19: Device Status ................................................................. 55
Table 20: Context Menu of a Device .................................................. 56
Table 21: Color Coding of Connectors .............................................. 57
Table 22: Operating Elements in the Context Menu of a Connector ....... 58
Table 23: Key for Figure “MODBUS Master Variables” Tab ................ 62
Table 24: Key to Figure “Slave Variables Context Menu” .................... 63
Table 25: Key to Figure “Master Variables Context Menu” ................. 63
Table 26: Key to “Connection Settings” Tab ....................................... 64
Table 27: Key for Figure “CANopen Remote Device Tab” .................... 67
Table 28: Key for Figure “PDO Mapping Tab” .................................... 70
Table 29: Key to Figure “Service Data Object Tab” ............................. 70
Table 30: Key to Figure “CANopen I/O Mapping” Tab ....................... 72
Table 31: General Configuration ..................................................... 73
Table 32: CANbus I/O Mapping ...................................................... 76
Table 33: CANopen Remote Device .................................................. 77
Table 34: PROFIBUS Connection Configurator ................................. 81
Table 35: PROFIBUS Configurator ................................................... 83
Table 36: Elements of the Device View ............................................. 85
Table 37: Context Menu of a Device .................................................. 86
Table 38: I/O Mapping ................................................................. 87
Table 39: Elements of the Program Editor ........................................ 90
Table 40: Library Manager ............................................................. 92
Table 41: “PLC_Task” Main Task ..................................................... 94
Table 42: Context Menu of an Application ........................................ 104
Table 43: Buttons of the “Product Catalog” Panel ............................... 105
Table 44: Message Bar Elements .................................................... 110
Table 45: Elements of the Message Window ...................................... 110
Table 46: Legend for Figure Backstage View ..................................... 112
Table 47: Backstage View Operating Elements, “Open” Page ............... 114
Table 48: Backstage View Operating Elements, “Print” Page .............. 116
Table 49: Backstage View Operating Elements, “Import/Export” Page ............ 118
Table 50: Backstage View Operating Elements, “Project Settings” Page ...... 119
Table 51: Backstage View Operating Elements, “Product Catalog” Page ...... 120
Table 52: Backstage View Operating Elements, “Options” Page ............... 122
Table 53: Backstage View Operating Elements, “Help” Page ................... 124
Table 54: General Key Combinations .................................................... 140
Table 55: Key Combinations, “START” Tab ......................................... 140
Table 56: Key Combinations, “PROGRAM” Tab .................................... 140
Table 57: Key Combinations, “PROGRAM” Tab .................................... 141