ELOP II

New Features

ELOP II Version 5.1

SAFETY NONSTOP

All HIMA products mentioned in this manual are protected by the HIMA trade-mark. Unless noted otherwise, this also applies to other manufacturers and their respective products referred to herein.

All of the instructions and technical specifications in this manual have been written with great care and effective quality assurance measures have been implemented to ensure their validity. For questions, please contact HIMA directly. HIMA appreciates any suggestion on which information should be included in the manual.

Equipment subject to change without notice. HIMA also reserves the right to modify the written material without prior notice.

For further information, refer to the HIMA DVD and our website http://www.hima.de and http://www.hima.com.

Contact

HIMA Address

HIMA Paul Hildebrandt GmbH

P.O. Box 1261

68777 Brühl, Germany

Tel: +49 6202 709-0

Fax: +49 6202 709-107

E-mail: info@hima.com

Revision index	Revisions	Type of Change	
		technical	editorial
3.00	Revised for ELOP II version 5.1	X	X
3.01	Changed: chapters 2.1.3, 2.3.1.2, and 7.2	X	
3.02	Changed: chapter 7.6		X

Table of Contents

CHAPTER 1

Introduction

Information for Updating
ELOP II from Version 4.1 to 5.1

Object-oriented

Programming system

based on IEC 61131-3

for the HIQuad system family

Operating systems: Windows XP, Windows 7

1 Introduction

1.1 About ELOP II Version 5.1

This document presents the new features of ELOP II version 5.1. The screenshots and all information should help you to become rapidly familiar with all additional features of the new version.
The document provides an overview of the new features. Detailed information is provided in the online help and in the ReadMe files READ1ST.HTM and README.HTM.

Note READ1ST.HTM and all others ReadMe files (PROBLEMS, WHATSNEW, HISTORY, README) can be found in the ELOP II Control Center: Look for the required file under *Documentation*.
An electronic version of this manual is also contained in the Control Center: Click *New Features*.

---

i Refer to the online help for more details about already available features .

1.2 Spelling

Character format	*Usage for*
Italics:	Cross references, references, labeling, placeholders
SMALL CAPS	Keyboard shortcut If multiple keys are pressed simultaneously, they are separated from one another by a plus sign (+). ***Example:*** ALT+S means that the ALT-key and the S-Key are pressed simultaneously.
()	They are used to include one or multiple parameters when involking a function.
`Source`	Commands, options, parameters, source examples

`BEGIN`
`.`
`.`
`.`
`END`
	Blank lines in source examples

...	The following parameters with identical spelling
[]	They identify optional parameters ***Example:*** `COMPILE [-Option1] [-Option2] PROJECT`
|	The parameter located before or after this character must be entered.

1.3 How to Activate the Current Version in the License

Activating the Version via Signature

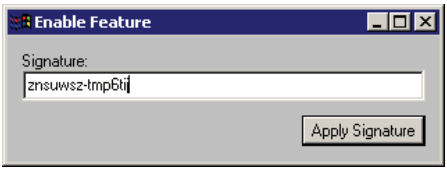
Starting with version 3.4, each new version included in a valid license must be activated!
The version is activated using a signature or **activation key** (a character-digit combination) which is provided by HIMA.

In case of a new delivery or license extension, HIMA provides a dongle already enabled for the corresponding version.

When upgrading to a newer version, proceed as follows to activate the required version:

1. Install the ELOP II base system.
2. Attach the dongle to the computer.
3. Have the signature (activation key) for the new version ready.

4. Start the Project Management.
5. A message appears informing that the current dongle is not enabled for the required version.
6. Click *OK* to confirm this message.
7. A dialog box opens. Enter the signature:



8. Click Apply Signature.
9. A message appears informing that the activation was successful. Confirm the message.
10. Click  to close the dialog box.
11. The Project Management is started.

Note Use this dialog box to directly enable additional products from within the Project Management:

1. Start the Project Management.
2. Open the *Tools* menu on the menu bar.
3. Select Enable Feature.
4. The *Enable Feature* dialog box appears: Proceed as described for the remaining actions.

Additional info... Additional information is provided in the online help (Search for "Enabling").

1.4 Installation with User Privileges is Possible

Upon completion of the installation process, ELOP II version 5.1 can be used with all Windows user accounts of the current workstation. User privileges of group Users are sufficient for operating ELOP II.

Restriction for operating different ELOP II versions on one workstation: Power user privileges (applies to Windows XP) or administrator privileges (applies to Windows 7) are required. If the policies of your company set user privileges of group "Users" as default and different ELOP II version must be operated, you must register the product

components of the specific ELOP II version each time before this ELOP II version is started. Ask your administrator to support you.

1.5 New: Windows 7 Supported

With version 5.1, ELOP II can be installed under the following operating systems:

- Windows XP Professional (service pack 2 or higher)
- New: Windows 7 Professional (32-bit)
- New: Windows 7 Ultimate (32-bit).

Note Applying to all supported operating systems:
To ensure that ELOP II operates error-free and safely, HIMA recommends installing the most recent service packs and security updates (available via Windows Update).

CHAPTER 2

Conversion

"What happens with my old data?"
This is a question that users frequently ask themselves when using new versions.

User-defined data can be used in the new version, but it must be converted first.
This chapter describes the required steps.

Observe these instructions under all circumstances!

2 Conversion

This chapter describes how to convert an unchanged project. If the project has been modified, proceed as described in the Safety Manual (HI 800 013 E).

Note Consequences that might endanger the system's safe operation depend on the application and must be agreed upon together with the test authority responsible for the final inspection.

2.1 Overview: Using Data from Previous Versions

2.1.1 Conversion from V3.0 to V3.5

A project created with ELOP II V3.0 must be first converted from ELOP II V3.0 to ELOP II V3.5. The conversion leads to a new code version.
In such a case, observe the instructions specified in the manual *Instructions for conversion of user programs from ELOP II-NT V 3.0 to ELOP II V 3.5* (HI 800 107 E).

2.1.2 Conversion from V3.5 to V4.1

A project created with ELOP II V3.5 must be first converted from ELOP II V3.5 to ELOP II V4.1. The conversion may lead to a new code version if specific features (e.g., step chains) have been used in V3.5.
In such a case, observe the instructions specified in the manual *Conversion of Projects from ELOP II V3.5 to ELOP II V4.1* (HI 800 317 E).

2.1.3 Conversion from V4.1 to V5.1

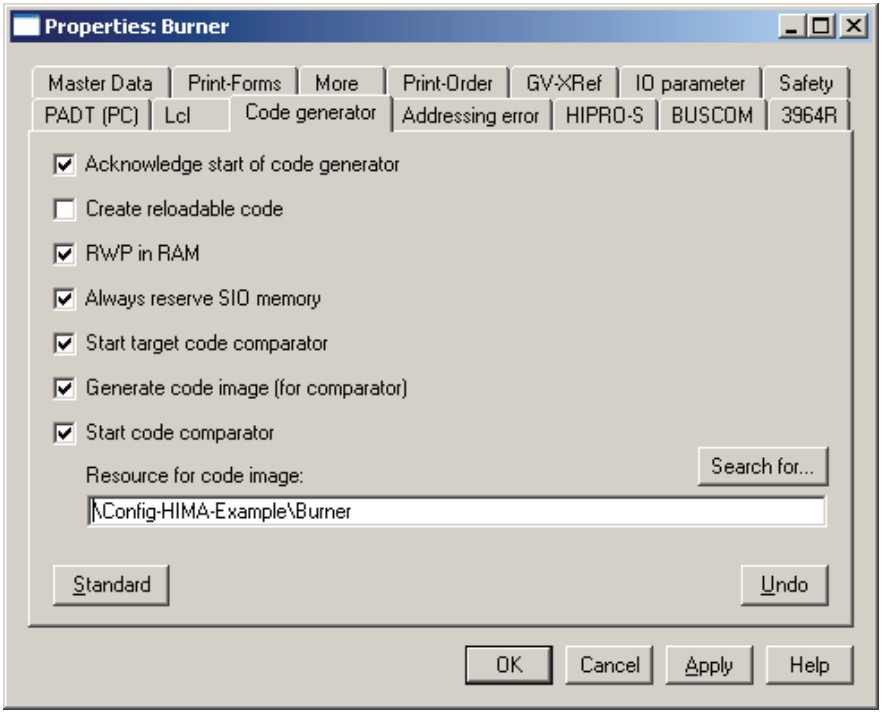
A project created with ELOP II V4.1 can be directly converted to ELOP II V5.1. The conversion of a project from ELOP II V4.1 to ELOP II V5.1 does not result in a changed CRC.

2.2 Processing Steps in ELOP II V4.1

Perform the following steps to prepare the conversion of an ELOP II V4.1 project.

1 In ELOP II V4.1, open the project to be converted.

2. Follow the steps below for each resource contained in the project
   - Right-click the resource and select *Properties* from the context menu.
   - Select the *code generator* tab.
   - Activate the following settings:
     - Activate *Process to object code comparison*
       The object code comparator compares 2 object codes generated successively by the C-compiler (GCC). This action prevents errors that might be due to an unsafe standard PC.
     - Activate *Generate code comparator image*
     - Activate *Perform code comparison*
       The C-Code comparator is used to identify the changes performed in the user program. Click **Browse**, for the code comparator image (C-Code (old)), specify the same resource used in the current project.
   - Click **Apply** to confirm the settings.



Additional Info... Before the C-code comparator is activated in the resource properties, a C-code must be available for each resource. Generate the code if no C-code is available for the current resource.

3. Right-click the resource and start the code generator using the context menu.
   If the option *Generate Reloadable Code* is set in the *Code Generator* tab, the
   reloadable code must be generated for the current resource such as specified in the Operating System Manual (HI 800 105 E).
4. When all resources have been processed, close the project selecting Project->Close on the menu bar.
5 Archive the project in ELOP II V4.1 prior to converting it to the required version.

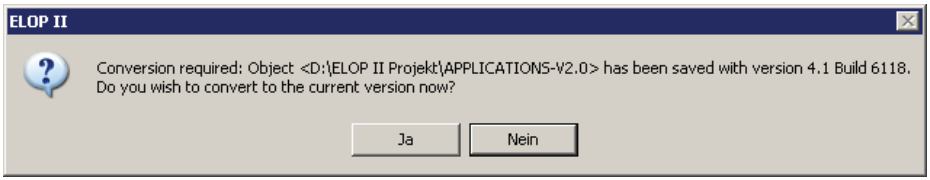
Additional Info... The conversion of a project from ELOP II V4.1 to ELOP II V5.1 cannot be reversed! Archive the project in ELOP II V4.1 prior to converting it to the required version.

6. Exit ELOP II V4.1.

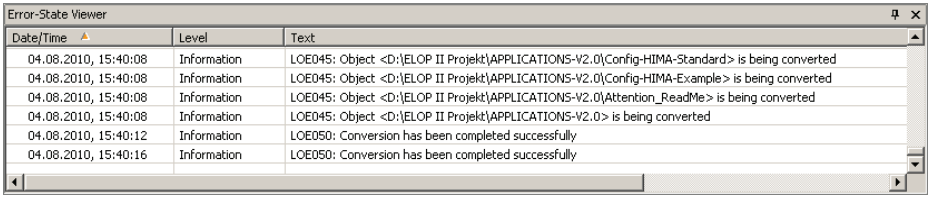
2.3 Processing Steps in ELOP II V5.1

Perform the following steps to open the project prepared in ELOP II V4.1 and convert it to ELOP II V5.1:

1. Start ELOP II V5.1.
2. Open the prepared project .
3. Click **Yes** to confirm the dialog box.



4. On the menu bar, select *Tools->Status Viewer* and check whether the conversion was successfully completed.
   If the conversion was not successful, convert the project with LCConVer, see Chapter 2.3.2.

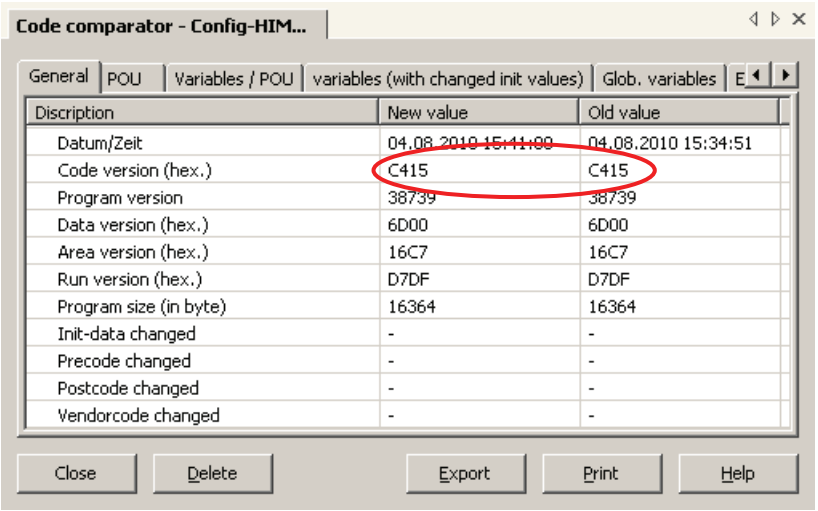


2.3.1 Comparing the Code Versions

Follow the steps below for <u>each</u> resource contained in the project

1. Select a resource and start the code generator.During the code generation, the *Code Comparison* dialog box opens to display the result of the code comparison.

2. Check in the *Code Comparison* dialog box whether the code version (hex.) changed.



2.3.1.1 The Code Version did not Change

If the code version did not change, no change test is required.

2.3.1.2 The Code Version Changed

If the code version has changed, contact the HIMA support.

2.3.2 Checking the Object-Integrity when Updating to Newer Versions

In rare cases, errors in ELOP II might occur due to unpredictable reasons. Potential inconsistencies can result in the object data which would only be recognized when visually controlling the object (e.g., opening the POU in the POU editor) during the code generation or in the online operation.

The LCCheckObj tool or the *Check Object Data* command (context menu of all ELOP II objects) can be used to check the integrity of the ELOP II objects and detect such kind of errors at a previous stage.

HIMA recommends checking the object integrity whenever an update to a new version is performed. Refer to the online help for details (e.g., searching for "Integrity, Checking Object-Integrity" in the index).

2.4 Conversion with LCConvVer

If a critical error is contained in an ELOP II V4.1 project (e.g., a broken connector), the conversion process is aborted. Nevertheless, the project can be converted to V5.1 by performing the following steps:

1. Prepare the ELOP II V4.1 project following the instructions specified in Chapter 2.1.1 for conversion to ELOP II V5.1.
2. Open the *ELOP II Control Center*.
3. Open the *ELOP II Command Prompt*.
4. In the command prompt, enter the command
   LCConvVer –f –r –d –x [path/project] >Info.txt.
   The option **-x** removes critical errors occurred during the conversion (e.g., potential inconsistencies in the object data). For more information on the *LCConvVer* conversion tool, refer to the ELOP II Control Center (select *Documentation->Tool Documentation* or in the online help (e.g., searching the term "Conversion, Repair Option").

   If the ELOP II V4.1 project was successfully converted to ELOP II V5.1, the following message appears in the command prompt:
   `Information: LOE050: The conversion was successfully completed`
5. Perform the following steps to check the removed errors:
   - Open the *Info.txt file* and look for error messages.
     The Info.txt file contains all messages (including error or status messages) that were created during the conversion to ELOP II V5.1.

Additional info...

All functions in which errors (e.g., broken connectors) have been removed, must be verified in the user program.

- Compare the code versions and verify the changes such as described in Chapter 2.3.1.

CHAPTER 3

Using the User Interface Efficiently

The better a program's user interface meets the user's individual requirements, the more efficiently and productively the user can work with that program.

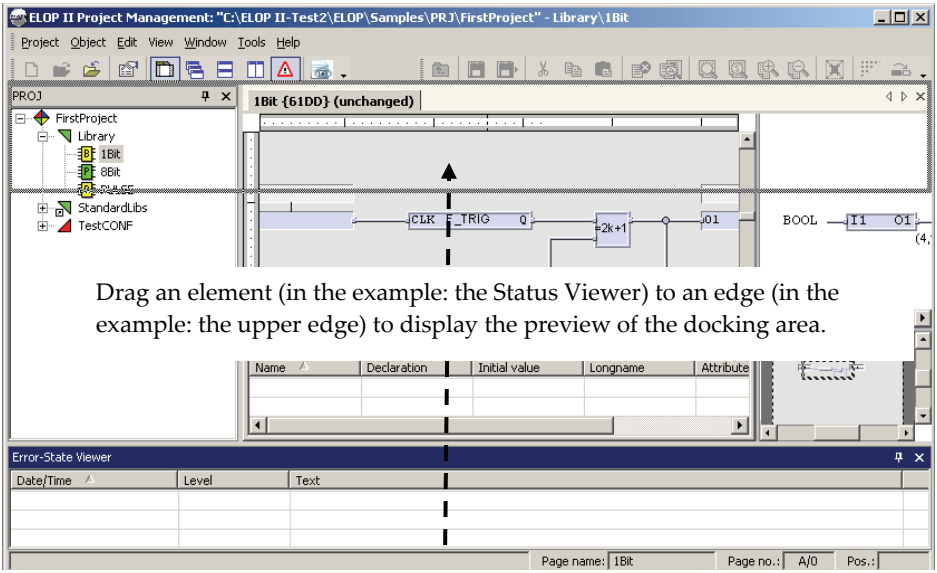
Based on this principle, ELOP II offers new options for customizing the user interface.

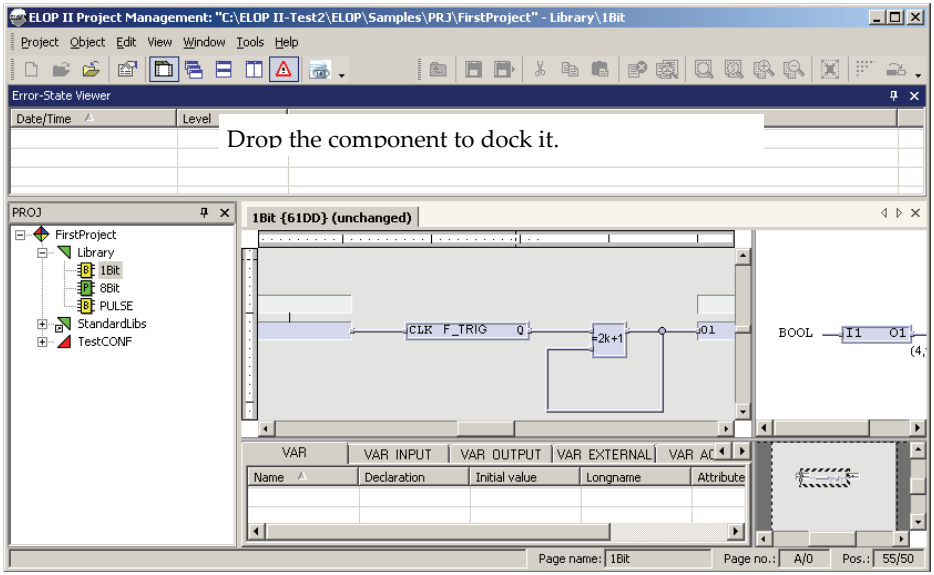
3 Using the User Interface Efficiently

3.1 Dockable Windows: Moving or Docking Elements

Thanks to the "dockable windows" functionality, the main components of ELOP II can be moved to any position on the screen, even to additional screens (see on page 23).

Hit the **docking areas** at the window edges of ELOP II and dock the components there.

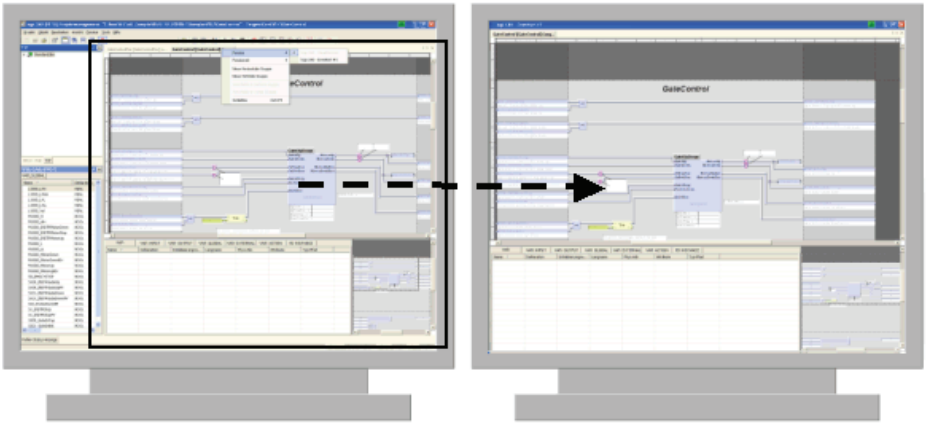




Based on this principle, the main components of ELOP II can be arbitrarily moved and arranged – and even combined into one unit. It might sound sophisticated, but it is really easy to use. Just give it a try – the default layout can be restored at any time (using the menu *View*, *Restore Default Layout*)

3.2 Using ELOP II on Additional Screens

Based on the "dockable windows" functionality, individual ELOP II elements can be displayed on additional screens. This results in more screen space for working in ELOP II.



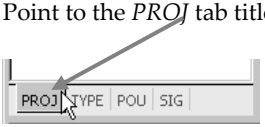
Should, for instance, the structure windows be moved to a 2nd screen (tabs *PROJ*, *POU* and *TYPE*)?

1. Point to the title bar of the *PROJ* tab.

2. Press and hold mouse button 1.
3. Drag the structure window to the 2nd screen.
4. Release the mouse button.

Should only the *PROJ* tab be moved to a 2nd screen?

5. Point to the *PROJ* tab title.



6. Press and hold mouse button 1.
7. Drag the tab to the 2nd screen.
8. Release the mouse button.

Should the FBD Editor be moved to a 2nd screen?

9. Right-click the tab title of the FBD Editor (showing the POU name) to open the context menu.

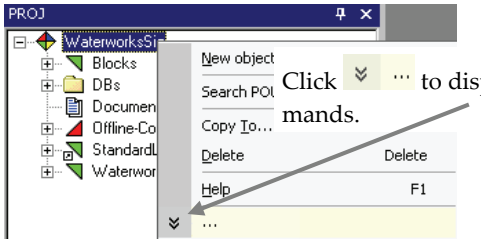
10. Point to *Window* and select *ELOP II - Extended #1.*
    An additional sub-command is displayed below *Window* for each additionally available screen.

TIP If an element cannot be moved to the additional screen, check the following: Is the *Floating* command displayed in the context menu?

- If yes, select *Floating* and then drag the element to the additional screen.
- If no, the component is part of an ELOP II section. It cannot be moved (as an individual element) to an additional screen.
  Example: The Variable Declaration Editor is part of the FBD Editor. In this case, drag the ELOP II element (e.g., the FBD editor) to the additional screen and the component (e.g., the Variable Declaration Editor) will be moved together with it.

3.3 Shorter, Personalized Menus

By default (if the user profile for beginners is activated), personalized menus are available in ELOP II. That means that short versions of the menus are shown – only the basic commands or often used commands are displayed.



Click ⋙ ⋯ to display all menu commands.

Note To show all new features of V5.1, the default settings must be active in the project properties. To this end, open *Project Properties*, tab *Additional* and check the following settings:

- Remove all settings manually by the user (black text).

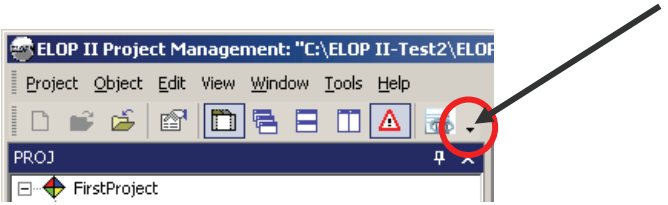
- Activate the *Parent Data* checkbox.

3.4 Office-Like Toolbars and Menus

The ELOP II interface has been enhanced so that it is close to the Office-style.

3.4.1 Adjusting Toolbars

More comfort The toolbars of the Project Management are offering this new button:

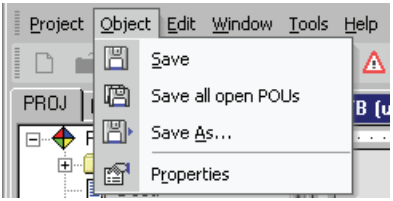


It features Office-like functionality, e.g., to visualize/hide buttons permanently, to create customized toolbars, etc.

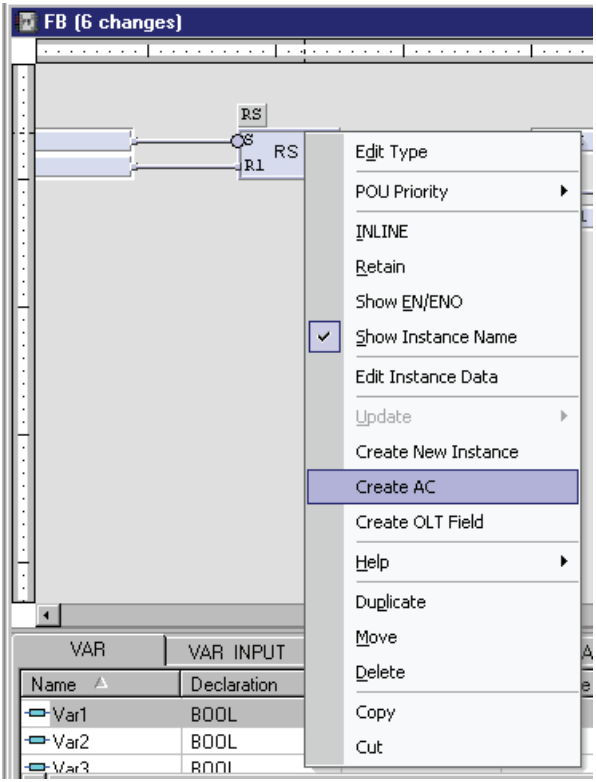
3.4.2 Menus and Context Menus

Additionally, the menus (of the menu bar) and the context menus have been adapted to Office-style as well.

Example for a menu in ELOP II:



Example for a context menu in ELOP II:



CHAPTER 4

Project Management

This chapter describes the new features of the Project Management.

4 Project Management

4.1 Support of Unicode: Input and Representation of Complex and Non-Latin Characters

Beginning with version 5.1, complex characters (Unicode characters that are not included in the ANSI character set) can be used in ELOP II .

What is Unicode? Unicode is an international standard that provides a unique number for every character – irrespective of the system, program and language.

This allows one to also represent characters of non-Latin languages correctly (e.g., Greek, Russian, Japanese). In this document, these characters are referred to as complex characters or characters not included in the ANSI character set.
Unicode is thus a pre-requisite for being able to globally exchange characters (i.e., texts) without loosing information.

Note Use the ASCII code instead of the Unicode for names that are loaded into the H41q/H51q controller. The following names are concerned:

- Name of the program instance
- Name of the resource
- Name of the configuration
- Name of variables
- Name of POU types
- Name of POU instances
- Name of data types

4.1.1 Are You Fit for Unicode?

The Web site of the Unicode Consortium http://www.unicode.org/ offers detailed information about Unicode. For instance, http://www.unicode.org/standard/WhatIsUnicode.html provides hyperlinks to pages in different languages and different character sets.

Additionally, the ELOP II online help provides the most important facts, e.g., what is UTF-8, an ANSI character set, etc.

4.1.2 Is Your Operating System Ready for Unicode?

If rectangles or question marks appears in the browser instead of the expected characters, the operating system is not configured correctly.

Recommended by HIMA

Use **Windows XP** or Windows 7 Professional or Windows 7 Ultimate as operating system!
Please contact HIMA for further information or support, if unexpected representation problems occur under Windows XP/7 and no workaround is provided in Chapter "Unicode in ELOP II".

Configure the operating system as follows:

- To configure how Unicode characters will be displayed
- Under Windows XP:
  Control Panel, Regional and Language Options, tab Languages, activate Install files for complex script and right-to-left languages as well as Install files for East Asian languages
- Under Windows 7:
  no actions required
- **Install Unicode-enabled font file:**
  Control Panel, *Fonts*, menu *File*, *Install New Font...*
  Note: Tahoma or Lucida Sans Unicode are Unicode-enabled font files and already available as a default under Windows XP/7. Beginning with Microsoft ® Office 2000, the Arial Unicode MS font is also available.
- To configure how Unicode characters are entered (optional)
- Under Windows XP:
  Select *Control Panel*, *Regional and Language Options*, tab *Languages*, click *Details…*, click *Add*, select the required input language and keyboard layout, then click *OK*
- Under Windows 7:
  Select *Control Panel*, click *Clock, Language, and Region*, click *Change keyboards or other input methods*, tab *Keyboard and Languages*, click *Change keyboards...*, click *Add*, select the required input language and keyboard layout, then click *OK*
- Additionally, under Windows XP/7:
  Click the current input method icon located near the system time on the Windows task bar and select the required input method

Additional Info... On how to set up the operating system for Unicode support can be found in the Windows help.

4.1.3 Unicode in ELOP II

To avoid that rectangles or question marks are displayed in ELOP II instead of the expected characters, check the following points prior to installing ELOP II, and perform the required steps:

- **Install** ELOP II **in** a **directory** the **name** of which contains only characters included in the **ANSI character set**. Such directories must not contain, e.g., any Greek or Japanese characters.
- If some components of ELOP II display rectangles or question marks, but other the expected characters, a **wrong font** has been set for the component with the incorrect characters.
  Example: The name of the variable is displayed correctly within the Variable Declaration Editor, but not in the value field set in the FBD drawing area. Explanation: The value field has not been configured for using a Unicode-enabled font (properties for value field, tab *Text*, setting under *Font*).
  TIP: In the template project, define a Unicode-enabled font as default for future projects/objects: menu *Project*, *Open Template Project*, open the appropriate object template (e.g., Function Block), context menu for the drawing area, *Properties*, tab *FBD Objects*, corresponding button (e.g., *Value Field*), tab *Text*, select the specific object type and modify the setting under *Font*.
- Prior to printing ELOP II projects containing complex characters, make sure that the used fonts are supported by the printer driver. If the projects are printed to a PDF printer driver, HIMA recommends to embedding the fonts. Otherwise, the complex characters might not be displayed correctly in the printout or the PDF file.
  Refer to the documentation of the used printer driver for details on the supported/embedded fonts.
- The **offline simulation** and the **functions for connecting the target system** (e.g., code generation) only support the representation of characters included in the ANSI character set.
- In **DXF forms**, only enter characters included in the ANSI character set. Other characters such as Japanese characters are not displayed correctly in the printouts (ELOP II printouts are based on the DXF forms).
  Nonetheless, if such characters are required in the DXF forms, use

the following workaround: Define user-specific identifiers n the DXF form (see online help, search for Identifier). In the *Master Data* tab, define the identifier as key and the corresponding Unicode characters as its value (see online help, search for Print, Entering User-Defined Additional Information).
Alternatively: Enter the required characters directly in the DXF form. In doing so, specify the characters in accordance with UTF-8. This entries are processed correctly in the ELOP II printouts. Condition: The Unicode support for ELOP II must be extended by the ELOP II administrator.

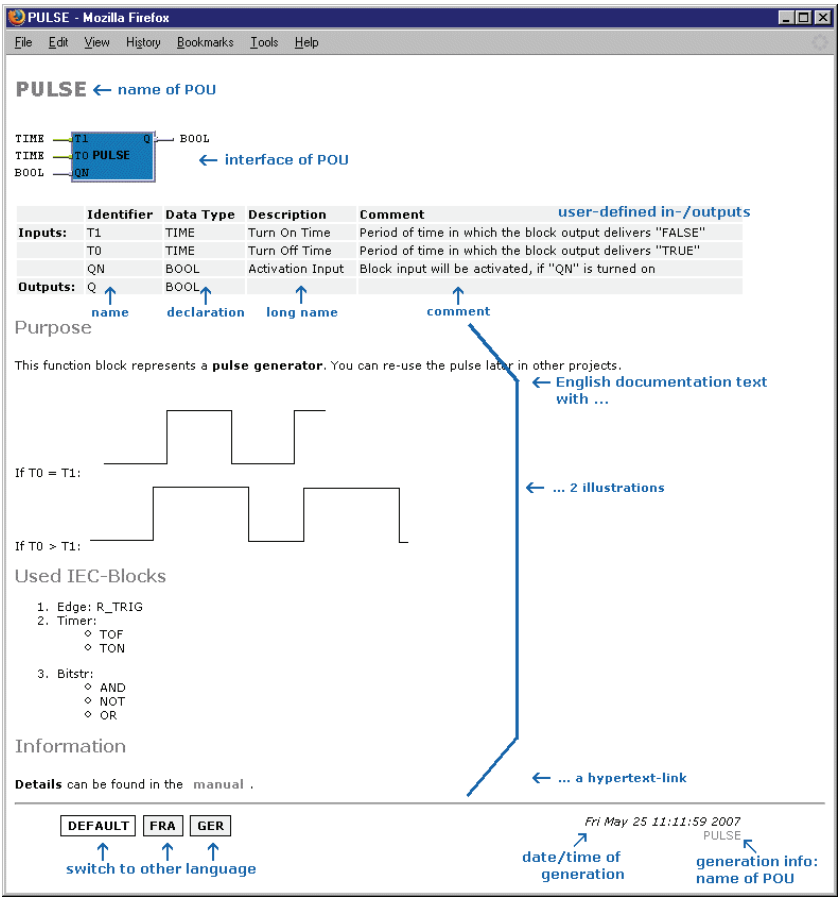
- ELOP II features associated with import, export and conversion were extended to allow data processing in ASCII and UTF-8. Shortly:
- `--IOOut:` New option for **additional tools** LCExpSet, LCConvVer and LCCheckObj
  This option is used to define the format used by the additional tools for the text output.
- Select file format for **output in text files**: ASCII or UTF-8
  The command *Dump All to Text File....* is sued to export the displayed table content into a text file (e.g., the variables displayed in the Variable Declaration Editor). The requested file format can now be selected.
  The file format can also be selected when logging the oscilloscope data.
- Format for **export/import** of variables/signals, instance data (**with definition files**): ASCII or UTF-8
  By default, data are exported/imported in ASCII format when exporting/importing with definition files. However, if the UTF-8 format should be used for the export/import, define `$EXPORTFORMAT` as new keyword of the definition files and add `UTF8` in the next line.

  Refer to the online help for more information on how to export or import with definition files.

4.2 Easily Generating Multilingual HTML Documents for POU

At this point, HTML online documentation for ELOP II POUs can be entered, generated and displayed in different languages. The online documentation is based on the information entered directly in the POU. This means that POUs and all data relevant for the documentation are managed together.

Example for a multilingual documentation (in HTML format) that was generated based on POU data:

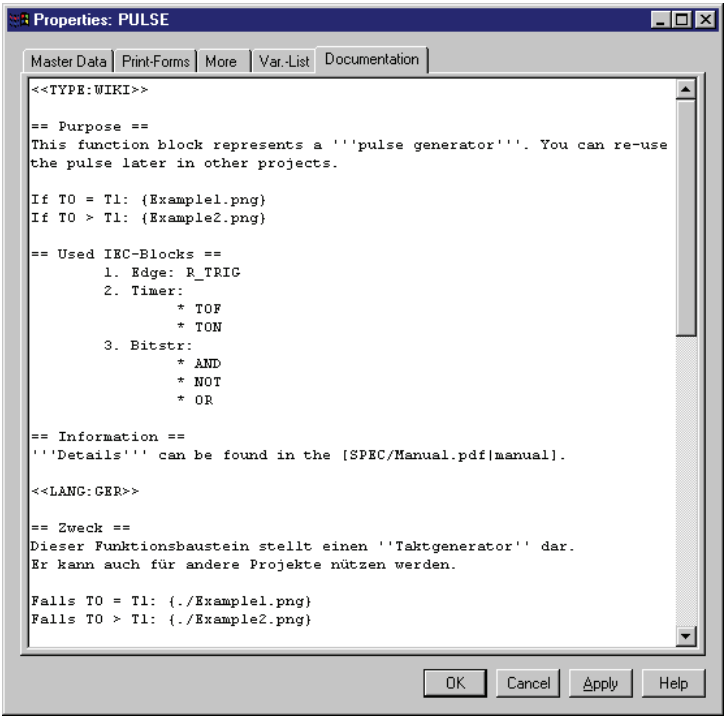


4.2.1 Sources of the POU Documentation

All data used in this example were previously entered in the POU:

- **POU Name** (context menu associated with the POU, command *Rename*)
  The name of the POU becomes the title and header of the online documentation.
- **POU interface** (such as defined in the Interface Declaration Editor)
  The interface is integrated in the online documentation as a graphic.
- **User-defined in-/outputs** (i.e., variables that were defined in the `VAR_INPUT` and `VAR_OUTPUT` tabs of the Variable Declaration Editor, but without the default input `EN` and default output `ENO`)
  The table contains the variable name (and potentially, an alternative identifier), the declaration, the long name and the comment for the variable. The row/column names of the table are always generated in English language.
- **Documentation text** in the *Documentation* tab (located in the POU properties)
  It is used to define the documentation body. Use formatting instructions! See the following example.
  These formatting instructions are used to define multilingual online documentation, graphics and hyperlinks. They also define the text formatting layout.
- If required: **property `GENDOC.COLOR`** in the *More* tab (located in the POU's properties)
  Define this property for the POU and enter an HTML color code as its value. The HMTL color code `#c0c0c0` was entered in the example and the color gray is then used, e.g., for the headers.

Example for data entered in the *Documentation* tab:



Additional info.. On the formatting instructions can be found in the online help for ELOP II (search for POU, Enter User-Defined Additional Text).
The online help contains a list of all possible instructions with examples.

4.2.2 How to Generate Documentation

1. Right-click an object to open the corresponding context menu (e.g., a POU, a library or the project).
2. Point to POU Documentation.
3. Select *Generate*.
4. The online documentation (in HTML-format) is generated for the user-defined POU (or all user-defined POUs existing in the directory object).

4.2.3 How to Display Documentation

The fastest and easiest way is to select the POU and to press the F1-KEY. Alternatively, right-click the POU and select *Help* from the context menu.

The user-defined documentation is displayed in the browser usually used for HTML files.

If no user-defined documentation is available for the object, the ELOP II help is usually displayed.

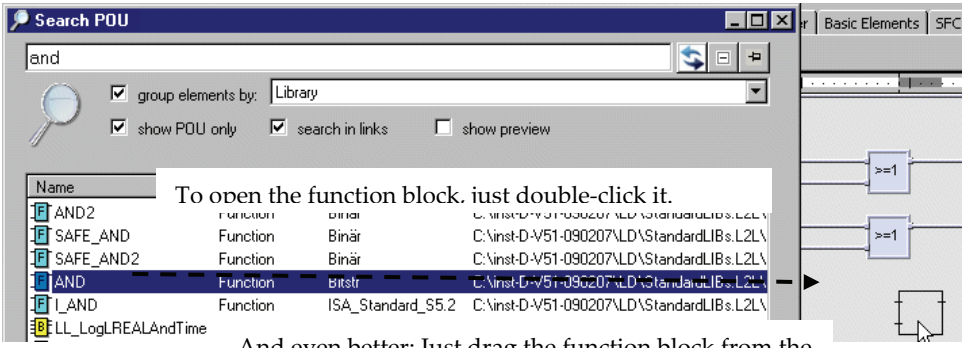
TIP If errors are detected in the generated documentation, it is best to check the generated files together with the sources created for the documentation.
To go quickly to the directory containing the generated HTML-files: context menu associated with the POU, *POU Documentation* and *Explorer*

4.3 Quickly Finding/Using POUs (Quick Function Block Search)

ELOP II offers numerous POUs (function blocks) in the standard libraries. A specific POU is needed, but cannot be found?
The POU search in ELOP II provides a solution to this problem. ELOP II performs the search and finds the required POU!

Start the POU search, e.g., selecting *Search POU...* from the context menu associated with the project. In the dialog box, type a part of the POU name and the search results are listed next to the search box ("live search).

Advantage The POUs can be directly used from the dialog box.



To open the function block, just double-click it.

And even better: Just drag the function block from the dialog box into the open function chart to set it there.

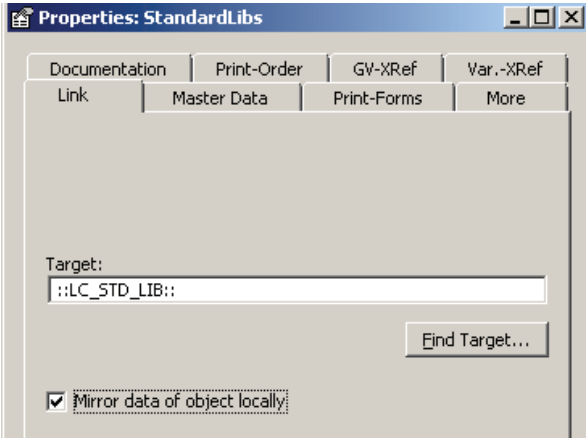
4.4 Efficiently Working in Team: Mirroring and Updating Network Data

With ELOP II, data (e.g., libraries) contained in the network can be locally mirrored in the ELOP II projects!

Advantage The locally mirrored data enables faster access to data, whereas the originals in the network can continue to be modified without restrictions. Offline work, i.e., without direct connection the originals in the network, is also possible.

4.4.1 Mirroring Network Data

1. Open the properties for a link: context menu, *Properties*
2. Go to the *Link* tab.
3. Select Mirror data of object locally.



4. Click *OK* in the tab.
   Result: A prompt appears asking whether the data should be locally mirrored at once.
5. Click *Yes* to confirm.
   Result: Data is mirrored locally. The duration of the mirroring process depends on the size of the data to be mirrored.

4.4.2 Updating Mirrored Data

ELOP II indicates whether the network data have changed this by changing the icon for the mirrored link.

For instance, the icon indicates that the data mirrored for the library is out-of-date. Notice the red-colored rectangle in the icon!

Advantage The updating function maintains the mirrored **data up-to-date**, and **the changes are only adopted after a previous check**:

1. Right-click the link to open the context menu.
   Alternatively: Right-click the project to update all mirrored data contained in the project.
2. Select *Update Mirrored Data*.
   Result: The locally mirrored data is updated. The duration of the mirroring process depends on the size of the data to be mirrored.

Shortcut Select the link or the project and press the F9-KEY.

TIP Prior to updating, check which data will be updated: Select the *Update Mirrored Data (Preview)* command or press CTRL+F9.

Additional Info... On the mirrored data (e.g., how to delete it or which icons for mirrored data are of interest) can be found in the online help of ELOP II (search for Mirrored Data).

4.4.3 Feedback on Cross-References Valid for Global Variables

If the cross-reference list for global variables should be printed out or displayed in ELOP II, the cross-reference list must have already been created one time. The *Cross-References* menu (located in the context menu associated with the configurations, resources, etc.) contains the appropriate commands.

Starting with version 5.1, the *Cross-References* menu also gives a feedback on whether or when the cross-references were created:
e.g., *Cross-references not generated yet* or *Cross-references last generated on: 2009-02-11 10:47*

Based on this information, the user can decide if the cross-reference list can be displayed or printed out immediately, or if it must be created beforehand.

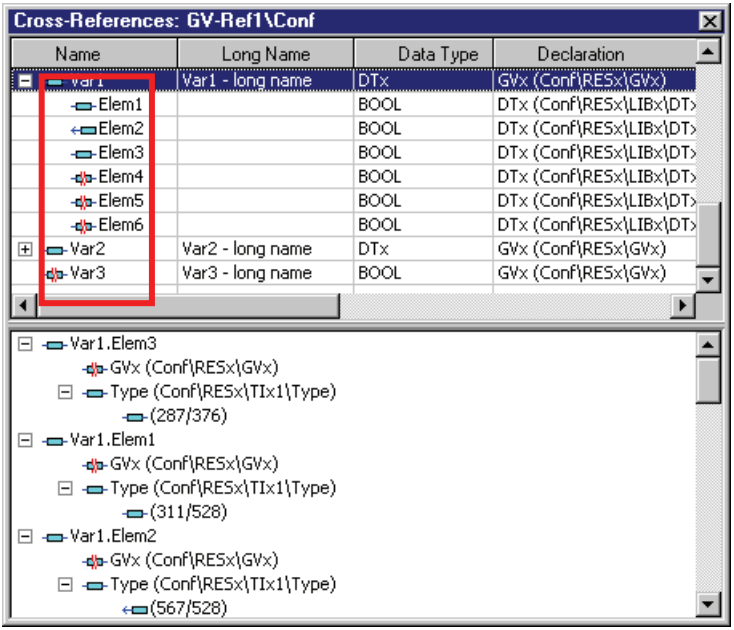
4.5 Improved Display of Cross-References for Global Variables

The *Display GV Cross-Reference...* command (right-click, e.g., the resource and select *Cross-References*) is used to display cross-references for global variables in a dialog box.

The following actions are possible:

- Sorting the displayed variables/elements according to their definition (context menu in the upper part of the dialog box: *Sort*, *By Definition*)
- Displaying unused variables/elements (context menu in the upper part of the dialog box: *View*, *Unused Elements*)
- Displaying the usage type of variables/elements (context menu in the upper part of the dialog box: *View*, *Usage*)

In the dialog box, the icon located next to the name provides details about the usage of the variable and element:



4.6 Adding/Removing the POU Protection

The new command *Add/Remove POU Protection...* is used to protect the internal logics and thus the know-how of POUs against unauthorized access.

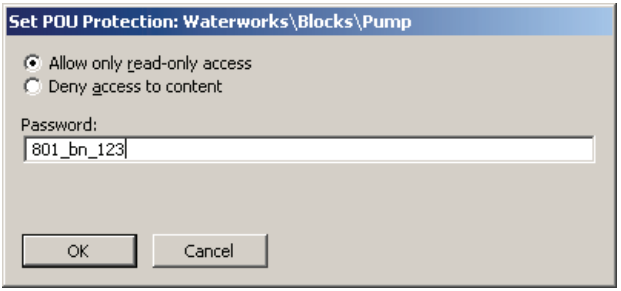
Users can use such protected POUs as usual and:

- they can view the internal logics but not modify it (i.e., the POU is opened in read-only mode) or
- they cannot view the internal logics at all (i.e., the POU cannot be opened in the POU editor).

Adding To add the POU protection for one POU:

1. Right-click a POU and select *Add/Remove POU Protection...* from the context menu.
   Result: the *Add POU Protection* dialog box is displayed.
2. Select one of these options.

3. Enter a password.



Please take note of this password because it is required for removing the POU protection.

4. Click *OK* to confirm.

The protected POU can be used as any other POU and can be forewarded for further processing. The POU protection applies until it is removed.

Removing To remove the POU protection, proceed analogously: Once again select the command *Add/Remove POU Protection...*, enter the password in the dialog box and click *OK* – done!

TIP The POU protection can be added or removed for several POUs: Right-click a library, point to *POU Protection* on the context menu and select *Add POU Protection...* or *Remove POU Protection...*.
Alternatively, the POU protection for POUs can be added or removed in the sub-folders of the library: Tick the *Apply to all POUs in sub-folders* option of the dialog box.

Password Forgotten? Password for the POU forgotten? Please contact HIMA!
HIMA will help removing the POU protection.

CHAPTER 5

Editors

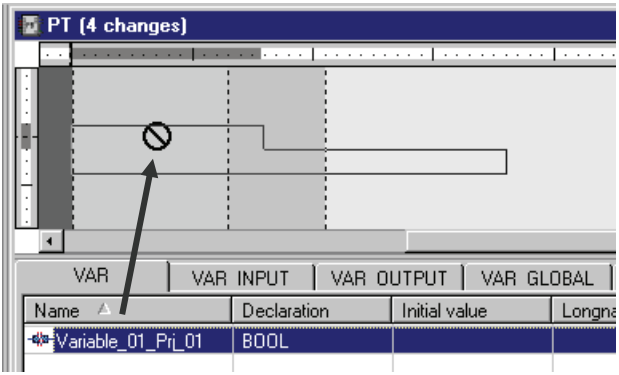
This chapter describes the new features of the editors.

5 Editors

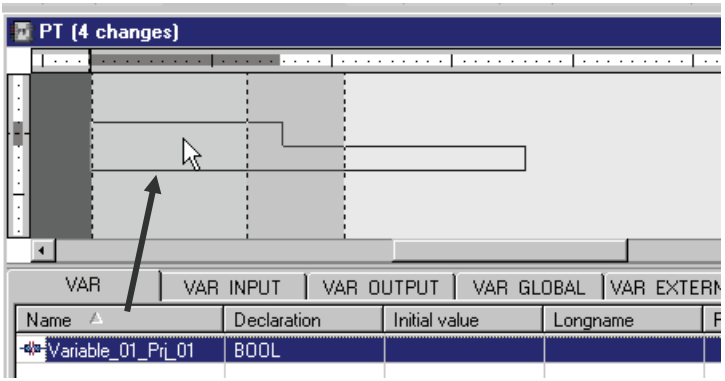
5.1 Value Fields Exceed Value Field Area

A common habit is to define variables with long names and thus to use wide value fields to prevent the variable names from being truncated. These value fields, however, do not fit into the value field area! And the **connector field area next to** the value field area does not allow value fields in it!

Previously: ELOP II version 4.1 does not allow the use of value fields exceeding the connector field area.



New: ELOP II version 5.1 allows the use of value fields exceeding the value field area.

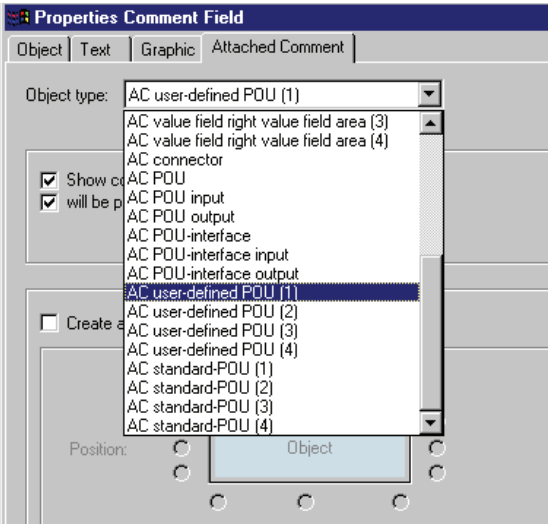


No need? Is the old functionality preferred to the new one, ask the system administrator to reactivate it using `IOFieldsNotInConnArea`.

5.2 Creating Multiple Comment Fields Attached to POUs

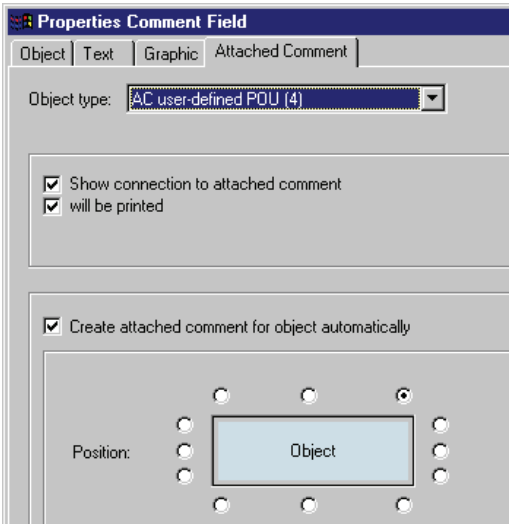
Dragging a POU into the FBD Editor, up to 4 attached comment fields can be created for each POU instance:

1. Open the pre-settings for the comment field (select *Properties* from the context menu associated with the drawing area, tab *FBD Objects*, *Comment Field*).
2. Go to the Attached Comment tab.
3. Select the required object type.
   Example: Object type *AC user-defined POU (1)* = The comment field is attached to a user-defined POU dropped in the drawing area.



4. Tick the checkbox Create attached comment for object automatically.
5. Define the position of this attached comment field by selecting one of the options under *Position*.
6. Now select the next object type *AC user-defined POU (2)*, tick the checkbox, define the position.

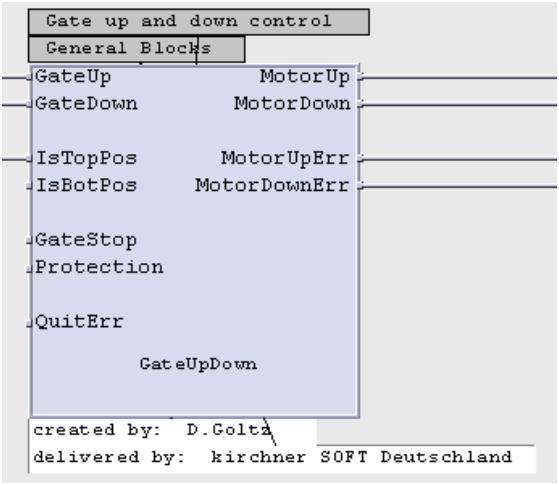
7. Repeat these steps for the object types *AC user-defined POU (3)* and *AC user-defined POU (4)*.



8. If required, repeat these steps also for the object types *AC standard POU (1)*, *AC standard POU (2)*, *AC standard POU (3)* and *AC standard POU (4)*.
   In doing so, comment fields can also be created for POUs from the standard library.
9. Click *OK* (in the properties of the comment fields and the FBD Editor).
10. If a POU is set in the drawing area (e.g., by dragging the POU from the structure window), multiple attached comment fields will be created.

Note Different contents and sizes can be defined for these comment fields – also in the pre-settings for the comment field: in the tabs *Object* and *Text*

Example POU instance with 4 attached comment fields, automatically created when setting the POU instance:



Clear text and format strings were entered as contents for these attached comment fields. The format strings are evaluated in conjunction with the POU as the specific pieces of information (e.g., `D.Goltz)`.

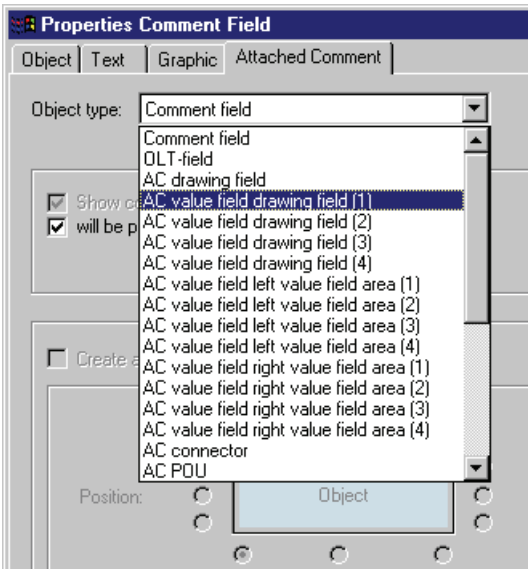
TIP Attached comment fields automatically created are particularly suitable for displaying additional information such as instance names or function-block-specific details.
Format strings can also be used to evaluate this instance data (see online help, searching for Function block, Show Instance Data).

Configuration Exceptions to the automatic creation of attached comment fields can be defined in the ELOP II Project Management.
To do so, define *More* properties (e.g., `CreateACForUserPou1`) with the POU. Refer to the online help for further details.

5.3 Creating Multiple Comment Fields Attached to Value Fields

Analogously to Creating Multiple Comment Fields Attached to POUs (on page 45), up to 4 attached comment fields can be created for each value field, if a value field is set in the FBD Editor:

1. Open the pre-settings for the comment field (select *Properties* from the context menu associated with the drawing area, tab *FBD Objects*, *Comment Field*).
   Go to the Attached Comment tab.
   Select the required object type.
   Example: Object type *AC value field drawing field (1)* = The comment field is attached to a value field located in the drawing area.

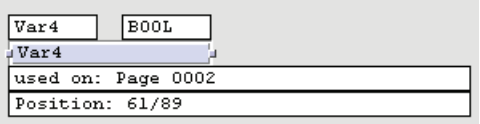


   Note: The very first object type, e.g., *AC value field drawing field (1)*, replaces the object type existing up to now, e.g., *AC value field drawing field*.
2. Tick the checkbox Create attached comment for object automatically.
3. Define the position of this attached comment field by selecting one of the options under *Position*.
4. Now select the next object type *AC value field drawing field (2)*, tick the checkbox, define the position.
5. Repeat these steps for the object types AC value field drawing field (3) and AC value field drawing field (4).
6. Click *OK* (in the properties of the comment fields and the FBD Editor).

7. If a value field is set in the drawing area (e.g., by dragging a variable from the Variable Declaration Editor), multiple attached comment fields will be created.

Note Different contents and sizes can be defined for these comment fields – also in the pre-settings for the comment field: in the tabs *Object* and *Text*

Example *V*alue field with 4 attached comment fields, automatically created when setting the value field:



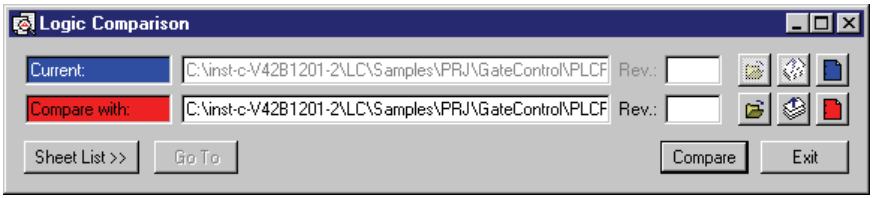
5.4 Visual Logic Comparison for POUs

ELOP II version 5.1 offers a comfortable **logic comparison** for already drawn logics!

What for? The logic comparison is available for **POUs** (created with or converted to the current version) and for **archived POU versions** (without having to copy them from the version management).
The current edit state of a POU can also be compared to its last saved state.

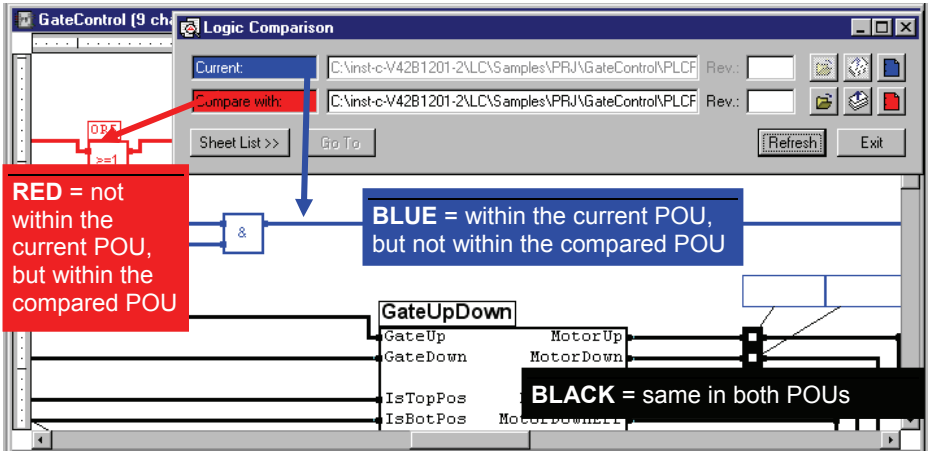
To use the logic comparison:

1. Open a POU (FBD) in ELOP II.
2. Click  (*Start Logic Comparison*) in the toolbar.
3. Enter the POU to be compared under *Compare with* (or click  to select a POU):



4. Click *Compare*.

5. The logic comparison provides an immediate **visual overview** of the **differences** between the POUs. For example:



Advantage: Continue editing The current POU can be (further) edited: e.g., new function blocks can be set in the drawing area or the existing ones can be manipulated, the view can be modified (selecting a different zoom factor), etc.
This makes it really easy to **copy** the **various logic elements** from the compared POU to the currently edited POU **or** to **comment changes** with comment fields.

Additional info... On how to use the logic comparison (e.g., how to compare the current POU with an archived version) can be found in the online help. under Comparing Logic for POU(FBD) (e.g., by searching for Logic, Comparing POUs).

5.5 Starting the Search from within the FBD Editor

The well known command *Find/Replace...* can be used to find POUs and global variable objects in the ELOP II Project Management and to find/replace certain texts in them.

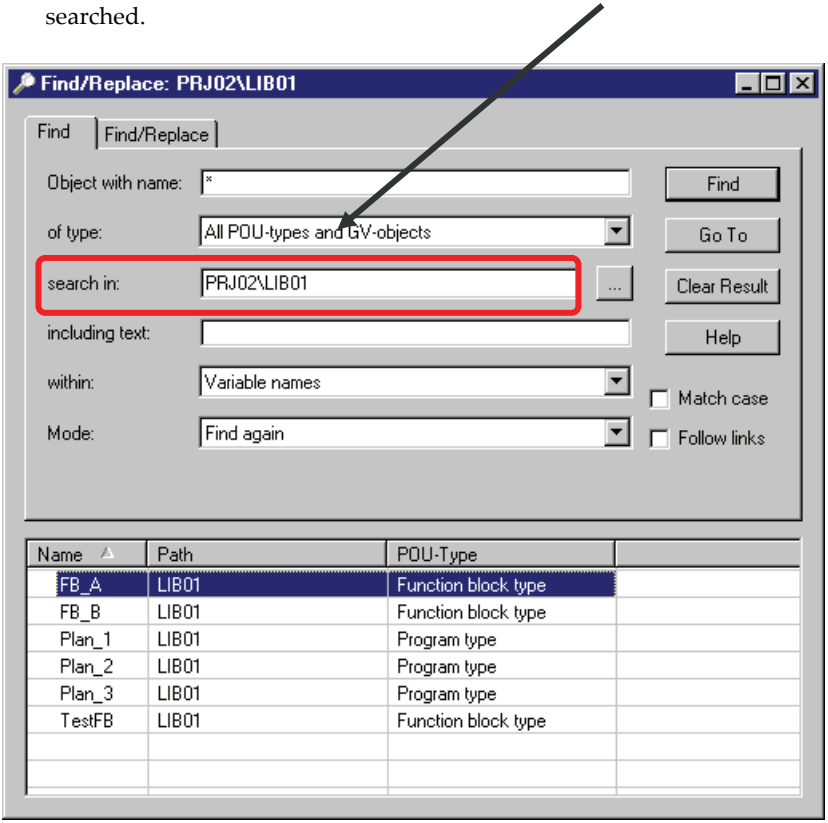
Up to now, this feature could only be started from within the Project Management. To achieve more comfort, the search can now be started from additional positions:

- **Variables** in the Variable Declaration Editor (command *Find Occurrences...* on the context menu) and
- **Value fields** in the drawing area (command *Find Occurrences...* on the context menu)

Note The search criteria for the Find feature are automatically set for the current variable context, e.g., the variable name as search term.

5.5.1 Defining Search Range

Additionally, the **search range** can be defined for each search process. In doing so, only the objects contained in the directory object are searched.



Enter the object name in the field next to *Find in* or click ... and select the object via a selection dialog.

5.6 Using Search Results: Go To, Status Area and Dragging into POUs

The search results are displayed in the lower part of the dialog box (see image above). Interesting options are available:

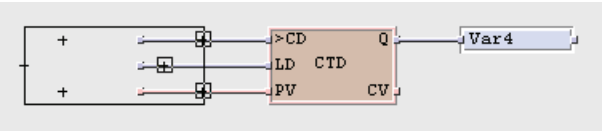
- Stepping through **all text usages** (e.g., variable usages) in the found POUs: Select a POU in the list of the search results and click *Go To*. Click the *Next* button to display the next usage.
- The new status area of the dialog box provides **additional information about POUs**:
  When searching for variables, details about the usages of variables in the selected POU are given: how often the variables are read, how often they are written to, etc.
  When searching for connectors, the status area shows the number of source and sink connectors in the selected POUs.
- Additionally, any found **function block type or function** can be **dragged** easily **from** the **list** of the search results into a POU – thus, creating a function call or FB instance in the POU.

5.7 Docking Selections

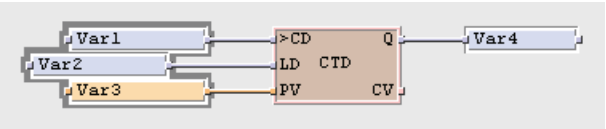
What is docking? The docking feature of ELOP II consists in moving objects (such as function blocks or value fields) in the drawing area of the FBD Editor and connecting them to already existing lines.

So far, only individual objects could be docked. In version 5.1, also selected elements can be connected to lines!

How? Use the same procedure as for docking individual objects:
Drag the selection onto or over a line. A preview shows whether docking is possible:



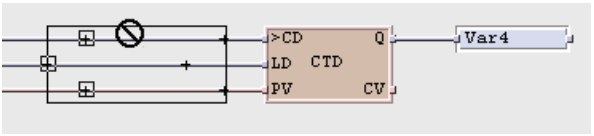
Release the mouse button (i.e., position the selection) and the selection is docked:



As usual, ELOP II automatically shortens the lines and indicates type conflicts.

Docking not possible

There is a difference to docking single objects:
Selections cannot be dragged and docked **over lines** that exceed the preview margins:



As usual, the pointer with the shape of a prohibitory sign (⃠) indicates that the selection cannot be docked.

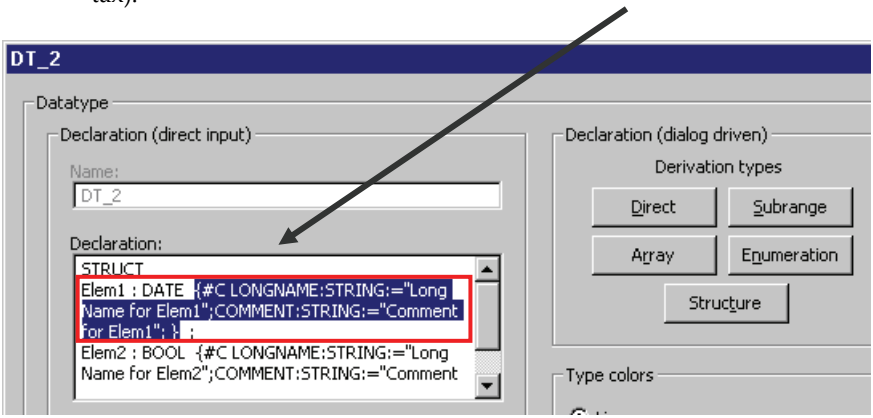
5.7.1 Docking with Pasting via Clipboard

The docking feature is additionally supported in function pasting via clipboard.

5.8 Fast Editing Long Name and Comment for Structure Declaration

Up to now, the long name and/or comment for elements of the structure declaration was entered or modified in the *Declaration - Structure* dialog box.

Starting with version 5.0, the long name and comment are displayed in the *Declaration* field of the Data Type Editor (according to the IEC syntax):



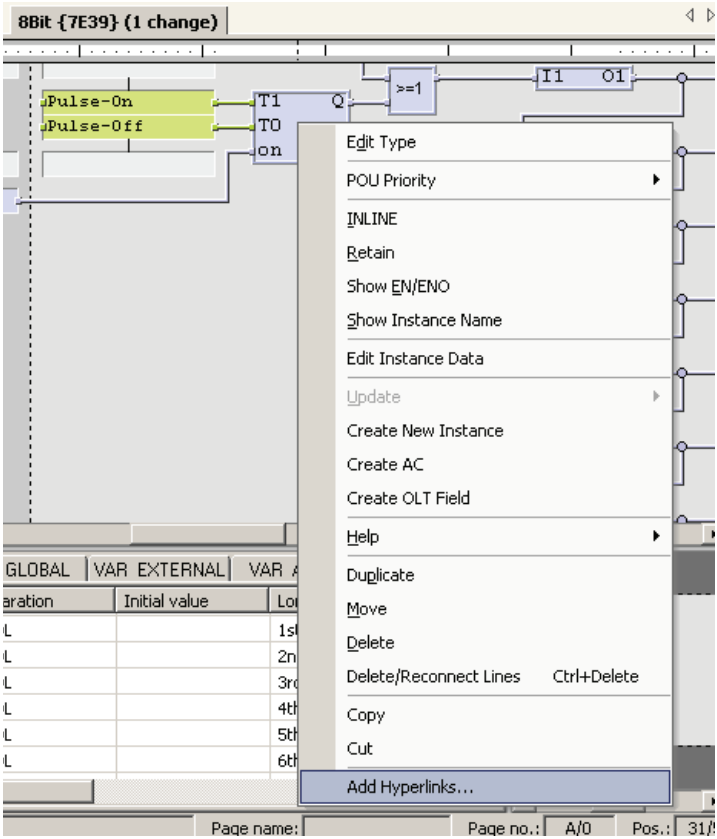
TIP Use this field to rapidly change an already defined long name or comment: Just overwrite the text for the long name or comment.
If the IEC syntax is not familiar: Continue to use the dialog-supported declaration.

5.9 Inserting Hyperlinks for FBD Elements

Insert hyperlinks to any files (e.g., documents) or URLs within the existing logic, i.e., for the drawing area or any FBD element (excepted lines) set in the drawing area.

This makes it easy to open these documents or Web pages directly from within ELOP II or to navigate to a FTP server.

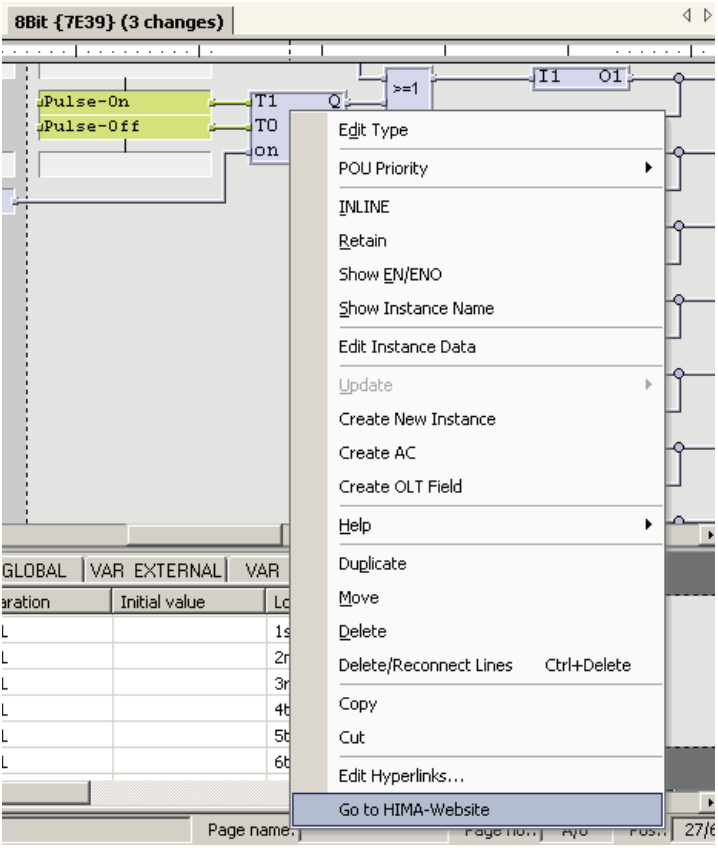
1. Open the context menu, e.g., with a function block.
2. Select *Add Hyperlinks...*.



Result: The *Hyperlink Editor* dialog box is displayed.

3. Click *Create New Link...*
   Result: The *Add Link d*ialog box is displayed.
4. Enter the name of the hyperlink under *Name*.
5. Select the type of the hyperlinks under *Type*: *HTTP link, FTP link* or *File link*
   Restriction: With *File link*, only files within the current project can be selected.
6. Enter the target of the hyperlink under *Address*.
   Alternative for file link: Click ... and select the target in the selection dialog box.

7. If required for the *HTTP link* or *FTP link* type, tick the *Server needs Authentication* option and enter the corresponding data under *User* and *Password*.
8. Click *OK* in the *Add Link* dialog box.
   Result: The hyperlink is displayed in the *Hyperlink Editor* dialog box.
9. Click *OK* in the *Hyperlink Editor* dialog box.
   Result: The context menu associated with the function block displays the name of the hyperlink as additional command.



10. Select this command to open the target of the hyperlink.

Note If the type of a hyperlink is modified, but the target of the hyperlink under Address has already been entered, the corresponding text

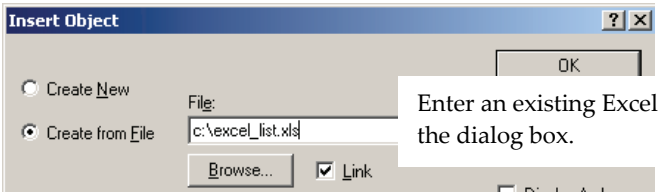
"`http://`" or "`ftp://`" under *Address* must be manually corrected/deleted.
Example: If a ready HTTP link is changed to a File link, the text "`http://`" under *Address* must be deleted.

5.10 Inserting Office Documents (OLE-Objects) in the Function Chart

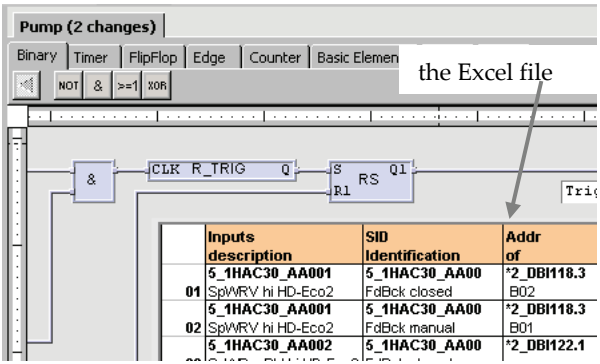
Are the following Office documents part of the function chart?

- Specifications (Microsoft® Word files or PDF files),
- Spreadsheets (Microsoft® Excel files) or
- Similar documents (e.g., Paint Brush images)

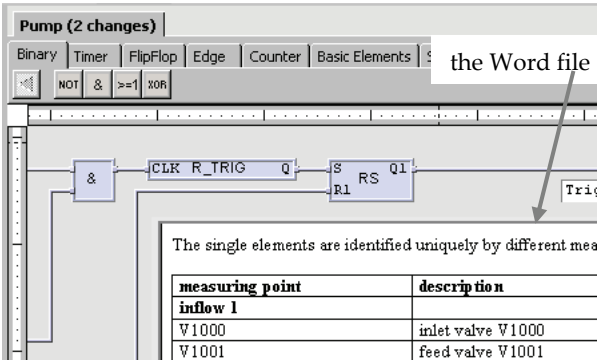
ELOP II makes it easy to insert them as linked or embedded OLE object in the function chart: Select *Insert OLE-Object...* from the context menu associated with the drawing area



Example for an inserted Excel file:



Example for an inserted Word file:



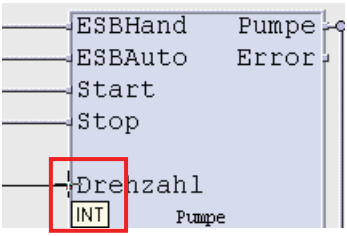
Note Changes to the source files of **Attached OLE Objects** are reproduced in the function chart, however, the source file must be provided with the (POE) function chart. In contrast, embedded OLE objects are a part of the POU and need not be provided with the function chart.

Still an advantage: The OLE object can be directly modified in or from within the function chart.

5.11 More Information on FBD Objects as Screen Tips

Starting with version 5.1, ELOP II shows screen tips for certain object types located in the FBD Editor.

Example: Trying to connect a line to the the function block input, the `INT` data type is displayed as screen tip.



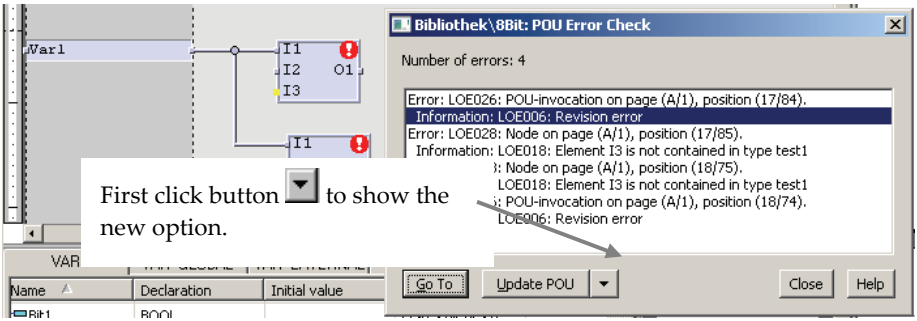
It is only a brief feedback, but immediately providing the required details.

Start ELOP II and find out which objects within the FBD Editor display a screen tip!

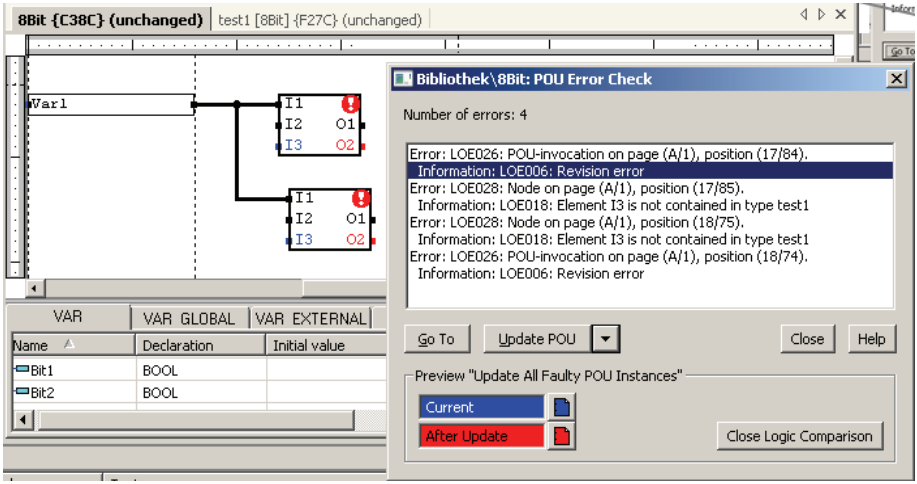
5.12 Better Recognizing Changed POU-Interfaces: Preview for Updating the POU Interfaces

If function calls and FB instances (i.e., POUs) in the FBD Editor are marked as faulty with , the interface of the POU changed. It must be updated quickly to get the correct interface.

As usual, the POU error check indicates the faulty interface. Now it offers the new option *Preview "Update All Faulty POU Instances"*:



The new option starts a preview providing an overview of the differences before and after the update:

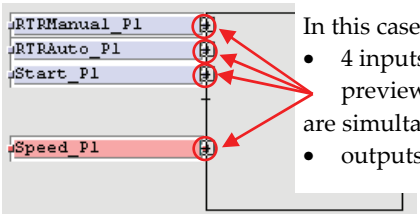


This preview matches the logic comparison (see "Visual Logic Comparison for POUs" on page 49).
So it is still possible to modify the current POU and – of course – to update the POU interface. For a quicker update, new options are provided under *Update POU*.

5.13 Setting Function Blocks/Signals/Variables and Drawing Lines

ELOP II offers the instant possibility to set FBD-elements with nodes (such as blocks, variables) in the drawing field and to have the connection lines to existing connection nodes (in-/outputs) drawn at the same time:

1. Drag the function block (e.g., from the library) or the variable (e.g., from the Variable Declaration Editor) into the drawing area.
   Result: The preview of the function block or value field appears.
2. Position the preview so that at least one input or put of the preview lies over an output or input of an already set element.



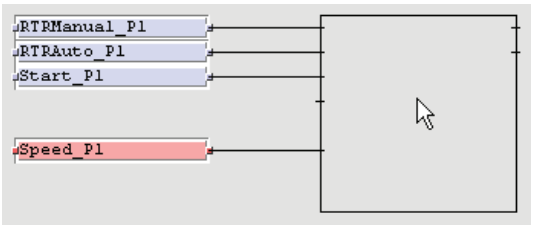
In this case

- 4 inputs of the function block preview

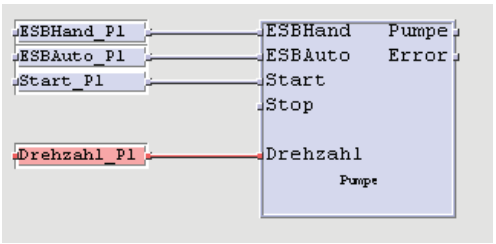
are simultaneously located on

- outputs of 4 already set value fields

3. Click mouse button 2 and drag the preview of the function block or value field to the desired position.
   Result: In addition to the function block preview, also the preview of the connection lines between the given output and input is displayed.



4. If required and as often as necessary: Position the preview once again so that at least one input or output of the preview lies over an output or input of an already set element. Then, once again click mouse button 2.
5. Set the function block or value field (by releasing mouse button 1).
   Result: The connection lines are automatically drawn.



Note A wrong input or output was accidentally used as connection node? Then, press the SPACE BAR **prior to** setting the function block or value

field (and thus before the lines are drawn). The SPACE BAR is used to undo the selection of the last connection node.

TIP ELOP II also offers this instant option of setting elements and simultaneously drawing lines in the following cases:

- Dragging blocks from the POU search into the drawing area.
- Moving/copying function blocks or value fields within the drawing area.

Just try where to use this option!

5.14 Changing Alignment, Size, Spacing of Multiple FBD Elements

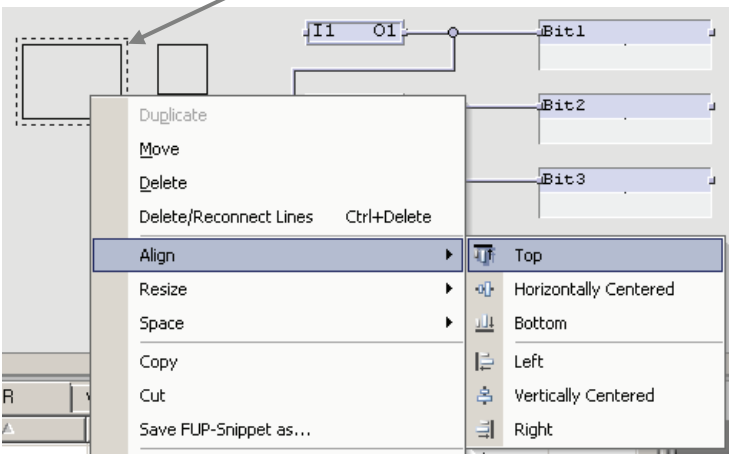
If multiple FBD elements have already been set and aligned in the drawing area, each extension or modification of the existing logic might result in the requirement that all FBD elements must be re-aligned.

Example The 11th value field is set and the name of the variable is truncated. Must each of the 11 value fields be resized or re-aligned individually?
No, ELOP II offers layout features to change alignment, size and spacing of various FBD elements in a few steps.

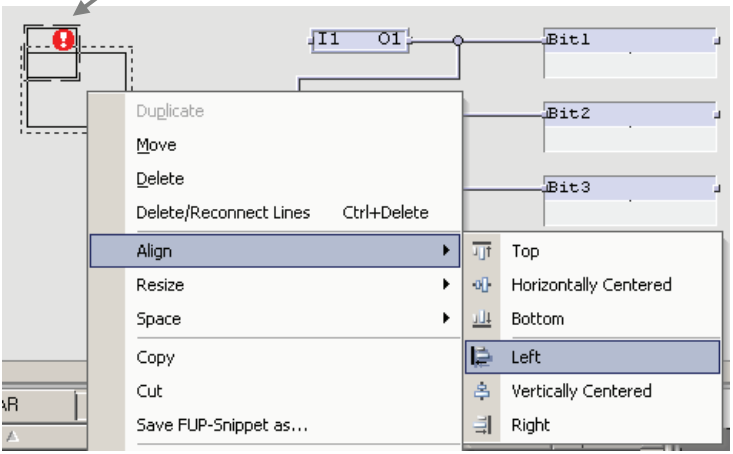
To align FBD-elements

1. Select the FBD elements that should be aligned.
2. Open the context menu over the reference object.
   The FBD elements are aligned according to this reference object.
3. Point to *Align*.

4. Point to the required command. Example: *Upper Edge* or *Left Aligned*
Result: ELOP II displays a preview showing the order in which the FBD element will be aligned.



Please observe : means that the alignment is not available (e.g., because the FDB elements would overlap).



5. If the preview is satisfactory, select the command.
Result: The selected FBD elements are aligned accordingly. If required, connected lines are re-routed by the auto router and attached comment fields (those not selected) are moved accordingly.

If FBD elements should be resized or their spacing changed, proceed as described above. The context menu associated with the selected FBD elements contains the required commands: *Resize* and *Space*.

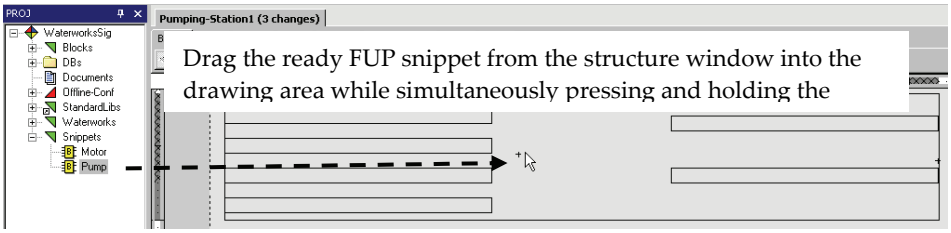
Note ELOP II provides additional options for manipulating lines and base points (e.g.,for drawing longer lines or moving base points).
Refer to the online help for more information (search for the topics "Segments (Lines) Connect Base Points" and "Base Points are Line Junctions"

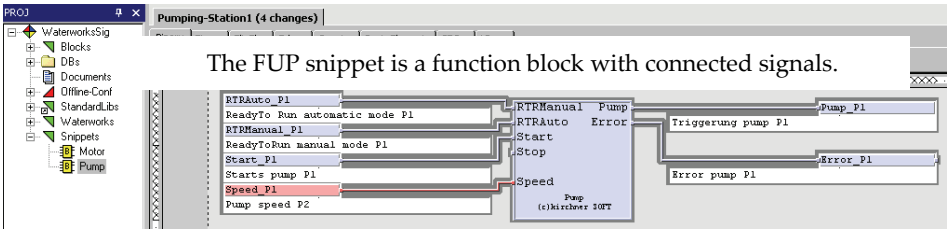
5.15 FUP Snippets: Combining Multiple FBD Elements for Further Use

Should the same FBD elements be used again and again within the function charts? Then, ELOP II increases productivity by providing FUP snippets.

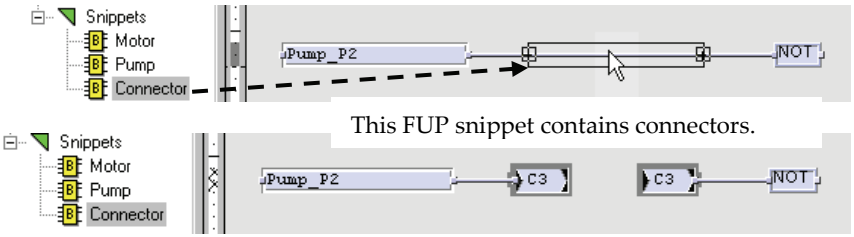
FUP Snippet? A FUP snippet is a function chart fragment that can be re-used. It consists of an arbitrary number of FBD elements.

A FUP snippet is quickly created: Just select the already set FBD elements, select *Save FUP Snippet as...* (from the context menu associated with the selection) and save the FUP snippet within the current project.





One more example:



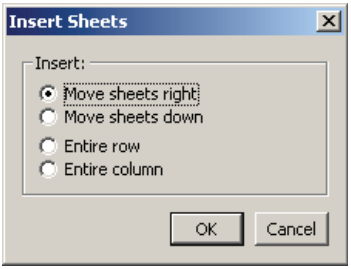
Note The ready FUP snippet can also be added using the *Insert FUP Snippet from...* command located on the the context menu associated with the drawing area.

5.16 Inserting/Deleting FBD Logic Sheets

Do the function blocks have numerous logic sheets and a new sheet must be inserted right in the middle or an existing sheet should be deleted? Do not bother to move the remaining sheets.

Here the solution: ELOP II provides the solution - the new command *Insert Sheets...*

1. Go to the sheet in the overview window where the new sheet should be inserted.
2. Right-click and select *Insert Sheets...* from the context menu.
3. Select one of the options in the dialog box to determine how to move the existing sheets.



These options should already be known from Microsoft ® Excel.

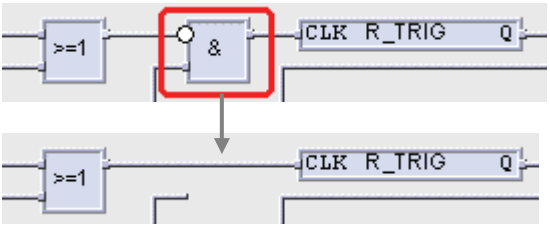
4. Click *OK*.
   Result: An empty sheet is inserted. Existing sheets are moved according to the selected option.

If sheets should be deleted, proceed as described above. But use the command *Delete Sheets...*

5.17 Deleting Function Blocks and Reconnecting Lines

The new command *Delete/Reconnect Lines* (located on the context menu associated with the function block) deletes the function block displayed in the drawing area and reconnects the connected input and output lines.

Example The AND function block is deleted with the *Delete/Reconnect Lines* command:



Please observe: The inverted node of the deleted function block was not adopted into the remaining logic.
The line between input IN1 and output OUT1 was reconnected. Reason: The base points were positioned on the same level and their data types were compatible.

Shortcut Point to block, click mouse button 1 and press CTRL+DEL.

5.18 Sorting Expanded Variables

How to sort structure declarations of variables in the Variable Declaration Editor? Very simple in ELOP II version 5.1:

1. Open the context menu over the column title *Name* (in the Variable Declaration Editor).
2. Point to *Sort*.
3. Select the sort order *Also Elements*.

Result: The structure elements are sorted according to the defined sort order. The column title displays (for ascending order) or (for descending order).

5.19 Fast, Central Definition of Snap Range

So far, the snap range could only be defined in the drawing area properties (tab *Miscellaneous*, *Snap Range*).
Now, ELOP II offers an option for fast, central definition:

1. Open the properties of Project Management (menu *Project*, *Properties*).
2. Go to the *More* tab.
3. Define the property `SensitivityDistance` and enter the required snap range (in pixel).
4. Click *OK* (in the *More* tab).
5. Open a project.

Result: The setting for the snap range is applied to all POUs of the project (if the *Include Parent Data* checkbox is ticked in the properties).
Alternatively: The property for a project, library or even a POU can be defined by the user (select *Properties* from the context menu).

What is the snap range?

The snap range is the area around an object in the drawing area of the FBD Editor.
As soon as the cursor is moved to the snap range, the next action is started with this object. The smaller the snap range, the closer the cursor must be moved to the addressed object. The larger the snap range, the larger the distance of the cursor to the addressed object.

CHAPTER 6

Additional Products

This chapter informs you about the new features of the additional products.

6 Additional Products

6.1 XML-Export/Import of Project Data

ELOP II offers an additional export/import option: **to** and **from** a text file in XML format

Here the details:

- During export, the ELOP II project data is saved as an XML file.
  This allows one to export any project data from ELOP II and to further process it with another system (e.g., to edit it with another text editor or to evaluate or modify it with familiar programs after appropriate transformation).
- During import, the XML file is converted to create ELOP II objects or even a complete ELOP II project.
  This allows one to import data, which has already been created in other applications, into ELOP II. Already exported ELOP II data can also be edited in a text editor and then re-import as new ELOP II object.

Advantages This export/import offers the advantages XML is known for:

– The ELOP II project data exist as hierarchically structured text files.
– XML is best suited for data exchange between different systems.

How to do export/ import? The project data is exported with **LCxmlExport** (optional tool). **LCxmlImport** (also optional) is used to import ELOP II project data.

Both tools must be started within an "ELOP II Command Prompt" (*Start*, *Programs*, *ELOP II*).

Optional? The optional tools **LCxmlExport** and **LCxmlImport** are add-ons to the standard delivery range of ELOP II.
Please contact HIMA for purchasing and using these optional tools.

Additional Info... On the tools can be found in the HTML manuals that are included within the scope of delivery.

1. Start the ELOP II Control Center.
2. Click Documentation and then Tools Documentation.

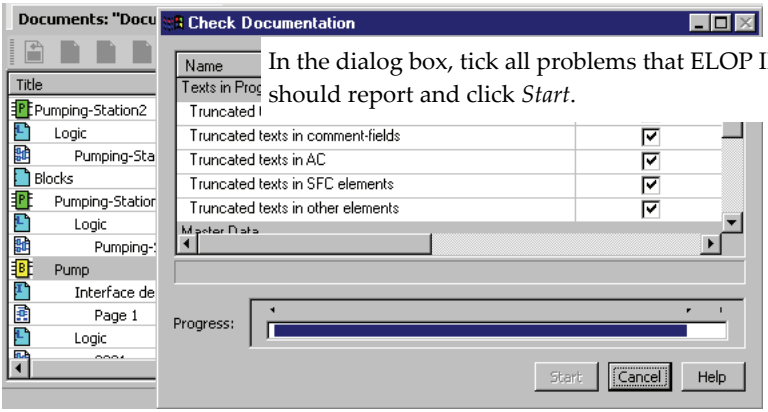
3. Search for the hyperlink for LCxmlExport or LCxmlImport and click it.

6.2 Finding Problems in the Documentation Printout

The function chart is ready and printed out, but the printout is not up to the expectations?

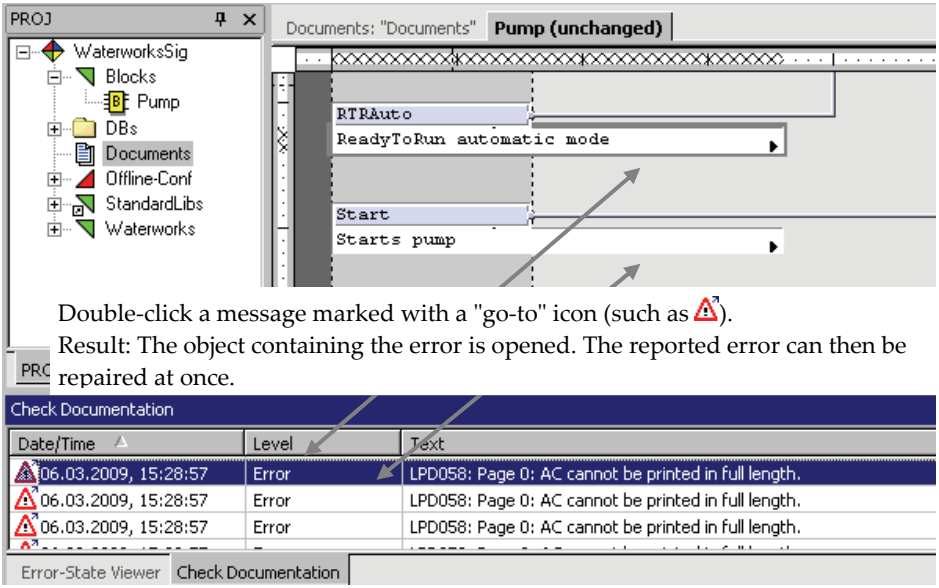
- Are printouts for certain objects missing?
  Possible cause: A printout form has been entered for the object, but this form does not exist.
- Do value fields or comment fields show black arrows (▶)?
  These arrows represent truncated texts.
  Possible causes: Some signal or long names have been modified belated. Or the signal has been set multiple times and now more sets of cross-reference information are output.
- Are texts missing in the printouts, but placeholders have been defined in the print forms?
  Possible cause: No master data definitions exist for the object to replace the placeholders. Or the definitions do exist, but no text has been entered.

The printout check of ELOP II detects such problems! Start the check in the Documentation Editor: Select the *Check Documentation* command from the context menu, then, e.g., the command *Check All…*



In the dialog box, tick all problems that ELOP II should report and click *Start*.

Result: The Status Viewer displays a new tab reporting the problems detected in the printout:

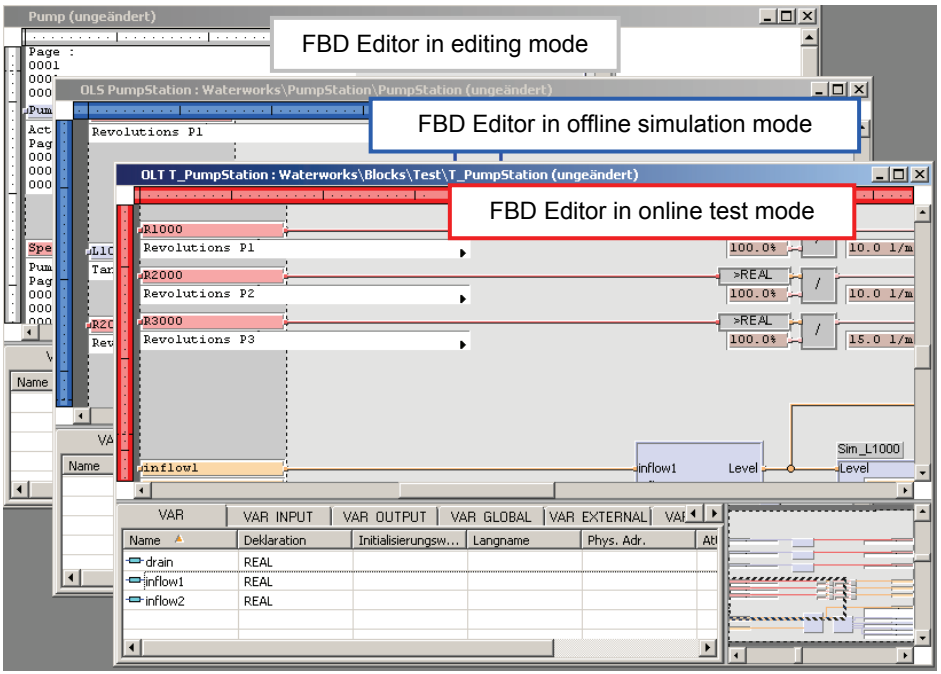


Double-click a message marked with a "go-to" icon (such as).

Result: The object containing the error is opened. The reported error can then be repaired at once.

6.3 Emphasizing the Online/Offline Editor Mode (Colored Ruler per Mode)

Are FBD Editors used in different modes at the same time?
Now the distinction between the different ELOP II editor modes can be made easily and quickly: The standard ELOP II configuration uses a different color for the ruler of each editor mode.



TIP User-specific colors for the ruler can also be defined for each editor mode. To do so, follow the instructions described in the online help (search for Ruler, Configuring the Colors).

6.4 Quick Start of Offline Simulation

If an offline simulation was started and closed for a resource at a previous stage, the *Offline Simulation Quick Start* command can now be

used to start the offline simulation more quickly. The offline simulation is started without performing a new code generation.
Please observe: The *Offline Simulation Quick Start* command is only available in the context menu associated with the resources.

6.5 Further Improvements for the Offline Simulation

The offline simulation of ELOP II version 5.1 offers these operating improvements:

6.5.1 Shortcuts for Resume and Single Stepping

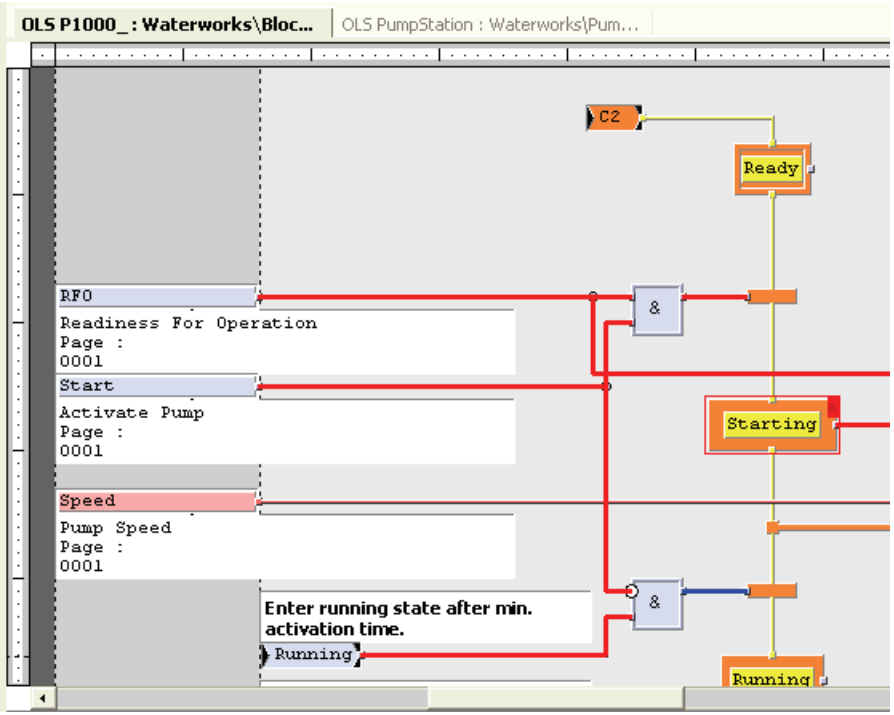
F5	Resume processing
F10	Single stepping: step over
F11	Single stepping: step into
SHIFT+F11	Single stepping: step out

6.5.2 Configurable OLS Structure Windows

The OLS structure window can be configured: Using the context menu, the displayed instance structure can refreshed and changed. All or only user-defined POUs can thus be displayed (by selecting *Show Standard POUs* and/or *Show Vendor POUs*).

6.5.3 Visualization of the Active SFC Step

The active step is highlighted in the processed logic by a red frame:



6.6 Analyzing the Logic Using logi.LINT Before Code Generation

Objective The additional product logi.LINT is used to perform a static code analysis on entire resources.

Findings:

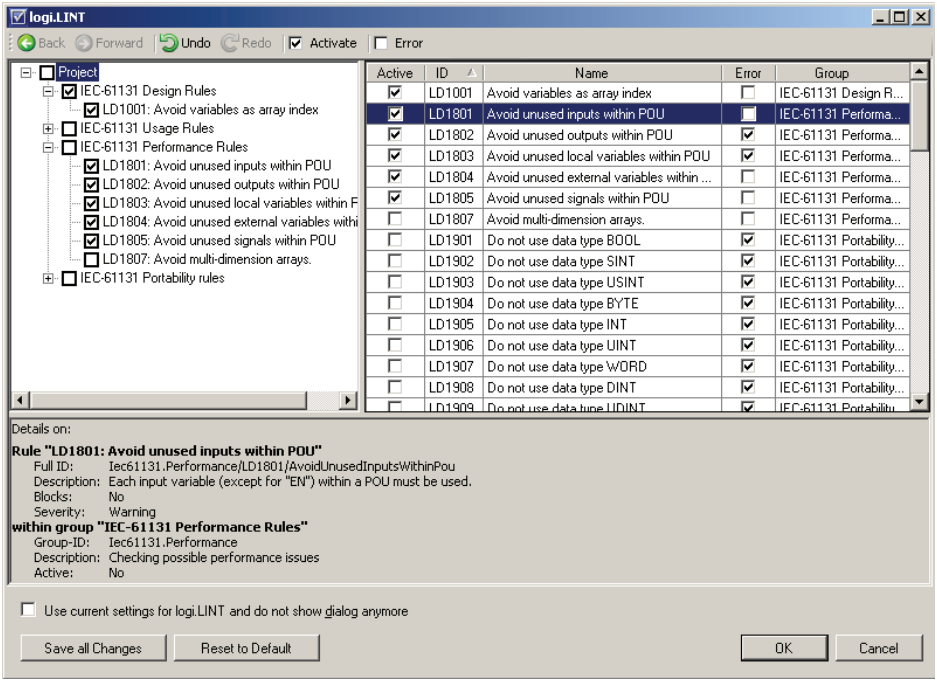
- Programming errors prior to compilation or code generation, such as not supported data types and POUs, not resolved external declarations and type conflicts.
- Typical programming errors with a correct syntax and therefore not detected during the compilation or code generation, such as multiple assignments to global/local variables, not connected inputs or outputs, open lines and open connectors.

- Starting points for optimizations, since logi.LINT is able to find unused variables, empty value fields or similar.

These checks are offered for the graphical languages FBD and SFC.

For ease of use, ELOP II automatically starts logi.LINT before the code generation. Perform the following steps:

1. Start the code generation: Right-click the resource and select *Code Generation* from the context menu
   Result: The *logi.LINT* dialog box is displayed.



2. Check the configuration of logi.LINT:

- Are all rules activated that should be checked?
- For which rules is there a check in the *Error* column? (Violations of these rules are treated as errors, meaning that the following action, e.g., the code generation, is not started. All other violations will be treated as warnings.)

3. Correct the settings in the dialog box as required and click *OK*. Result: The activated rules are checked. The Status Viewer reports which position in the programmed logic is violating a rule.
4. The code generation is only started if no rule violation is reported as error.
5. Check all violations of the rules:

- If the message is marked with a go-to icon (such as ) in the Status Viewer: Double-click the message to go to the position in the programmed logic.
- If not, go to the mentioned position within the programmed logic by other means. For instance, search for the POU using the *Search POU...* command

6. Fix the positions within the programmed logic according to the used guidelines.

Note logi.LINT can be configured in advance to use this configuration immediately for logi.LINT: Select *logi.LINT* and *Define Options....* from the context menu associated with the resource

If the *logi.LINT* dialog box should not be displayed, tick the *Use current settings for logi.LINT and do not show dialog anymore* option.

TIP logi.LINT can also be started without generating the code: Context menu associated with the resource, *logi.LINT* and *Start*

Additional info... Detailed instructions such as how to activate/deactivate rules, how to treat rule violations as errors or warnings, and other possible actions in the *logi.LINT* dialog box can be found in the online help (search for logi.LINT).

Note With projects that have been converted from a previous version, a new code must be generated with V5.1 prior to performing logiLINT. During this process, the code generator creates information required for the logiLINT analysis.

6.7 Force Markers List

The plug-in *Force Markers List* is no longer available in ELOP II V5.1!

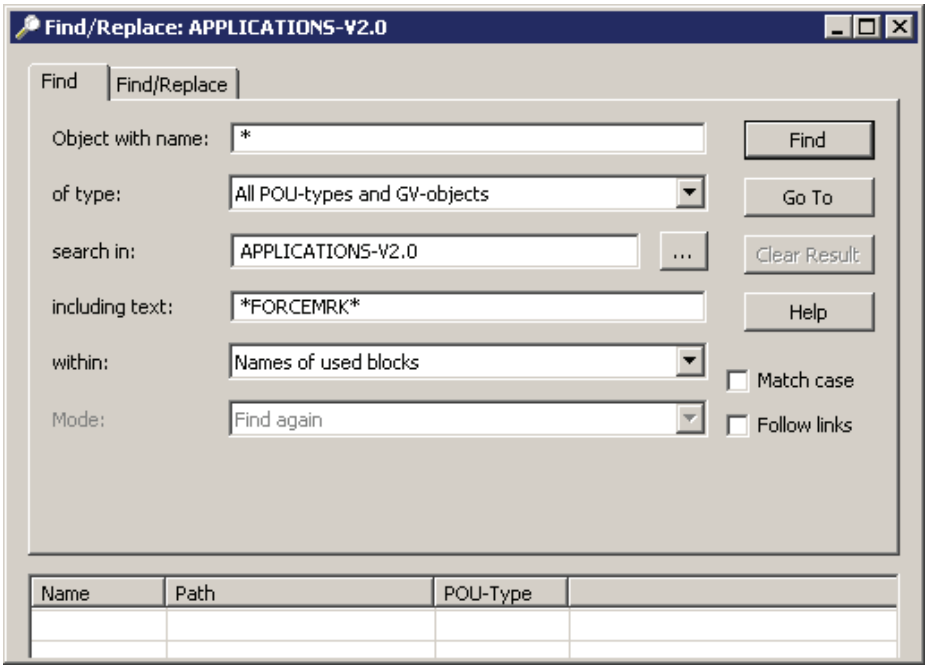
6.7.1 Consequences:

In accordance with the Safety Manual (HI 800 013), all force markers must be removed from the user program prior to starting safety-related operation or before an acceptance test is performed by a test institute!
If force markers were used in ELOP II V4.1, they are no longer available in the force markers list after converting the project to ELOP II V5.1.

6.7.2 Workaround

Using the Search/Replace function, the project can be checked for used force markers.
To determine whether force mark modules were used in a project, proceed as follows:

1. Open the ELOP II project.
2. Right-click the **Project** and select **Search/Replace...**
3. In the Search/Replace... dialog box, use the following settings to search for force markers:



Additional info... Double-click a search result to open the POU in which the force marker has been used. Use the plug-in POE Instance List within a POU to list all the instances of type ForceMark and to jump the corresponding position.

CHAPTER 7

Hardware-Specific Features

This chapter describes the new features available for the HIMA controllers of the H41/H51q system family.

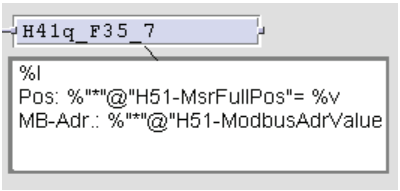
7 Hardware-Specific Features

7.1 Displaying HW-Attributes in OLS/OLT

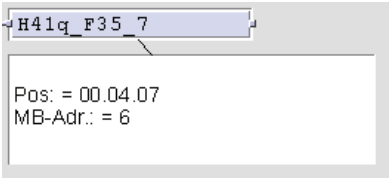
Hardware attributes (HW attributes) can now be displayed in the offline simulation (OLS) and in the online test (OLT).
With version 4.1, this was only possible in editing mode.

Here an overview of the required steps:

1. Open the POU in the FBD Editor (i.e., in editing mode).
2. Create an attached comment (AC) within the POU.
3. Enter the corresponding format strings for the controller properties in the attached comment:



4. Save the POU.
5. Start the offline simulation or the online test.
6. Open the POU in OLS or OLT mode.
   Result: The format strings are evaluated accordingly:



TIP The hardware attributes can also be displayed by default within an attached comment field. To do this, change the AC settings in the POU properties.

TIP The hardware attributes can also be displayed by default within the variable declaration. To do this, create user-defined columns and enter the corresponding format string as column content.

Additional info... On the format strings allowed for the controller properties can be found in the online help (search for Controller Properties, Format Strings).

7.2 Operating System Version in the Fault Diagnosis

Additionally, the system fault diagnosis contains the following information about the resource:

- The controller's operating system version
- Controller type

7.3 Documentation of the Loaded Project Provides Information about the Code Generation

The documentation of the loaded project provides information about the code generation. The code version of the loaded project is thus also documented.

7.4 SIL Value Displayed in the PES Properties

The EK value can be set in the PES properties. Additionally, ELOP II displays the corresponding SIL value according to the following assignment:

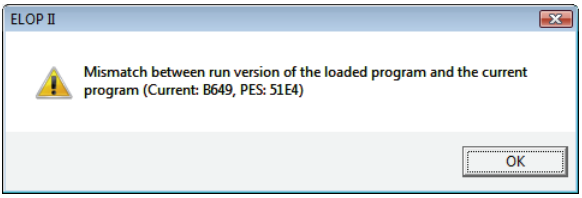
AK (configurable)	0 1 2 3 4 5 6
SIL (display only)	0 1 1 1 2 3 3

7.5 Improvements with Respect to Reload/Download

- When loading the user program into redundant central modules, ELOP II copies the reference data for the online test (OLT) function already after loading the first central module. This ensures that the OLT access to the loaded controller is ensured even if the load process of the central module was not successful.
- Improvement within the reload/download dialog box:
- The central modules in the MONO state are selected by default for not being loaded.
- After clicking the *Download* or *Reload* button, an additional prompt must be confirmed prior to starting the load process.

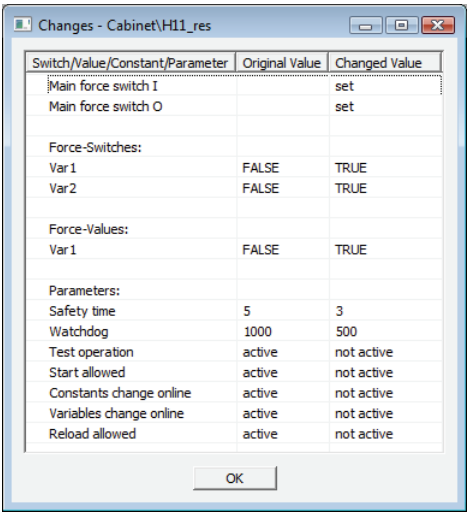
7.6 Visualization of Settings Changed Online

If system settings are changed online, the RUN version changes.



ELOP II displays the changed settings in a dialog box when opening or closing the online test and the Control Panel. The original and the modified value of the following settings is displayed:

- Constants (CONST)
- Force Switch
- Force Values
- Force Main Switch ON
- Force Main Switch OFF
- System parameter



Switch/Value/Constant/Parameter	Original Value	Changed Value
Main force switch I		set
Main force switch O		set
Force-Switches:		
Var1	FALSE	TRUE
Var2	FALSE	TRUE
Force-Values:		
Var1	FALSE	TRUE
Parameters:		
Safety time	5	3
Watchdog	1000	500
Test operation	active	not active
Start allowed	active	not active
Constants change online	active	not active
Variables change online	active	not active
Reload allowed	active	not active

HI 800 185 E

HIMA Paul Hildebrandt GmbH
Albert-Bassermann-Str. 28 | 68782 Brühl, Germany
Phone +49 6202 709-0 | Telefax +49 6202 709-107
info@hima.com | www.hima.com

SAFETY
NONSTOP

For a detailed list of all our subsidiaries and representatives, please visit our website: www.hima.com/contact