



Release Notes

Content

1	New 6 V48.SILworX.....	1
1.1	Compatibility with the PES Operating System.....	1
1.2	Compatibility with Existing Projects	2
1.3	Compatibility with the PC in Use.....	2
2	Improvements of V6.48.....	2
2.1	safeethernet, NSIP.....	4
3	Problems Corrected in V6.48.....	5
3.1	Variables	6
3.2	Online.....	6
3.3	FBD	7
3.4	safeethernet/NSIP.....	7
3.5	Hardware Editor	8
3.6	Protocols	9
4	Restrictions	10
4.1	Restrictions of V6.48.....	10
4.2	Basic Restrictions	12
4.2.1	General	12
4.2.2	Function Blocks and Functions	15
5	Upgrading from a Previous Version to V6.48	15
6	References.....	16

1 New 6 V48.SILworX

This document describes the improvements and new functions of V6.48 compared to the previous version:

- Chapters 2 and 3 describe the new functions and improvements.
- Chapter 4 specifies the current restrictions of V6.48.
- Chapter 5 describes the migration procedure from the previous version.

1.1 Compatibility with the PES Operating System

SILworX V6.48 can be used for the following HIMA system families:

- HIMax
- HIMatrix F systems
- HIMatrix M45

1.2 Compatibility with Existing Projects

The version can convert and edit projects created with a previous version. If a code generation is performed for the unchanged project, the existing CRC is maintained except for:

- X-OPC server, see Chapter 4.1, point 4
- X-OTS, see Chapter 4.1, point 4
- Projects in which SCF steps are used with Retain = TRUE, see Chapter 3, point 4
After converting the project, SILworX V6 sets the retain attribute of SFC steps to FALSE.
- Projects containing licenses for certain system capabilities, see Chapter 4.1, point 15.
- Projects containing redundancy groups from X-CI 24 51. [HE23091]

1.3 Compatibility with the PC in Use

The minimum requirements to the computer used for running SILworX are specified on the corresponding **installation DVD**.

In particular with very large projects, old PCs may require long processing times and thus be inappropriate for this task. Therefore, computers should as far as possible comply with the state of the art. Enhanced hardware features such as computing power and memory space ensure improved performance.

2 Improvements of V6.48

1 Support for HIMatrix M45

SILworX can be used for engineering controllers of the HIMatrix M45 family.

2 Support for HIMax X-CPU 31

When engineering HIMax controllers, processor modules X-CPU 31 can be used in rack 0.

3 Programming of function blocks and functions in structured text

Support for programming function blocks and functions in structured text.

4 Reload functionality

- Reload of the safe**ethernet** configuration.
- Reload of the configuration for alarms and events.
- Cold reload.

Modules that are not capable of reload, e.g., due to a previous firmware version, are first set to STOP during the reload procedure and then to RUN again. For communication modules, this process runs automatically. Processor modules and system bus modules must be stopped and restarted manually upon request from the reload process.

5 Improvements of the Hardware Editor

- HIMax racks can be located at any position. The rack IDs can also be freely defined.
- System variables are grouped in a specific tab of the rack detail view.
- Comment fields can be added.

SILworX Programming Tool V6.48 Release Notes

6 Improvements of online outputs

- Dialog boxes for command feedbacks include graphical symbols for success or failure.
- Extensive fault description.
- Option for copying diagnostic messages to the clipboard.
- Multiline diagnostic messages are possible.
- Filter criteria can be selected from a list.
- Reviewed message texts.
- To avoid operational errors, diagnostic messages can no longer be saved as CSV files.

7 Enhancements of graphic editors

- The zoom reduction factor was increased by a factor of 10.
More pages can thus be visualized at the minimum zoom level.
- If a selection rectangle is used in a graphic editor to select multiple objects, the selection only includes objects completely enclosed by the selection rectangle.
- More precise drag&drop functionality for graphic objects.
When dropping graphic objects onto a specific position, more attention is paid to the shape of the existing objects and of the objects to be inserted. As a result, objects can now be added closer to the existing objects or between them.

8 Improvements of the Global Variable Editor

- Global variables can be filtered based on their usage.
The usage of global variables can be displayed in additional columns that users can activate or deactivate. These columns can be used to filter variables.
- Columns containing the global variable names are wider.
- The *Search and Replace* dialog box is equipped with the **Help** button.
Click this button to open the corresponding online help.

9 File paths in the dialog box for selecting files remain selected when the same dialog box is open again

10 *Reading* is the setting used in the Alarms&Events Editor to indicate how global variables are accessed to.

The minus sign (-) was displayed in the previous version.

11 Performance enhancements

- Code generation.
- XML import.
- Restore.

12 The system variable *SB Essential Information* was renamed in *Responsible Module Essential*

This takes account of the fact that the X-CPU 31 can also be responsible and essential.

13 The minimum interval for executing the ComUserTask can be set to 2 ms.

14 Redundant power supply can be configured for HIMatrix F10 PCI 03

Redundant power supply via rail 1, 2 or both can be configured in the Hardware Editor for a resource of type HIMatrix F10 PCI 03.

- 15** The default value for the resource property *Max. Duration of Configuration Connections* is 12 ms

The default value was increased to ensure that no warnings are sent if the X-CPU 31 processor module is used.

2.1 safeethernet, NSIP

1 Reload code generation for safeethernet

In hardware capable of reload, a reload code generation can be performed for **safeethernet**: HIMax, HIMatrix PCI 10 03, F30 03, F31 03, F35 03, F60 CPU 03, HIMatrix M45.

Consequently, the *safeethernet CRC* configuration switch no longer exists.

2 New *Code Generation* parameter available for processor modules and communication modules

The detail view of the processor and communication modules that can also be engineered with previous SILworX versions is equipped with the new *Code Generation* parameter:

- If **Up to V6** is set, the code generation is performed in accordance with the previous versions to ensure that the CRC checksum is retained in a converted project.
- If **V6 and higher** is set, an optimized code generation supporting **safeethernet** reload is executed.

3 Cold reload

All the changes performed to a communication module can be applied by performing a reload. During the reload process, the module to be loaded stops running in system operation (cold) and restarts afterwards.

4 Redesigned **safeethernet** Editor and **safeethernet** online view

- Code generation can be set to **Up to V6** for converted projects and to **V6 and higher** to ensure support for reload.
- A resource can be selected as the *Timing Master*.
- The following editors can be opened by selecting the **Edit** context menu function (or **Detail View** in previous versions), or double-clicking:
 safeethernet Editor for connections between resources, or
 Hardware Editor, in the detail view of a remote I/O.
- In the online view, attribute names were changed, the arrangement of attributes was modified and the new *Version State* attribute was added to support the reload process.

5 The default value for *Behavior on CPU/COM Connection Loss* for the Modbus slave is **Retain Last Value**

This setting is more convenient if a cold reload is performed for the communication module.

3 Problems Corrected in V6.48

This chapter lists problems within previous versions that have been resolved in this version 6.48.

- 1 Code generation terminated if only the second OPC server of an OPC server set existed
If only the second OPC server existed within an OPC server set, the previous version terminated during code generation. [HE22259]
- 2 The code generator terminated in conjunction with an open Multitasking Editor
The previous version terminated if a code generation was started while the Multitasking Editor was open. [HE22316]
- 3 Connections and ENO no longer existed when updating conflicts
After a function block was moved to another location within the structure tree, the previous version reported that all references between interface variables and local/global variables were removed. If the function block was then updated, the connections and ENO were deleted. [HE22532]
- 4 TRUE *Retain* attribute for SFC steps
In the previous version, the *Retain* attribute could be set to TRUE for an SFC step. This is no longer possible in V6. SILworX V6 changes the attribute when converting a project. In doing so, when code is generated with SILworX V6, the CRC of <User Program Name>_retain.config is changed with respect to a previous code generation if the program <User Program Name> or a function block contains one or multiple SFC steps that have been previously declared as *Retain*. [HE23097]
- 5 The previous version terminated when trying to use the search and replace function within tables while the cell was in edit mode
The following sequence of actions is an example of how to cause the previous version to terminate:
 - Opening the *Search/Replace* dialog box in the Global Variable Editor.
 - Double-click any cell in the global variable table to set it to the edit mode.
 - Clicking the Search and Replace dialog box to exit the edit mode.
 - Using **Replace All** with settings ensuring successful completion of the action.
 This problem was removed in SILworX V6. [HE22543]
- 6 Reference of OLT field to deleted variables
In the previous version, the reference of an independent online test field to a variable sub-element could no longer be removed if the structure or array variable was deleted. The online test field had to be created again. [HE22627]
- 7 The previous version terminated when restoring a project with PROFINET
The previous version terminated when a project archive containing PROFINET was restored. [HE22265]
- 8 Hardlock license file is located in the proper directory
The license file required for activating the license via hardlock is located in the *IOLicense* directory of the USB stick used as hardlock. [HE22436]

3.1 Variables

- 1 When generating code, SILworX displays a warning if global retain variables are not maintained in the memory.
If there is no user program to read and write to a global variable with the *Retain* attribute, the variable loses its value during a warm start. If this occurs, SILworX V6 outputs a warning while generating the code. [HE14283]
- 2 Global variables with the *Const* attribute can be transmitted via **safeethernet** [HE17514, HE17847, HE13788]
- 3 Arrays displayed in the cross-reference list
If a local variable or a global variable used as VAR_EXTERNAL was composed of an array with additional sub-elements, the use of the first sub-element in the opened FBD Editor was not displayed in the cross-reference list. This problem was removed in SILworX V6. [HE22739]

3.2 Online

- 1 Online Goto function for global variables with target in function blocks located above the resource
In online mode, a Goto function can be applied to global variables having the Goto target (VAR_EXTERNAL) in the structure tree above the resource. The Goto target is located in function blocks contained in libraries within the Project or Configuration structure tree element. [HE22626]
- 2 Progress dialog box terminates if connection problems occur during an operating system download
If the connection between PADT and PES is lost during an operating system download, SILworX terminates the progress dialog box with an error message. [HE20037]
- 3 Error message appears in the dialog box when the **Set Responsible Attribute** command is being executed
When attempting to change the *Responsible* attribute while a system bus module is running in system operation, an error message appears not only in the logbook, but also in the dialog box. [HE22209]
- 4 Program ID integrated in command response dialog boxes and diagnostic texts
The user program ID is integrated in command response dialog boxes and diagnostic messages that refer to the user program. This enables easier allocation of the messages. [HE18246]
- 5 Program states deviating from the system state are displayed as system error or system warning
Program states deviating from the system state are displayed in the Control Panel as system error or system warning in accordance with their classification.
(The *System* LED is activated accordingly.) [HE21048]

3.3 FBD

- 1 Optimized representation of sequence changes in the version comparator
Compared to the previous version, changes to the processing sequence can thus be displayed in a different manner. [HE23283, HE 23763]
- 2 Current initial step name used to identify SFC networks in the version comparator
In the previous SILworX version, SFC networks were identified based on the initial step name used at its creation. It was therefore possible to display changes with a name that no longer exists in the current version.
In the following cases, the version comparator in SILworX V6 displays SFC networks as deleted in the previous version and as new in the current version:
 - The initial step was renamed after its creation.
 - The previous version was created with SILworX V5.
 - The current version was created with SILworX V6.
 - Changes were performed within the function block. [HE24104]
- 3 Updating an extendible function block not always possible
If the minimum extendibility of an extendible function block was increased and inputs/outputs were added, an existing function block instance could no longer be updated, if it was extended to a value less than the minimum extendibility set last. [HE21622]
- 4 Faulty error detection during changes within the function block type of extendible function blocks
If a function block instance was extended up to a certain point, no errors were reported during validation if additional inputs or outputs were added in the extended area of the corresponding function block type. [HE22309]
- 5 Changes to interface variables of functions/function blocks caused SILworX to terminate
SILworX could terminate if changes were performed to the interfaces of functions or function blocks and the instance(s) was(were) then updated, e.g., conversion of inputs to outputs or vice versa. [HE22582]
- 6 Cutting and pasting assigned comments or online test fields caused SILworX to terminate
The fault does not occur when simultaneously cutting and pasting a variable and an assigned comment or online test field. [HE21719]
- 7 Connections and ENO lost when updating conflicts
After a function block was moved to another location within the structure tree, the previous version reported that all references between interface variables and local/global variables were removed. If the function was then updated, the connections and ENO were deleted. [HE22532]

3.4 safeethernet/NSIP

- 1 Deleting **safeethernet** partners no longer results in deleted references of data types to system variables
In the previous version, if a partner was deleted during a **safeethernet** connection, the PADT also deleted the reference of the data types to the system variables. The verification returned 'No data type referenced.' [HE23602]

- 2 **Assign Fragments Automatically** for OPC connection caused SILworX to terminate
The previous version terminated when trying to execute **Assign Fragments Automatically** from within the detail view of a **safeethernet** connection. The **safeethernet** connection was defined between resource and X-OPC server. No variables were assigned. [HE22828]
- 3 Online view displays the *Time Stamp* in milliseconds resolution
In the previous version, the *time stamp* in the **safeethernet** online view (channel view) had a second, instead of millisecond accuracy. [HE23922]
- 4 Export for cross-project communication with *Fixed* profile caused SILworX to terminate
In the previous version, exporting a resource for cross-project communication with ELOP II Factory caused SILworX to terminate if the *Fixed* profile was selected. [HE21594]
- 5 Only export of default data types allowed for cross-project communication with ELOP II Factory
For export in connection with cross-project communication, the previous version also allowed data types not permitted in ELOP II Factory, in particular structures and arrays. [HE21029]
- 6 New **safeethernet** objects can be created at project and configuration level
In the previous version, it was not possible to create **safeethernet** objects at project and at configuration level. [HE22279]

3.5 Hardware Editor

- 1 Variable conflicts can be updated and resolved
In the previous version, when assigning global variables to system variables from within the detail views of the I/O modules, no menu function could be selected to update the global variables. As a consequence, conflicts resulting from actions such as deletion of used global variables could not be resolved. [HE22240]
- 2 No superfluous attributes during XML export
In the previous version, attributes only configurable via the redundancy group, were added to the XML export data of some modules. [HE21720]
- 3 The standard interface is copied when copying modules with IP connections
The value of the *Standard Interface* parameter is copied when copying modules with IP connections. The previous version used a random value for the copy. [HE22597]
- 4 Hardware online view update returns proper status
If both the Hardware Editor and its online view are opened in the OFFLINE state, a change in the Hardware Editor is adopted in the online view, but the displayed state does not change. In such a case, the previous version displayed the state ONLINE, although no connection existed to the PES. [HE21671]
- 5 Range of values for rack ID of remote I/Os changed to 1...1023
In SILworX V6.48, the rack IDs of remote I/Os can be set to a value included in 1...1023 (instead of 200...1023 in the previous version). [HE20561]

SILworX Programming Tool V6.48 Release Notes

- 6 SILworX no longer terminates in connection with the deletion of a redundancy group
The previous version terminated if a redundancy group was deleted repeatedly and the detail view of one of the affected modules was closed. [HE21860]
After a redundancy group was deleted, the previous version terminated if the image of the deleted redundancy group was right-clicked. [HE21740]
- 7 SILworX no longer terminates while closing a redundancy group
The previous version terminated when the following step sequence was performed while closing the project:
 - Create a new HIMax PES and drag any module above the rack.
 - Select the module, slightly move it and drop it again onto the same slot.
 - Save and close the Hardware Editor.After these steps, the Hardware Editor could no longer be opened, and SILworX terminated when closing the project. [HE23576]
- 8 Inputs of the X-DI 32 02 module represented in the Force Editor with the corresponding channel number
The Force Editor of the previous version did not specify the channel number of X-DI 32 02 modules, if the project was created in a version up to SILworX V2.6. [HE22218]
- 9 During the creation of X-MIO 7/6 01 modules the parameter *Supply Used* is set to the default value ON for all the DI channels
In the previous version, the *Supply Used* parameter had a random value. [HE22577]

3.6 Protocols

- 1 Modbus slave: the SRS of the redundant module is updated properly when simultaneously connected to HIMax and HIMatrix
The previous version did not update the SRS when switching between online views of the Modbus slave for a HIMax and a HIMatrix resource. [HE22331]
- 2 Host editor information relating to X-OPC/X-OTS is adopted in the documentation
The previous version did not adopt the host editor information for OPC and OTS into the documentation:
 - IP addresses
 - PADT port [HE23125]
- 3 Changeability of structures in X-OPC server improved
The changeability of the *EU Initial Value*, *EU Final Value*, *Default Display* properties of structures and structure elements was improved in the OPC server:
 - These properties cannot be changed for structures.
 - These properties can be changed for structure elements.
 - If *EU Initial Value*, *EU Final Value* or *Default Display* are defined for structure elements, additional properties are available in the X-OPC server.
- 4 Collision check in the PROFINET editor
SILworX outputs collision warnings if data structures overlap one another due to unfavorable offset specifications. [HE16763]

SILworX Programming Tool V6.48 Release Notes

- 5 Error message if the GSDML file name is invalid (PROFINET)
If the GSDML file name for PROFINET does not comply with the valid naming conventions, SILworX V6 rejects the file.
In such a case, the previous version terminated. [HE21522]
- 6 Generation of valid data values for iParameters (PROFIsafe)
For the bit data type, SILworX V6 generates valid data values for the iParameters. [HE21905]
- 7 **Goto...** in connection with PROFINET error messages jumps to intended target [HE22385]

4 Restrictions

When using SILworX V6.48, take the following restrictions into account.

If the following instructions are observed, the restrictions have no influence on safety and on the availability of the code generated for a controller.

4.1 Restrictions of V6.48

- 1 Progress dialog box in connection with actions applying to structured tree sub-elements
If an action such as a copy should apply to a structure tree element including its sub-elements, SILworX opens a progress dialog box. If errors occur during the action, the progress dialog box remains open even if the parameter *Automatically close the dialog upon success* is activated.
The dialog box offers the option to abort the running process. A running process is also aborted, if an error occurs. In both cases, however, actions that have already been executed cannot be undone.
- 2 Sequential function chart: No indication of deadlocks
Combined use of selection and simultaneous nodes causes deadlocks, i.e., undefined states in which either all steps or no steps are active. SILworX does not warn the users. [HE17716]
- 3 Value of global variables used as VAR_EXTERNAL is not displayed
If global variables with Struct or Array data type are used as VAR_EXTERNAL, the FBD Editor does not display for the sub-elements the information entered in the column *Initial Value, Description, Additional Comment and Technical Unit* [HE19688]
- 4 CRC change during OPC configuration
Generating a new code for X-OPC or X-OTS in a project converted from a previous version, the CRC of the `opc.conf` file changes.
- 5 Cross-project communication
Cross-project communication files between SILworX projects are now exchanged using the Archive and Restore functions.
Existing connections are converted to normal connections.
The assignment of global variables with other names to the transport variables is no longer supported during import (up to SILworX V4).

SILworX Programming Tool V6.48 Release Notes

Proxy resources remain available, their semantic remain the same (resources without code generation).

- 6** The MUL function block provides erroneous values in concomitance with the following circumstances:

- HIMatrix standard resource
- Data type LREAL
- One input has the value $\pm\infty$, the other input the value *nan* (not a number)

In this case the result is $-\infty$, and not *nan* as specified. [HE21924]

- 7** Unfavorable positioning of objects in the FBD Editor

Objects with open line ends can be moved by drag&drop or using hotkeys such that they appear as they were connected to other objects. Even if these objects are position directly behind other objects.

Workaround: The verification procedure detects open line ends. [HE24238]

- 8** Extending function block instances and open line ends

If the two following conditions are met, extending function block instances can cause SILworX to terminate:

- The unused connections resulting from the extension immediately dock to open line ends.
- Docking is not possible due to data type incompatibility.

In these cases, the function block can be damaged to an extent that a repair by HIMA may be necessary. [HE23898]

- 9** Conflict resulting from changing the constant attribute for global variables after their use

A conflict occurs during the code generation, if a global variable is used as VAR_EXTERNAL and is set from Constant to Changeable or vice versa, when a value is assigned to this VAR_EXTERNAL and the global variable is constant.

Workaround: Add the variable once again at all positions in which it is used. [HE24487]

- 10** Conflict icon remains visible, inspite of removed conflict

In the following cases, the conflict icon remains visible although the invalid action was canceled and the valid value displayed:

- Invalid name is entered for a variable.
- An existing sequence number is assigned to an interface variable.

Workaround: Start a verification or update process [HE24339]

- 11** Moving a rack above another difficult to pursue

In the Hardware Editor, if a new rack is moved above an existing rack while dragging it from the Object Panel onto the workspace, the existing rack appears in the foreground and the moved rack is represented in the background. It is therefore difficult to pursue the movement of the rack. [HE24401]

- 12** Copying obsolete online values

In the Force Editor and other force tables, online values can be copied to the clipboard. If values that were not located in the visible window are copied, the values may be obsolete. [HE23314]

SILworX Programming Tool V6.48 Release Notes

13 Adding the 64th M45 module can cause SILworX to terminate

The maximum number of modules permitted for the HIMatrix M45 system is 63, including the processor module. Trying to add an illicit 64th module can cause SILworX to terminate. [HE24522]

14 Under Windows XP, administrator rights are required for SILworX

Trying to start SILworX under Windows XP as standard or guest user causes SILworX to terminate.

Workaround: Start SILworX as administrator.

15 Licenses are sorted by names which may result in a changed CRC

During code generation, SILworX V6.48 and higher no longer stores the licenses sorted by entry order, but by names. This may result in a changed CRC.

Workaround: Use suitable names, ask for HIMA technical support.

16 Reload code generation terminates if global variables named ENO are used

If a global variable named ENO is assigned a value, e.g., through a physical input, the code generator terminates when generating reloadable code. [HE24642]

17 SILworX terminates when renaming safe**ethernet** connections to remote I/Os

In particular, this occurs if the renaming action is not saved and one tries to open the remote I/O detail view within the Hardware Editor using **Edit** or a double-click. [HE24815]

18 Inconsistent priorities of a safe**ethernet** connection result during a reload in a connection loss

For a safe**ethernet** connection, the priority of the direction (partner) can be set to different values. The different priorities result in differing safe**ethernet** signatures, which are saved in the configuration file. During a reload performed at a later point in time, this causes the connection to be lost. The problem occurs under the following conditions:

- There is a connection between controllers, but no OPC connection
- The *Codegen* parameter is set to *V6 and higher*. [HE24864]

4.2 Basic Restrictions

4.2.1 General

1 In the Hardware Editor, the scaling settings for an analog value are read as REAL

SILworX reads the values specified for the vertices of an analog value (at 4 mA and 20 mA) as REAL. They are, however, further processed as LREAL. LREAL can also be used in the user program. This restriction is only relevant with very large or very small vertex values. [HE16388, restriction]

2 Logic operations of BOOL variables having values that originate from external systems, can provide results that differ from those expected.

The cause is that the coding of BOOL values used in the third-party system deviates from the coding used in the HIMAx.

Two workarounds are possible:

- The external system only provides 0 for FALSE and 1 for TRUE.
- A correction circuit is implemented in the user program for all relevant BOOL variables to normalize the value to 0 or 1:

non-normalized variable -> AtoByte function block -> AtoBOOL function block ->
normalized variable [HE13042, restriction]

3 Impossible to save certain changes in a SILworX editor

After specific changes are made in an editor, the message 'Impossible to save changes' appears while attempting to save. After confirming the message, however, the changes are saved.

If the SILworX editor is then closed and re-opened, the message 'The required data is being processed' appears.

An example of changes in which this problem occurs is the cyclic renaming of variables (A => B, B => C, C => A).

Workaround: Avoid exchanging names.

If required, restart SILworX. [HE11613, Restriction]

4 Variations of the cycle time during LREAL calculations

The cycle times can strongly vary during calculations with variables of type LREAL. To measure the watchdog time, the cycle time must be determined under realistic conditions.

[HE12115, restriction]

5 Sequential function chart: Step-internal TON starts a cycle later than normal TON

A reload is performed and leads to the following changes:

- A new step is added and must be immediately active after the reload.
- A TON function block with the input set to TRUE is added.

Afterwards, the step-internal TON starts one cycle later than the TON function block in the program logic. [HE16288, restriction]

6 Use of hardlocks

Licensing SILworX using hardlocks (U3 USB sticks, standard USB sticks) is handled differently among the various operating systems:

- With Windows XP, administrator rights are required in the following cases:

- a** For installation.
 - b** For operation, if SILworX was licensed using U3 USB sticks.
The rights of a standard user are sufficient for operation, if SILworX was licensed using standard USB sticks.

- With Windows 7, administrator rights are required to perform the installation.
For operation, hardlocks can be used to license SILworX to all types of users.

Workaround for Windows XP: Use softlock licenses or other standard USB sticks.
[HE17056, restriction]

7 Value change for VAR_INPUT variables for user-defined function blocks

For user-defined function blocks, SILworX handles VAR_INPUT variables differently, depending on how the inputs are connected:

- If the inputs are wired with variables of a default data type, the value of the variable is transferred to a copy within the function block (call by value).
- If the inputs are connected to variables of a user-defined data type, a reference to the variable is transferred to the function block (call by reference).

Note: If the VAR_INPUT variable is a global variable, take into account that it may additionally be used as VAR_EXTERNAL in the called function block and may be modified. During the subsequent reading of the corresponding VAR_INPUT variable in the function block, value changes of VAR_EXTERNAL result in the following actions:

- For a user-defined data type, the new values are read.
- For an elementary data type, the previous values are read. [HE17740, restriction]

SILworX Programming Tool V6.48 Release Notes

- 8** If the diagnostic view is opened during a system login and the connection is closed, SILworX offers the module login when attempting to re-establish the connection. [HE11926, restriction]
- 9** Online help associated with a POU not printable
The document management cannot print the content of the online help associated with a user-defined POU.
Workaround: Use Windows to display the online help content and print out the individual topics. [HE14244]
- 10** Various elements of a structure variable cannot be written simultaneously from different sources
The user program and the hardware or communication cannot simultaneously write to two different elements of the same structure variable.
Workaround: Use different structure variables for those elements written to by the user program and for those elements written to by the hardware or communication. [HE15700]
- 11** Elements of structure variables used as index
It is not possible to use elements of variables with structure data type as array index. [HE16159]
- 12** Value of user program's system variables during the online test and offline simulation
The value of user program's system variables is not displayed during the online test and offline simulation:
- The OLT field is empty.
 - The value of digital system variables is not represented by the color of the corresponding line.
 - The Process Value column in the System Variables tab of the Object Panel is empty.
 - The Force Editor contains no system variables
- Workaround: Most of the information is displayed elsewhere, e.g., in the Control Panel. To display it in the OLT, connect the system variable to a global variable (VAR_TEMP) and connect this to an OLT field. Forcing can only be performed in HIMax, if the program is connected to the system variable via a variable. The variable can be forced. [HE15396, restriction]
- 13** Import of export files from a previous version
It cannot be ensured that key terms in the export or import files (.CSV, .XML) do not change between SILworX versions. If this occurs, SILworX imports the corresponding data as default values, and outputs an error message.
Example: In versions up to V5.xx, is the data type for the **English** language setting is defined as 'Data Type', and 'Data type' with V5.xx and higher. When an export file is imported from a version up to V5.xx, SILworX creates all the variables with the default data type BOOL.
Workaround: Adjust the corresponding key words in the file to be imported. [HE21691]
- 14** Misleading indication of the force status for local forcing in connection with HIMatrix standard systems
For HIMatrix standard systems, the parameters indicating the status of local forcing (located above the force table) are set to regular values as if the information was actually available. In particular, these parameters are 'Force State', 'Forced Variables', 'Remaining Force Duration' and 'Force Time Reaction'. [HE23021]

4.2.2 Function Blocks and Functions

- 1 DIV_TIME with REAL typecast reports an error on ENO for divisor := +/-INF
The DIV_TIME function from the standard library improperly sets the ENO error output ENO to FALSE and reports therefore an error under the following conditions:
 - The IN2 input (divisor) is of type REAL.
 - The value of IN2 is +/-INF. [HE15199, Restriction]
- 2 ENO output in connection with user-defined function blocks may be overwritten during reload
With user-defined function blocks, in which the ENO output only depends on the EN input, ENO may be set to FALSE during a reload. Such function blocks do not themselves write to ENO. [HE19129]
- 3 Timer function blocks cannot be correctly used with the retain attribute
If a timer function block is used with the retain attribute, the time counter may adopt any potential value after a warm start.
Workaround: Do not use timer function blocks with the retain attribute. [HE17252]
- 4 SILworX terminates if two variables have the same name
If a variable is created in a function or function block and has the same name as an existing variable, SILworX terminates during the code generation. This may happen, e.g., if a variable is created and named ENO; the variable would then exist twice due to the predefined ENO variable. [HE24641]
- 5 The number of instances of function blocks restricts its reloadability
If the user program has a very nested structure, the maximum number of operations necessary to perform a reload may be exceeded.
Only resources with 21845 or less instances can be reloaded. Depending on its structure, a user program may not be capable of reload in connection with a significant lower number. [HE23791]

5 Upgrading from a Previous Version to V6.48

Project data from previous versions can continue to be used in V6.48.

No CRC changes occur as long as the **minimum configuration version** setting remains unchanged for a resource and none of the cases described in Chapter 1.2 has occurred. SILworX maintains the CRCs compatible provided that no changes occur or no new features are used.

Observe the following procedure to upgrade from V2.36 to V6.48:

- Generate code all resources prior to conversion. This allows potential deviations after the conversion to be detected during generation.
- Prior to converting the project, save it, e.g., on a removable medium.
- Open the project in V6.48 and convert it.
- Since the conversion is extensive, check the project integrity after completing the conversion.
- Generate the code in V48 to detect potential errors and check if CRCs has changed.
- Remove detected errors and re-generate the code to detect changed CRCs.
- If no CRC changes are detected, the migration was completed successfully.
- If CRC changes are detected, verify if they can be accepted.
- If the changes can be accepted, the migration is successfully completed.

SILworX Programming Tool V6.48 Release Notes

- If they cannot be accepted, continue to work with corresponding previous version.

Conversion Notes:

- The procedure to convert up to V2.36 is described in the release notes to V2.36.
- For very large projects, the conversion can take several hours.

6 References

- SILworX first steps manual V6, HI 801 103 E
- SILworX communication manual V6, HI 801 101 E