SAFETY NONSTOP

Release Notes

Content

1 New SILworX Version 5.30

This document describes the new functions of version 5.30, its restrictions and improvements compared to previous versions:

- Chapters 2 and 3 describe the new functions and improvements.
- Chapter 4 specifies the current restrictions of version 4.116.
- Chapter 5 describes the upgrade procedure from a previous version.

1.1 Compatibility with the PES Operating System

Version 5.30 can be used for the following PES with the mentioned operating system versions:

- HIMax controllers, firmware version 2.0 and beyond.
- HIMatrix controllers standard, CPU operating system version 7.0 and beyond, and COM operating system version 12.0 and beyond.
- HIMatrix controllers F10 PCI 03, F30 03, F31 03, F35 03 and F60 CPU 03, CPU operating system version 8.0 and beyond, and COM operating system version 13.0 and beyond.

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
- X-OTS

Export files (.CSV, .XML) created with a previous version might be not completely imported, see Chapter 4.2.1, Point 13.

1.3 Compatibility with the PC in Use

The minimum requirements for the PC used to operate SILworX are specified in the corresponding installation DVD. In particular with very large projects, old PCs may require long processing times and thus be inappropriate for this task. As far as possible, computer hardware should therefore comply with the state of the art. Enhanced hardware features such as computing power and memory space ensure improved performance.

2 New Features of Version 5.30 Compared to Version 4.116

This chapter describes the enhancements and extensions of version 5.30 compared to version 4.116.

1 Project archiving

- A project can be archived and restored from the archive. A project archive is stored in a special format (.PA3).
- Archives are automatically reorganized during creation so that the minimum memory space is required.

2 Auto archiving during reload/download

During download or reload, the current project can be archived in a user-defined directory. This ensures that product data is backed up after every loading procedure and prevents data from being lost due to operating errors. A history of all project changes can also be created automatically. A document detailing the differences between project versions can be issued using the version comparator.

3 Selecting Reload/Download in a dialog box

Reload and Download are combined in a single dialog box. The dialog box displays the last loaded code as well as the most recently generated code, and provides the possible loading options. Reload is the default setting for code generation and within the load dialog box (and in this particular case, only if the code is reloadable).

4 Dual code generation with automatic code comparison

Dual code generation can be selected during code generation. In this case, the code is generated two times and automatically compared to exclude faults caused by the hardware or by the PC operating system.

If a fault results from the comparison, no code is generated.

5 Coping logic networks and pasting them as typicals

Logic networks copied in the FBD Editor can be added as typicals. All the copied variables can be renamed in a dialog box. Names can be changed using table-specific functions such as Search and Replace, or Import.

6 Creating and using typical function blocks

In addition to standard function blocks and functions, libraries can also contain typicals. A typical can include any type of FBD logic networks. Adding a typical by drag&drop, also adds the logic contained therein. The existing names can be modified, as when the menu function **Typical Paste** is used.

7 Watchpages

Watchpages are available online for global or local variables. All the variables to be displayed in the watchpage can be selected by drag&drop. The Watchpage functionality allows users to edit force data, perform transfer procedures to the controller, store and load force entries into the project and import or export the selected variables.

8 Extension of the version comparator

- The I/O comparator can be used for all HIMax modules.
- Comparator is available for the CPU configuration.
- Comparator is available for safe**ethernet** configuration.

9 FBD Editor

Lines can be reconnected in the FBD logic plan. To do so, use Shift+Click to select the line endpoint of a connected pin and connect it to another pin of the same type by drag&drop.

10 Data type editor enhanced with table functions:

Search and Replace, Import and Export, Copy and Paste, Filter and further functions to ensure consistent functionality within all the tables.

11 **Go to** improvements, also for online **Go to**

Go to function from version comparator to Hardware Editor improved

12 Enhanced performance

- When handling user-defined data types
- When connecting references
- Within tables

13 Improvements within tables:

- Search and Replace
- Import and Export
- Copy and Paste
- Filter function
- The last table column is no longer stretched to the table margin. The space on the right hand-side of the last column remains empty.
- Sorting arrays

14 Configuration of HART over IP protocol for HIMax X-HART 32 01

2.1 Support for New HIMax I/O Modules

SILworX supports the planning of HIMax systems with the new I/O modules:

- HART module: X-HART 32 01
- Overspeed trip module: X-MIO 7/6 01

For more details, refer to the manuals HI 801 307 E (HART module) and HI 801 305 E (overspeed trip module).

2.2 New System Variable for HIMax

A new system variable *SB Essential Information* is now supported for the HIMax system. This system variable provides information on the state of the system bus modules to the user program.

3 Improvements of Version 5.30

This chapter lists problems within previous versions that have been resolved in version 5.30.

3.1 General

1 Global variable warning symbols in the online cross-references

SILworX 4.116 did not display the exclamation point in the online cross-references after the initial value of a global variable was changed. [HE20311]

2 Generating new names while copying objects

While copying objects in the FBD Editor, other editors or structure tree, "_1" is no longer suffixed to the names of copies, but the existing suffixes are increased by one. [HE15910]

3 Minimum memory space required for archives

Archives are automatically reorganized during creation so that the minimum memory space is required. [HE17320]

4 **Connecting references** also successful for very large projects

The memory space required for the **Connecting References** function was reduced. The function can now be successfully performed for very large projects. [HE19271, HE20988]

5 Corrupted projects can be converted to version 5

V3 projects, which were corrupted after copying a value field with OLT and comment fields [HE17261], can be converted to V5. The corrupted function blocks can then be deleted. [HE21372, HE21405]

6 Changed behavior of dialog boxes with progress bar

When automatically closed, the behavior of most dialog boxes with progress bar has changed. After the action required in the dialog box was successfully completed, the dialog box remains visible for a second before closing automatically. [HE15002, HE19790]

7 Changed default value for *Target Cycle Time Mode*

Whenever a new resource is created, the *Target Cycle Time Mode* parameter is now set to the default value *Fixed-tolerant*. [HE21581]

8 Revised representation of X-AI 32 51 inputs in Force Editor

Inputs are represented as AIs. [HE19577]

9 System parameter *Main Enable* renamed to *Allow Online Settings.* [HE19268]

10 I*/O Faults* renamed to *Field Faults*. [HE20991]

11 The data type editor supports the CSV import and CSV export for user-defined data types. [HE19088]

12 The ComUserTask configuration files are properly taken into account by the online version comparator.

In the previous version, the ComUserTask configuration files caused the online version comparator to display a different CRC. [HE21134]

13 Correct maximum values in the user program's detailed view during online version comparison

In the user program's detailed view, SILworX also displays correct maximum values for resources of type HIMatrix F30 03, F31 03, F35 03, F60 CPU 03, F10 PCI 03 during the online version comparison. [HE 20137]

14 Online diagnostic file contains the system statuses

In the previous version, the system statuses were not contained in the diagnostic file. [HE13345, HE19901]

15 Underrunning the value recommended for *Max.Com.Time Slice ASYNC [ms]* results in a warning message instead of an informational message [HE21312]

16 The *Process Data Refresh Rate* for ComUserTask can be edited.

In the previous version, this parameter could not be changed and was set to 0.

3.2 Function Block Diagrams

1 No termination after language change

After the language was changed, the previous version terminated when action blocks were copied. [HE21163]

2 Force icon represented at the transition input

When forcing a transition input in the previous version, the force icon was not displayed in the graphical online test since the graphical representation of the transition input was not refreshed during forcing. The force icon only appeared when the transition input was displayed in the Force Editor. In the same way, the force icon remained visible when forcing was stopped. [HE21345]

3 SILworX terminated when using output structure variables

SILworX allows one to reference the output parameter A of a function block instance X in the format X.A.

Attempting to reference all output parameters as structure X by using a text entryin the value field caused SILworX 4.116 to terminate. [HE20256]

4 Representation of the execution order

The execution order displayed in the FBD Editor was adjusted to that displayed in the version comparator. [HE20029]

3.3 Modbus Save

1 Addresses of register variables for Modbus slave

The addresses of register variables in the Modbus slave must be comply with the register limits, i.e., the number of bytes must be divisible by 2 without remainder. The SILworX code generation adds additional bytes, if required. [HE17253]

2 SILworX always displays the Modbus register address as double-digit bit position [HE20057]

3.4 **safe**ethernet Applications

1 With X-OPC and X-OTS, all the system variables are available at the OPC interface

System variables such as *Force Deactivation*, *Read-Only in RUN* and *Reload Deactivation* are now available at the OPC interface, even with read-only access. [HE18378]

2 A new global variable immediately appears in the safe**ethernet** Editor

In the previous version, the safe**ethernet** Editor had to be reopened. [HE20677]

3.5 Document Management

1 Rack ID and slot number specified in the hardware documentation of redundancy groups. [HE19191]

2 Hardware documentation of HIMatrix system is marked in yellow [HE19915]

3 Hardware documentation of HIMax systems features improved rack alignment with respect to the printed pages.

For racks with 18 slots, the page margin is no longer located within a module. [HE10083]

3.6 Hardware Editor

1 In the Hardware Editor, the keyboard can be used for copy and paste actions. [HE16444]

2 If no processor module is specified, the code generation provides a warning message. [HE11495]

3 Parameter setting of power rails for HIMatrix:

- The parameter setting for F30 03, F31 03, F35 03 and F60 03 was deleted.
- With F10 PCI 03, the parameters can be set for both power rails. [HE19986]

4 Improved selection of operating mode for X-AI 16 51. [HE20067]

3.7 Protocols

1 The time unit for TCP/SR *Keep Alive Interval* changed to seconds.

In the previous version, the time was specified in *[ms]*. [HE21385]

4 Restrictions

When using SILworX version 5.30, 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 Version 5.30

1 Sequential function chart: No indication of deadlocks

The combined use of selection and simultaneous nodes results in undefined states, in which either all steps are active or none of them (deadlocks). SILworX does not warn the users. [HE17716]

2 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 sub-elements the values entered in the columns *Initial Value, Description, Additional Comment* and *Technical Unit* [HE19688]

3 Cutting and pasting assigned comments or online test fields causes SILworX to terminate..

The fault does not occur when cutting and pasting assigned comments or online test fields together with a variable. [HE21719]

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 procedures.

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

6 The MUL function block provides erroneous values, in coincidence with the following circumstances:

- HIMatrix standard resource
- Data type LREAL
- One input has the value +/-∞, the other input the value *nan* (not a number)

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

7 Communication is possible from OTS to OPC server, not in the opposite direction.

Communication between OTS and OPC server is only possible from OTS to OPC. If the OTS resource contains global variables, which read information from the connection with the OPC server, the code generation is aborted with an error message. [HE22246]

8 Code generation terminates if only the second OPC server of an OPC server set exists

If only the second OPC server exists within an OPC server set, SILworX terminates during code generation.

Workaround: Replace the two OPC servers in the safe**ethernet** Editor. This ensures that the CRC checksum from version 3 is maintained. [HE22259]

9 The XML files that the Hardware Editor of SILworX V.5.30 reads from during import and writes to during export, are not compatible with the corresponding XML data from the previous version.

4.2 Basic Restrictions

4.2.1 General

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

SILworX reads the values specified for the base points of an analog value as REAL (at 4 mA and 20 mA). 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 base point values. [HE16388, restriction]

2 Logic operations of BOOL variables having values that originate from third-party 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 third-party 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]

SILworX programming tool V.5.30

3 Saving changes in a SILworX Editor is not possible

After specific changes are made within 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 reopened, the message "The required data is being processed" appears.

Examples of changes in which this problem occurs are the cyclic renaming of variables (A → B, B → C, C → A).

Workaround: Avoid renaming variables.

If required, restart SILworX. [HE11613, Restriction]

4 Variations of the cycle time during LREAL calculations.

The cycle times can vary strongly 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 active immediately after the reload.
- A TON function block with the input set to TRUE is added.

The step-internal TON then 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 managed 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 standard USB sticks. [HE17056, Restriction]

7 Handling of VAR_INPUT variables in connection with user-defined function blocks

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

- If the inputs are connected to variables of a default data type, SILworX transfers the value of the global variable to a copy within the function block (call by value). Changes to the original variables do not affect this copy (writing to the corresponding VAR_EXTERNAL). The VAR_INPUT retains its value.
- If the inputs are connected to variables of user-defined data types, SILworX transfers the reference to the variable (call by reference).

  If the variable is a global variable, take into account that it can be modified in the function block. Additionally, a global variable can be used in a function block as VAR_EXTERNAL. Changes to such variables (writing to the VAR_EXTERNAL) are effective immediately when the corresponding VAR_INPUT variable is read in the function block. As such, the value of the VAR_INPUT can change while executing the function block. [HE17740, restriction]

SILworX programming tool V.5.30

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 variables.

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 system variables during the online test and offline simulation

The value of 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 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 of a previous version

If the data type column cannot be found in the file to be imported (.CVS, .XML), all variables are created with the default data type BOOL.

Workaround: Rename the Data Type column in the file to be imported to Data type. [HE21691]

4.2.2 Functions and Function Blocks

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 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 is only dependent 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]

5 Upgrading from one Previous Version to Version. 5.30

Project data from previous versions can continue to be used in V.5.30.

No CRC changes occur as long as the **minimum configuration version** setting of a resource remains unchanged. SILworX maintains the CRCs compatible provided that no changes occur or no new features are used.

Observe the following procedure to upgrade from version 2.36 to version 5.30:

- Generate code for all resources prior to conversion. This allows potential deviations after the conversion to be detected during generation.
- Prior to converting, save the project, e.g., on a removable medium.
- Open the project in version 5.30 and convert it.
- Since the conversion is extensive, check the project integrity after completing the conversion.
- Generate the code in version 5.30 to detect potential errors and check if CRCs 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 whether they can be accepted.
- If this is the case, the migration was completed successfully.
- If they cannot be accepted, continue to work with the corresponding previous version.

Conversion Notes

- The procedure to convert versions prior to 2.36 is described in the release notes to version 2.36.
- For very large projects, the conversion can take several hours.