



## Content

|     |   |   |
|-----|---|---|
| 1   | New SILworX Version .....                                 | 1 |
| 2   | Remarks about Version 2.52.....                           | 1 |
| 2.1 | Support of 3 GB Memory .....                              | 1 |
| 2.2 | Notes for Using the Additional Memory .....               | 2 |
| 2.3 | Improvements in the FBD Editor .....                      | 2 |
| 3   | Restrictions of the Version 2.52 .....                    | 3 |
| 3.1 | Restrictions with Respect to the FBD Editor .....         | 3 |
| 3.2 | Restrictions with Respect to Online View and Test .....   | 4 |
| 3.3 | Restrictions with Respect to OPC Configuration .....      | 4 |
| 3.4 | Further Restrictions.....                                 | 4 |
| 4   | Migration from the Previous Version to Version 2.52 ..... | 7 |

## 1 New SILworX Version

- Version **2.52** for:
  - HIMax controllers, firmware version 2.0 and beyond.

This version is not applicable for HIMatrix controllers!

This document describes the new functions of version 2.52, its restrictions, and its improvements compared to the previous version:

- Chapter 2 contains the new functions.
- Chapter 3 specifies the current restrictions of version 2.52.
- Chapter 4 describes the migration procedure from the previous version.

## 2 Remarks about Version 2.52

Refer to the release notes for versions 1.12 and 1.20 for more information on the extensions and improvements of version 2.52 compared to the previous versions 1.12 and 1.20. The migration procedure from the previous versions 1.12 and V.1.20 to version 2.52 is identical with that specified in the release notes for versions 2.36 and 2.46.

### 2.1 Support of 3 GB Memory

Version 2.52 can manage 3 GB virtual memory. This version is thus able to compile and to print extensive projects. Previously, version 2.46 was terminated when attempting to compile or to print extensive projects and the error message "Out of memory" or other concomitants appeared.

To be able to compile and print extensive projects, however, the 32-bit versions of the operating systems 'Windows XP' and 'Vista' must be configured such that they provide SILworX with up to 3 GB of virtual memory.

Windows XP:

- Add the option "/3GB" to the row containing "WINDOWS" in the boot.ini file located under "[operating systems]".

Windows Vista:

- Edit "Boot Configuration Data (BCD)":  
BCDEDIT /Set increaseUserVa 3072

### **2.2 Notes for Using the Additional Memory**

Increasing the memory for SILworX reduces the memory available to the operating system. This can prevent Windows from loading some drivers or even from starting.

For this reason, the PC in use should preferably only contain devices that can be used with standard drivers.

As few programs as possible should simultaneously run with SILworX. These programs should be low memory intensive.

If such problems occur, SILworX can be provided with less than 3 GB memory.

Windows XP:

- Add the option "/UserVa=**x**" (with  $2048 < x < 3072$ ) to the row containing "WINDOWS" in the boot.ini file located under "[operating systems]".

Windows Vista:

- Edit "Boot Configuration Data (BCD)":  
BCDEDIT /Set increaseUserVa **x**  
with  $2048 < x < 3072$ .

### **2.3 Improvements in the FBD Editor**

- The FBD Editor opens much faster.
- It is now possible to open more editors than before. How many editors are opened depends on the data that they contain. The same applies when up to 2 GB of memory are used.

## 3 Restrictions of the Version 2.52

*When using SILworX, version 2.52, 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 HIMax controller.*

### 3.1 Restrictions with Respect to the FBD Editor

- The Retain property can be removed from function block instances.  
If the interface variables of type VAR\_OUT of a function block are set to Retain = TRUE, they can be set to Retain = FALSE for the function block instances.  
This does not affect the generated code. [HE14182]
- The processing sequence view is not refreshed.  
If the logics is changed in the Function Block Editor, the processing sequence is displayed as before the change. [HE13841]
- Priorities of branches in sequential function charts (SFC) are not verified.  
For an SFC branch (two transitions after a step), the following conditions must apply:
  - The priority of all transitions must either be set manually or automatically.
  - These priorities must be different.

These conditions are not checked during compilation. [HE13911]
- Logic operations of BOOL variables having values that do not result from safety-related communication, can provide results that differ from those expected.  
The cause is that the coding of BOOL values used in the external system deviates from the coding used in the HIMax.  
Two workarounds are possible:
  - The external system only transmits "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].
- Internal value fields do not adopt the entered value.  
Internal value fields of function blocks do not adopt the entered value, but the initial value specified in the type definition.  
Workaround: Use value fields instead of internal value fields. [HE15544]
- Invisible feed-back loops.  
Directly connecting an output to an input creates an invisible feedback loop, if SILworX was not able to find any suitable routing. For instance, if an online test field was added, drawing a connection line between the two connectors can create an invisible connection line. The connection only becomes visible after deleting the variable. [HE15365]

### 3.2 Restrictions with Respect to Online View and Test

- Reconnecting the diagnostic view causes a module login, instead of a system login.  
If the diagnostic view is opened during a system login and the connection is closed, the module login is offered when attempting to re-establish the connection.  
Workaround: Establish the connections to the system either using the Hardware Online View or the Control Panel.  
If the module login was opened, close all views of the module: The diagnostic and the module view. [HE11926]
- SILworX terminates during "Writing by MAC".  
SILworX terminates if the MAC address of a device not connected to the network was entered in the "Writing by MAC" dialog box. [HE14448]

### 3.3 Restrictions with Respect to OPC Configuration

- More than four OPC servers can be configured.  
It is possible to configure more than four OPC servers for alarms and events, and to compile the project successfully. Four of those OPC servers can safely connect to the resource, the remaining, however, have connection problems.  
Workaround: Make sure (manually) that not more than four OPC servers are configured. [HE14543]
- Settings for **safeethernet** of an OPC server set result in inconsistent configuration.  
If the **safeethernet** parameters "Event Priority" and "Condition Value Priority" are modified when configuring the OPC server, the configuration is no longer consistent.  
Workaround: The parameters "Event Priority" and "Condition Value Priority" must be identical. Recommendation: retain the default value 1!
- Global variables can be deleted in the Object Panel  
Global variables can be deleted from the Object Panel of the **safeethernet** Editor for OPC server sets, by pressing the delete key and clicking "OK" to confirm. These variables are deleted from the project and must be re-created, if required.  
Workaround: After accidentally pressing the Delete key, do not confirm the dialog box.
- Code generation for OPC server terminates.  
The code generation for the OPC server terminates if more than a view is used.  
Workaround: Set the parameter "safeethernet CRC" located in the resource properties to "compatible to V.2.36". [HE15331]

### 3.4 Further Restrictions

- Is it impossible to save changes in an Editor?  
After specific changes made within an Editor, the message "Impossible to save changes" appears while attempting to save. After confirming the message, however, the changes are saved.

## SILworX Engineering Tool

If the Editor is then closed and re-opened, the message "The required data is being processed" appears.

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

Workaround: Avoid exchanging names.

If required, restart SILworX. [HE11613, Restriction]

- Unrestricted use of SILworX is only allowed for users with administrator access. When SILworX is used with a Windows user account without administrator rights, the following problems occur:
  - No license despite the existence of an Windows XP license. Only demo mode is possible.
  - The SILworX log book log file is not written to in Windows XP.
  - SILworX cannot write on the INI file. For this reason, no settings can be saved with Windows XP and Vista.
  - If a U3 USB stick is used for licensing, the OLicenseServer is terminated during start up. For this reason, SILworX has no valid license! No work-around exists under Windows XP!

Workaround: Allow the affected user account full access to the corresponding folder. Administrator rights are required to perform this action.

For Windows XP, it may be necessary to first unhide the security settings by changing the folder settings:

- Select Tool->Folder Options to open the settings window.
- In the View tab, disable the option "Simple file sharing".

For the user account, allow full access to the following folders:

- C:\Documents and Settings\All Users\Application Data\HIMA\SILworX\_v2.38.0
- C:\Documents and Settings\All Users\Application Data\SILworX\_v2.38.0

With Window Vista, allow full access to the following folders:

- C:\ProgramData\HIMA
- C:\ProgramData\SILworX\_v2.38.0

[HE 14880]

- Repeated reload fault deletes the reload information in the project. Faults occurring after repeated, consecutive reload attempts can destroy the reload information contained in the current project that corresponds to the configuration loaded into the controller.  
Workaround: Absolutely save a copy of every project state that corresponds to the configuration loaded into the controller (e.g., on a server or data medium)! [HE15121]
- If "CPU" is set as event source, A&E state variables are not written. If "CPU" is set as event source while defining the events, the operating system does not write on the state variables, through which the event states can be used in the user program. The events, however, are created. [HE15181, Restriction]

- If the value is 0, A&E state variables are not written  
If 0 is set as threshold or hysteresis value while defining scalar events, the operating system does not write on the state variables, through which the event states can be used in the user program. The events, however, are created. [HE 15182, Restriction]
- Code generation with SER modules with I/O events is not possible.  
The code generator is aborted with an error message if the project contains SER modules in which I/O events are configured.  
Workaround: Set the parameter "safeethernet CRC" located in the resource properties to "compatible to V.2.36"; in such a case, however, reload is no longer language independent. [HE15323]
- Changing the module name of a redundant module results in an infinite loop.  
Changing the name of a module redundantly connected causes SILworX to loop endlessly.  
[HE14738]
- Faulty data volume check for Modbus function code 23.  
The check of permissible buffer volume for Modbus function code 23 does not work correctly:
  - Lower limit of the read buffer is not checked.
  - Upper limit of the write buffer is 242 bytes and not 240 bytes as stated in the error message.
  - If the maximum data volume was exceeded for a function code, the error message does not disappear even if data is reduced to the permitted amount.  
Workaround: Delete and re-create write and read requests. [HE12899]
- Non-redundant I/O channel can be nevertheless redundantly assigned.  
If a channel in a redundant module pair is defined as non-redundant, it can nevertheless be assigned with a global variable in the form for the redundant channels. Furthermore, SILworX does not prevent removing the channel redundancy from a channel already assigned with a global variable.  
No error message is created by SILworX during the code generation, but the redundant assignment is ignored. [HE15558]
- SILworX compiler terminates with remote I/Os  
The SILworX compiler terminates when translating the parameters for remote I/Os, if all the following conditions are met:
  - A user-specific data-type (Struct) is defined.
  - At least two global variables with this data types are contained in the parameters for the remote I/O.
  - At least two of such variables are using the same sub-element.Workaround: Ensure that not all these conditions are met. [HE15607]

- After printing documentation, close SILworX and restart it.  
By this, termination of SILworX during start of the code generation is avoided.  
When terminating in this way, SILworX issues the error message "out of memory".  
Tip: send the print-out to a PDF file. From the PDF file, the wanted pages can be selectively sent to the printer. [HE15653, HE15676]

## 4 Migration from the Previous Version to Version 2.52

Observe the following procedure to migrate from version 2.36 or higher to version 2.52:

- Prior to converting, save the project, e.g., on a data medium.
- Open the project in version 2.52 and convert it.
- Generate the code in version 2.52 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 corresponding previous version.