

The HIMA logo consists of a blue square with rounded corners. Inside the square is a white oval containing the word "HIMA" in blue, uppercase, sans-serif font.The SMART SAFETY. logo is a yellow rectangle with a folded top-right corner. Inside, the words "SMART SAFETY." are written in blue, uppercase, sans-serif font.

Programmierwerkzeug

# SILworX<sup>®</sup>

## Release-Notes V10.64

Alle in diesem Handbuch genannten HIMA Produkte sind mit dem Warenzeichen geschützt. Dies gilt ebenfalls, soweit nicht anders vermerkt, für weitere genannte Hersteller und deren Produkte.

HIQuad®, HIQuad X®, HIMax®, HIMatrix®, SILworX®, XMR®, HICore® und FlexSILon® sind eingetragene Warenzeichen der HIMA Paul Hildebrandt GmbH.

Alle technischen Angaben und Hinweise in diesem Handbuch wurden mit größter Sorgfalt erarbeitet und unter Einschaltung wirksamer Kontrollmaßnahmen zusammengestellt. Bei Fragen bitte direkt an HIMA wenden. Für Anregungen, z. B. welche Informationen noch in das Handbuch aufgenommen werden sollen, ist HIMA dankbar.

Technische Änderungen vorbehalten. Ferner behält sich HIMA vor, Aktualisierungen des schriftlichen Materials ohne vorherige Ankündigungen vorzunehmen.

Alle aktuellen Handbücher können über die E-Mail-Adresse [documentation@hima.com](mailto:documentation@hima.com) angefragt werden.

© Copyright 2020, HIMA Paul Hildebrandt GmbH

Alle Rechte vorbehalten.

## Kontakt

HIMA Paul Hildebrandt GmbH

Postfach 1261

68777 Brühl

Tel.: +49 6202 709-0

Fax: +49 6202 709-107

E-Mail: [info@hima.com](mailto:info@hima.com)

Revisions- index	Änderungen	Art der Änderung	
		technisch	redaktionell
10.02	Erstausgabe der neu erstellten SILworX V10b Release	X	X
10.03	Neu: Kapitel Nicht mehr unterstützte HIMA Steuerungen	X	X

## Inhaltsverzeichnis

<b>1</b>	<b>SILworX V10.64</b>	<b>5</b>
<b>1.1</b>	<b>Kompatibilitäten</b>	<b>5</b>
1.1.1	Kompatibilität zum Betriebssystem des PES	5
1.1.2	Nicht mehr unterstützte HIMA Steuerungen	5
1.1.3	Kompatibilität zu bestehenden Projekten	5
1.1.4	Kompatibilität zum verwendeten Rechner	5
1.1.4.1	Benutzung von Hardlocks	5
<b>2</b>	<b>Neue Funktionen</b>	<b>6</b>
<b>3</b>	<b>Verbesserungen der Version</b>	<b>7</b>
3.1	FBS-Editor	7
3.2	Structured Text	7
3.3	Watchpage-Editor	8
3.4	SmartSafety-Test	8
3.5	Hardware	8
3.6	Online	8
3.7	Archivieren/Wiederherstellen	8
<b>4</b>	<b>Einschränkungen</b>	<b>9</b>
4.1	FBS-Editor	9
4.2	Structured Text	11
4.3	Querverweise	12
4.4	Dateiauswahldialoge	12
4.5	Hardware	13
4.6	Anwenderprogramm	14
4.7	Versionsvergleicher	17
4.8	Codegenerierung	18
4.9	Reload	19
4.10	Protokolle	20
4.11	Projekt	22
4.12	Projekthistorie	23
4.13	Dokumentation	23
4.14	Benutzerverwaltung	24
<b>5</b>	<b>Besonderheiten</b>	<b>25</b>
<b>6</b>	<b>Upgrade von einer Vorversion</b>	<b>29</b>
6.1	Referenzen	29



# 1 SILworX V10.64

Dieses Dokument beschreibt die Verbesserungen, neue Funktionen und Einschränkungen von SILworX V10.64 gegenüber Vorversionen.

## 1.1 Kompatibilitäten

### 1.1.1 Kompatibilität zum Betriebssystem des PES

SILworX V10.64 ist für folgende HIMA Systemfamilien einsetzbar:

- HIMax
- HIMatrix F-Systeme
- HIQuad X

### 1.1.2 Nicht mehr unterstützte HIMA Steuerungen

- Ab SILworX V9.36 werden nachfolgende Steuerungen nicht mehr unterstützt:
- HIMatrix F10 PCI 03
- HIMatrix F20 01
- HIMatrix F30 01
- HIMatrix F31 02
- HIMatrix F31 03
- HIMatrix F35 01
- HIMatrix F60 01

Diese Steuerungen dürfen zusammen mit SILworX Versionen ab V9.36 nicht mehr benutzt werden (Haftungsausschluss). Dies betrifft vor allem die Codegenerierung, den Download und die Online-Dienste.

D. h. Projekte in SILworX dürfen mit den aufgeführten Steuerungen geöffnet werden z. B. für Updates/Upgrades-Aktionen (Ersatz durch Nachfolgeprodukte). Eine Verbindung mit den aufgeführten Steuerungen aufzubauen, ist mit den SILworX Versionen ab V9.36 nicht mehr zugelassen.

### 1.1.3 Kompatibilität zu bestehenden Projekten

Die Version V10.64 kann Projekte konvertieren und bearbeiten, die mit einer Vorversion erstellt wurden. Bei einer Codegenerierung des unveränderten Projekts bleibt der Konfigurations-CRC erhalten.

### 1.1.4 Kompatibilität zum verwendeten Rechner

Die Mindestanforderungen an den für den Betrieb von SILworX verwendeten Rechner sind auf der jeweiligen HIMA DVD angegeben.

Speziell bei sehr großen Projekten können ältere Rechner möglicherweise lange Verarbeitungszeiten aufweisen und dadurch ungeeignet sein. Die Rechner-Hardware sollte daher möglichst dem Stand der Technik entsprechen. Bessere Hardware-Eigenschaften wie Rechenleistung und Speicherausbau führen zu verbesserter Performance.

#### 1.1.4.1 Benutzung von Hardlocks

Erfolgt die Lizenzierung von SILworX unter Windows 7, 8, 8.1 und 10 mit Hilfe von Hardlocks (USB-Sticks), ist folgendes zu beachten:

- Für die Installation sind Administrator-Rechte nötig.
- Für den Betrieb sind Benutzer-Rechte ausreichend.

## 2 Neue Funktionen

- Die Parametrierung des Moduls X-MIO 7/6 01 wurde anwenderfreundlicher gestaltet. Die Parameter (Grenzwerte) im Hardware-Editor sind jetzt in U/min und U/min<sup>2</sup> editierbar, anstatt wie zuvor in mHz bzw. mHz/s. Die resultierende Frequenz wird weiterhin in mHz und mHz/s angezeigt.

SILworX berechnet jetzt den Skalierungsfaktor automatisch über den neuen Parameter Polrad-Zähne.

Zusätzlich gibt es zwei neue Parameter der X-MIO 7/6 01 im Hardware-Editor.

\* Mit dem Parameter *Max. zulässige Drehzahlabweichung [U/min]* wird festgelegt, wie weit die 3 aufgenommenen Drehzahlwerte voneinander maximal abweichen dürfen.

\* Mit dem Parameter *Gradientenüberwachung* wird die Überwachung der Beschleunigung (Max. Steigung positiv/negativ) aktiviert oder deaktiviert

Für die X-MIO 7/6 01 gibt es keine Abwärtskompatibilität! SILworX generiert für die X-MIO 7/6 01 eine neue io4io.config-Version. Daraus ergibt sich eine CRC-Änderung.

Ein Reload ist nicht möglich.

Um die neuen Funktionen verwenden zu können, muss die X-MIO 7/6 01 mit dem passenden Betriebssystem geladen werden. Bei der Konvertierung alter SILworX-Projekte, die eine X-MIO 7/6 01 enthalten, erfolgt eine Warnmeldung. Eine anschließende Code-Generierung liefert eine Fehlermeldung, da die maximal zulässige Drehzahlabweichung bei der Konvertierung auf den Wert 0 gesetzt wurde.

## 3 Verbesserungen der Version

### 3.1 FBS-Editor

- Durch das Umbenennen von Bausteinen konnten in der Logik Konflikte für alle Ein-/ und Ausgänge von beliebigen Bausteininstanzen entstehen, welche den gleichen Namen hatten wie der Baustein vor der Umbenennung. Dieser Fehler wurde behoben.
- Ein Absturz beim Speichern des FBS-Editors in bestimmten Situationen wurde behoben. Wurde ein Teil einer FBS-Logik mit Verbindungen während eines Kopiervorgangs beim Platzieren vom System abgelehnt und damit auch die Kopier-Aktion abgebrochen, konnte es passieren, dass das nächste Abspeichern des Editors zu einem Absturz führte.
- Das Löschen eines OLT-Feldes im Bausteineditor, welches an einem Anschlusspunkt einer Variable oder Bausteininstanz verbunden ist, führt nicht mehr zu einer fehlerhaften Meldung bei der Codegenerierung, wenn am Anschlusspunkt zusätzlich eine Verbindung existiert hat.

### 3.2 Structured Text

- Die FOR-Schleife wird korrekt beendet, wenn die Steuervariable den Wertebereich ihres Datentyps überschreitet oder unterschreitet.

Der CRC der POE wird bei der Codegenerierung gegenüber früheren SILworX-Versionen verändert, wenn die Möglichkeit besteht, dass die Steuervariable den Wertebereich ihres Datentyps überschreitet oder unterschreitet. Die Änderung wird im Versionsvergleich angezeigt.

Bei der Abarbeitung der FOR-Schleife kann es vorkommen, dass der eigentliche Endwert der Steuervariable außerhalb ihres Wertebereichs liegt, so dass es zur Laufzeit zu einem Überlauf kommt. In diesem Fall wurde die FOR-Schleife nicht korrekt beendet.

Wenn die Möglichkeit besteht, dass der Endwert außerhalb des Wertebereichs liegt, generiert SILworX korrigierten Code für die FOR-Schleife. Der CRC der POE wird dadurch bei der Codegenerierung gegenüber früheren SILworX-Versionen verändert. Die Änderung wird im Versionsvergleich angezeigt.

Wenn die Steuervariable den Datentyp LINT hat, kann die Korrektur und CRC-Änderung auch in Fällen stattfinden, in denen sie nicht nötig wäre.

### 3.3 Watchpage-Editor

- Watchpages: Die Verwendung von globalen/lokalen Variablen per Drag and Drop ist nun uneingeschränkt möglich. Konkret können Variablen mit einem speziellen Ziffernsuffix im Namen nach folgendem Muster verwendet werden: "\_0#" wobei die # eine oder mehrere Ziffern darstellt und die Anzahl der führenden Nullen beliebig groß sein kann, z.B.: "Programm.Poe1.Var\_01", "Globale Variable\_002".

### 3.4 SmartSafety-Test

- Wenn für eine vorzeichenlose GV zu einem Sollwert ein Toleranzwert angegeben ist, welcher größer als der Sollwert ist, wird dieser Toleranzwert nun korrekt verarbeitet. In SILworX V10.58.0 wurde für beispielsweise einen Sollwert 10 mit Toleranzwert 11 die untere Grenze des Toleranz-Bereichs nicht korrekt berechnet, so dass beim Testdurchlauf der Status "Not OK" angezeigt wurde, obwohl der Istwert innerhalb der geforderten Grenzen lag.

### 3.5 Hardware

- Der Anwender kann nun nach Hinzufügen oder Löschen einer Remote I/O eine Detailview der betreffenden Konfigurationsdatei öffnen. Weiterhin stürzt die Dokumentation nicht mehr ab, sobald der Versionsvergleich enthalten ist.
- Wenn WDZ oder Prozess-Sicherheitszeit online oder im Projekt eingestellt werden, erscheint nun auch für HIQuad X eine Warnung, die auf das Verletzten des Verhältnisses  $3 * WDZ \leq \text{Prozess-Sicherheitszeit}$  hinweist.
- Meldungen für Temperaturschwelle 1 der Netzgeräte F-PWR 01 werden als 'Info' angezeigt, unabhängig davon, ob der Anwender Warnungen für das Basis-Rack deaktiviert hat.
- In der Diagnoseübersicht für HIQuad-Systeme wurden kleinere Fehler behoben:
  1. Die voreingestellte Aktualisierungsrate wurde erhöht.
  2. Die Sortierung der Tabelle bleibt nach Aktualisierung erhalten.
  3. Meldungen bereits gelöschter Module werden entfernt.
  4. Meldungen für Temperaturschwelle 1 werden als 'Info' angezeigt.
  5. Projekte mit englischer Spracheinstellung zeigen englische Meldungstexte.

### 3.6 Online

- Bei lange laufenden Online-Kommandos, wie z. B. Firmware-Update, in Verbindung mit großen Systemen, wie z. B. bei einem HIMax-System mit vielen Racks, kann es nach längerer Zeit zu einer Fehlermeldung kommen und/oder es bleibt mindestens einer der Kommandodialoge hängen und lässt sich nicht mehr schließen. Dieser Fehler wurde beseitigt.

### 3.7 Archivieren/Wiederherstellen

- Wurde in einer Version vor V10 ein Archiv erzeugt, dessen Knoten einen Namen hatte, der kein gültiger Dateiname war (z.B. weil er ein / enthielt), so kann das Archiv in V10 nicht wiederhergestellt werden. Dieser Fehler wurde behoben.
- Hat man versucht, ein Archiv wiederherzustellen, welches Objekte enthielt, für die man keine Lizenz besaß, so terminierte SILworX (bei der Ausgabe der Fehlermeldung).



## 4 Einschränkungen

Beim Einsatz von SILworX sind die folgenden Einschränkungen zu beachten. Bei Beachtung der folgenden Hinweise haben die Einschränkungen keine Auswirkungen auf die Sicherheit und die Verfügbarkeit des Codes, der für eine Steuerung generiert wird.

### 4.1 FBS-Editor

- **Auswirkung:**  
Leere Seiten im Logikbereich des FBS-Editors sind nicht immer löschar

**Bedingung:**

Der Menüpunkt Leere Seite löschen im Kontextmenü ist nicht aktiv, wenn folgende Bedingungen zusammentreffen:

- Eine Linie verläuft über zwei oder mehrere Nachbarseiten der leeren Seite
- Die Linie verläuft nicht durch die leere Seite

Die leere Seite kann somit nicht gelöscht werden.

**Abhilfe:**

Es existiert keine Abhilfe für das Problem

- **Auswirkung:**  
Auf dem FBS-Editor wird der Text bestehender Seitenkommentare nach dem Einfügen neuer Seiten nicht angezeigt.

**Bedingung:**

Folgende Bedingungen müssen dabei zusammentreffen:

- Ein Seitenkommentar befindet sich neben einer zweiten Seite mit mindestens einem Logikelement.
- Auf der zweiten Seite wird die Aktion "Leere Blätter einfügen" -> "Spalte oder Zeile einfügen" so ausgeführt, dass die neuen Seiten zwischen der Seite mit dem Blattkommentar und der Seite mit dem Element eingefügt werden.

Der Text des bestehenden Seitenkommentars wird temporär nicht mehr angezeigt.

**Abhilfe:**

Der Text erscheint wieder, wenn der Kommentar bewegt oder der Editor geschlossen und neu geöffnet wird.

- **Auswirkung:**  
Auf dem FBS-Editor werden Seiteninformationen nach einem Projektimport aus ELOP II nicht korrekt platziert.

**Bedingung:**

Folgende Bedingungen müssen dabei zusammentreffen:

Ein zugeordnetes Kommentar- oder OLT-Feld liegt auf einer leeren Seite ohne weitere Logikelemente, während sich das Hauptelement auf einer anderen Seite befindet.

**Abhilfe:**

Zugeordnete Kommentar- oder OLT-Felder sollten sich auf der gleichen Seite wie ihre Hauptelemente befinden.

[HE 27520 t.koller]

- **Auswirkung:**  
Konflikt-Icon bei Variablen bleibt auch nach Behebung des Konflikts sichtbar.

**Bedingung:**

In folgenden Fällen bleibt das Konflikt-Icon auch nach Ablehnung einer ungültigen Aktion und Anzeige des korrekten Werts sichtbar:

- Eingabe eines ungültigen Namens an einer Variablen
- Vergabe einer bereits existierenden Reihenfolge-Nummer an einer Interface-Variablen

**Abhilfe:**

Verifizieren oder Aktualisieren.

- **Auswirkung:**  
Anzeige von globalen Variablen, die als VAR\_EXTERNAL verwendet werden:  
Werden globale Variable mit Datentyp Struct oder Array als VAR\_EXTERNAL verwendet, zeigt der FBS-Editor für die Unterelemente die Einträge in den Spalten Initialwert, Beschreibung, Zusatzkommentar und technische Einheit nicht an

**Bedingung:**

Global Variable vom Typ Struct erstellen und in einem Struct-Element eine Beschreibung eintragen. Nun diese GV in einem Programm als Wertfeld verwenden. Anschließend die erzeugte VAR\_EXTERNAL in der Lokale Variablen-Lasche anschauen.

**Abhilfe:**

Attributeigenschaften der VAR\_EXTERNAL in der entsprechenden Globalen Variable anschauen.

- **Auswirkung:**  
Durch das Löschen einer verwendeten POE und Neuanlegen/Einfügen eines Typicals mit dem Namen der gelöschten POE kommt es in der Codegenerierung bei diesem nicht unterstützten Typ zu einen Absturz.

**Bedingung:**

Einen FB POE "B" anlegen und diesen in einem Programm verwenden. Alles speichern und schließen. Die POE "B" löschen und dort ein Typical Baustein "B" anlegen. An der Verwendung von "B" die Aktualisierung aufrufen. Es werden die Fehler bereinigt und das Typical wird aufgerufen.

**Abhilfe:**

Die Änderung kann rückgängig gemacht werden, in dem der richtige Typ von Baustein verwendet wird. Der richtige Typ der Instanz kann durch einen Doppelklick auf die Instanz kontrolliert werden. Die dann aufgerufene POE darf nicht vom Typ Typical sein, zu erkennen am kleinen Icon neben dem Namen.

- **Auswirkung:**  
Gleichzeitige Änderung von Datentyp und Name einer GV führt zu Verlust der Referenz bei Verweisen auf Unterelemente dieser GV.  
  
**Bedingung:**  
Voraussetzungen: Array\_1[1..5], Array\_2[1..8], GV vom Typ Array\_1. Verweis auf GV[1].  
Aktion: GV-Editor öffnen. Datentyp der GV auf Array\_2 ändern. Name der GV ändern. Editor speichern.  
Ergebnis: Im Verweis steht noch immer der alte Name GV[1], der aber nun nicht mehr gefunden wird.  
  
**Abhilfe:**  
Entweder zwischen Änderung des Datentyps und des Namens speichern, oder - wenn es dazu schon zu spät ist - Verweis manuell anpassen.

## 4.2 Structured Text

- **Auswirkung:**  
Im ST-Editor sind 2700 aufeinanderfolgende Kommentarzeilen nicht möglich.  
  
**Bedingung:**  
Beim Auskommentieren mit 2700 aufeinanderfolgende Zeilen im ST-Editor terminiert SILworX.  
  
**Abhilfe:**  
Längere Kommentare unterteilen, z. B. jeweils 1000 Zeilen zu einen Kommentar zusammenfassen.
- **Auswirkung:**  
Es kommt zu einem Absturz, wenn man innerhalb eines ST-Editors einen benutzerdefinierten Datentyp von einer Variablen zu einer anderen kopiert und sich die Auswahlbox des Datentyps der Zielvariablen im Editiermodus befindet.  
  
**Bedingung:**  
Eine ST-Funktion oder einen ST-Funktionsbaustein öffnen und 2 Variablen anlegen (Variablentyp spielt keine Rolle). Der ersten Variable einen benutzerdefinierten Datentyp zuweisen, dann diesen Datentyp in der Auswahlbox selektieren und im Kontextmenü "Kopieren" auswählen. Anschließend auf dem Datentyp der zweiten Variable "Enter" ausführen, sodass die Auswahlbox in den Editiermodus wechselt. In diesem Zustand muss der zuvor kopierte Datentyp eingefügt werden.  
  
**Abhilfe:**  
Einen selektierten benutzerdefinierten Datentyp einer Variable innerhalb eines ST-Bausteins zu einer anderen Variablen kopieren, ohne die Datentyp-Auswahlbox der Zielvariablen in den Editiermodus zu versetzen, d.h. ohne die Enter-Taste zu betätigen oder ein Doppelklick auszuführen.

### 4.3 Querverweise

- **Auswirkung:**  
Es wird für Variablen-Querverweise keine Strukturinfo angezeigt, wenn diese in HIPRO-S benutzt werden, das Feld bleibt leer.  
  
Bedingung:  
Variable in HIPRO-S-Verbindung benutzen und im GV-Editor die Querverweise anschauen:  
Strukturinfo bleibt leer.  
  
Abhilfe:  
keine Abhilfe
- **Auswirkung:**  
Querverweise werden in der Sprache angezeigt, in der sie gespeichert wurden. Die aktuelle Spracheinstellung in SILworX spielt keine Rolle  
  
Bedingung:  
Querverweise erzeugen, die sprachabhängige Teile enthalten. Dann das Projekt mit einer SILworX-Instanz mit einer anderen Sprachversion öffnen.  
Die Querverweise werden nach wie vor die Texte in der vorherigen Sprachversion enthalten.  
Beispiel: Die Namen der Systemvariablen im Hardware-Editor werden im Querverweis der Globalen Variablen angezeigt, bleiben aber, wenn in einem deutschen SILworX verknüpft, auch in einem englischen SILworX in der deutschen Version.
- **Abhilfe:**
  - a) Ignorieren
  - b) Den Editor speichern, an dem die Verknüpfung existiert. Dazu muss zuvor eine Änderung in diesem Editor vorgenommen worden sein. Dabei werden die Querverweisinformationen neu erstellt.

### 4.4 Dateiauswahldialoge

- **Auswirkung:**  
Der Dialog des Hardware-XML-Imports bleibt nach fehlerhafter Dateiauswahl offen und reagiert nur noch auf den Abbrechen-Button  
  
Bedingung:  
Im Hardware-XML-Import-Dialog einen falschen oder ungültigen Dateipfad auswählen und den Import starten.  
  
Abhilfe:  
Dialog mit 'Abbrechen' beenden und Import erneut starten.

- **Auswirkung:**  
Im Wiederherstellungsdialog wird nur der Pfad der zuletzt wiederhergestellten Archivdateien angezeigt. Es fehlt die Liste der zuletzt angelegten Archive.

**Bedingung:**

Ein Projekt öffnen oder anlegen. Davon ein Archiv oder mehrere davon erstellen. Im Wiederherstellungsdialog die Auswahlbox unter Archivdatei öffnen: Diese ist leer, es werden keine anderen Archivdateien gelistet.

**Abhilfe:**

Archivdatei manuell suchen.

## 4.5 Hardware

- **Auswirkung:**  
Im Hardware-Editor ist in der Detailansicht-Toggle-Button nicht aktiv. Die Ansicht kann über den Schließen-Button geschlossen werden.

**Bedingung:**

Hardware-Onlineansicht starten und die Detailansicht einer Baugruppe öffnen: der Detailview-Toggle-Button ist nicht aktiv.

**Abhilfe:**

Nicht notwendig, da ein Schließen-Button in der Detailansicht existiert.

- **Auswirkung:**  
Ab SILworX V6 sortiert SILworX bei der Codegenerierung die Lizenzen nach Namen, nicht mehr nach der Reihenfolge des Eintragens. Dadurch kann es bei der Projektkonvertierung aus früheren Versionen zu CRC-Änderungen kommen.

**Bedingung:**

Lizenznamen in nicht alphabetischer Reihenfolge in V5 eintragen und Code generieren. Dann in V6 Code generieren.

**Abhilfe:**

Geeignete Namensvergabe, HIMA-Support hinzuziehen.

- **Auswirkung:**  
Die globale Fehlerstatistik wird nicht in RIOs sondern nur im Master angezeigt.  
Die Systemvariablen für die Fehlerstatistik und diejenigen für das Forcen sowie CPU-Autostart-Freigabe, CPU-Start-Freigabe, CPU-Haupt-Freigabe, ReadOnlyInRun, Start-Cycle werden bei RIOs nicht bedient.

**Bedingung:**  
Systemvariablen mit Globalen Variablen verschalten und entsprechende Fehler simulieren.

**Abhilfe:**  
Fehlerstatistik im Master verwenden

## 4.6 Anwenderprogramm

- **Auswirkung:**  
Folgendes Verhalten der Funktion EXPT weicht auf den PES-Systemen von der IEEE-754-Norm ab.  
1.0 \*\* NaN := 1.0 erwartet: NaN  
EXPT.ENO := TRUE erwartet: FALSE

NaN \*\* 0.0 := 1.0 erwartet: NaN  
EXPT.ENO := TRUE erwartet: FALSE

Mit OTS und Offline-Simulation funktioniert EXPT gemäß IEEE-754.

**Bedingung:**  
s.o.

**Abhilfe:**  
Bei ENO-Bedarf den Wert NaN an beiden Eingängen programmtechnisch abfangen / vermeiden.

- **Auswirkung:**  
Der Funktionsbaustein MUL liefert falsche Ergebnisse, wenn folgende Bedingungen zusammentreffen:
  - HMatrix Standard-Ressource (z.B. F30 01, F35 01, F60 01)
  - Datentyp LREALAm Eingang IN1 liegt der Wert  $\pm\infty$ , am Eingang IN2 liegt nan (not a number) an
  - In diesem Fall ist das Ergebnis  $-\infty$ , und nicht nan wie spezifiziert.
  - Der ENO liefert in diesem Fall korrekt das Ergebnis False.

**Bedingung:**  
s. Auswirkung

**Abhilfe:**  
Ergebnis in diesem speziellen Fall ignorieren

- **Auswirkung:**  
Die Funktion DIV\_TIME aus der Standardbibliothek setzt fälschlich den Fehlerausgang ENO auf FALSE und meldet somit einen Fehler unter folgenden Bedingungen:
  - Der Eingang IN2 (Divisor) ist vom Typ REAL.
  - Der Wert von IN2 ist +/-INF.

- **Bedingung:**  
DIV\_TIME mit EN/ENO verwenden und als Divisor +/- INF anlegen. (INF ist z.B. das Ergebnis aus 1.0 / 0.0)

**Abhilfe:**  
ENO in diesem Fall ignorieren

- **Auswirkung:**  
Die Funktion EXPT liefert bei Offline-Simulation und OTS für die Basis IN1 = 1.0 und Exponent IN2 = -INF den Wert NaN statt 1.0.

**Bedingung:**  
EXPT mit IN1 = 1.0 und IN2 = -INF (oder andere große negative Zahl aufrufen) und Ergebnis in der Offline-Simulation oder OTS ansehen.

**Abhilfe:**  
Wenn der Sonderfall für die Applikation relevant ist, muss dies in der Anwenderlogik entsprechend ausprogrammiert werden.

- **Auswirkung:**  
Bei der Abarbeitung einer POE werden zuerst die Schrittketten bearbeitet, danach die AS-Aktionen und dann die FBS-Logik. Das führt dazu, dass die Eingangswerte von AS-Transitionen und AS-Aktionen, die in der FBS-Logik beschrieben werden, immer aus dem vorherigen Zyklus stammen. Bei der konkreten Auswertung der Eingangswerte bestehen allerdings kleine Unterschiede:  
Der Eingangswert einer AS-Transition wird während der FBS-Bearbeitung in den Speicher der AS-Transition geschrieben und dort gemerkt und erst im nächsten Zyklus in der Schrittkette bearbeitet. Dies führt dazu, dass nach einem Kaltstart Schrittketten generell erst ab dem zweiten Zyklus weitergeschaltet werden.  
Der Eingangswert einer AS-Aktion wird bei der Abarbeitung der AS-Aktion aus der Quelle gelesen. Falls dies eine Funktion ist, wird der Initialwert gelesen, da Funktionen zu Beginn der POE-Ausführung initialisiert werden und erst nach den AS-Aktionen ausgeführt werden.

**Bedingung:**  
s. oben

**Abhilfe:**  
**AS-Transition:**  
Bei der Programmierung der Schrittkette muss berücksichtigt werden, dass frühestens im zweiten Zyklus eine AS-Transition durchgeführt wird.  
**AS-Aktion:**  
Um ein Funktionsergebnis als Eingangswert einer AS-Aktion zu verwenden, muss zwischen Ausgang der Funktion und Eingang der AS-Aktion eine Variable geschaltet werden.

- **Auswirkung:**  
Wenn die Ketten einer Auswahlverzweigung mit AS-Schritten enden, führt dies zu einem Deadlock. Wenn die Ketten einer Simultanverzweigung mit Transitionen enden, führt dies zu mehreren aktiven Schritten außerhalb der Simultanverzweigung.

**Bedingung:**  
s. oben

**Abhilfe:**  
Der Anwender muss durch geeignete Maßnahmen dafür sorgen, dass derart fehlerhafte Schrittketten nicht verwendet werden.

- **Auswirkung:**  
Beim Zugriff auf ein Array-Element mit einem Index außerhalb des Wertebereiches des Arrays wird nach einem definierten und performanten Verfahren auf ein Element innerhalb des Arrays zugegriffen, um willkürliche Zugriffe auf Speicherbereiche zu vermeiden.

**Bedingung:**  
s. oben

**Abhilfe:**  
Der Anwender muss durch geeignete Programmierung dafür sorgen, dass nur mit Indizes innerhalb des Wertebereichs des Arrays auf Array-Elemente zugegriffen wird.

- **Auswirkung:**  
Verschiedene Elemente einer Struktur-Variablen nicht von unterschiedlichen Quellen beschreibbar  
Es ist nicht möglich, dass sowohl das Anwenderprogramm als auch die Hardware oder die Kommunikation zwei unterschiedliche Elemente derselben Struktur-Variablen beschreiben.

**Bedingung:**  
s. oben

**Abhilfe:**  
Unterschiedliche Struktur-Variablen für Elemente, in die das Anwenderprogramm schreibt, und für Elemente, in die Hardware oder Kommunikation schreiben.

[HE 15700 s.braun]

- **Auswirkung:**  
Es ist nicht möglich, Elemente von Variablen eines benutzerdefinierten Datentyps als Index eines Arrays zu benutzen.

**Bedingung:**  
s. oben

**Abhilfe:**  
Wert der gewünschten Variablen in einfache Variable kopieren und diese als Index verwenden.



## 4.7 Versionsvergleicher

- **Auswirkung:**  
Die Detail-Ansicht einer POE beim Versionsvergleich zeigt unberechtigt eine Änderung für eine POE-Instanz an, wenn die beiden folgenden Punkte zutreffen:
  1. Die Vergleichsbasis wurde mit einer Version vor V4.116 erzeugt.
  2. Der aufgerufene POE-Typ enthält Umlaute im Namen.

**Bedingung:**  
s. oben

**Abhilfe:**  
In Bausteinnamen nur Leerzeichen und Zeichen aus der folgenden Liste verwenden

- 0 1 2 3 4 5 6 7 8 9
- A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
- a b c d e f g h i j k l m n o p q r s t u v w x y z
- \$ % & ( ) \* + - / : ; < = > ? \ ^ \_ ` { | }

- **Auswirkung:**  
Der Vergleich zwischen einer mit SILworX vor V9 generierten und einer mit SILworX ab V9 generierten Konfiguration zeigt die Meldung "Die Reihenfolge der Variablen- und Instanzendeklaration hat sich geändert." nicht mehr an
  - 1) Für alle Funktionen und Funktionsbausteine
  - 2) Für Programme die (in der ab V9 generierten Konfiguration) mindestens eine der folgenden Standardfunktionen verwenden: ADD, SUB, MUL, DIV, MOD, MOVE, AND, OR, XOR, NOT, SHL, SHR, alle Ato...-Bausteine, ADD\_TIME, SUB\_TIME, MAX, MIN, SEL, MUX, GT, LT, GE, LE, EQ, NE, PACK.

**Bedingung:**  
In SILworX vor V9 eine Ressource mit zwei Konfigurationen (1. geladen oder importiert, 2. generiert) erstellen, deren Versionsvergleich (zu Recht) eine Änderung der Deklarationsreihenfolge in einer POE meldet.  
Der Projektstand muss dem der generierten Konfiguration entsprechen.  
Das Projekt nach SILworX V9 konvertieren und Code generieren.  
Im Versionsvergleich fehlt die Reihenfolge-Meldung, obwohl die betreffende POE den gleichen CRC-Unterschied aufweist wie zuvor.

**Abhilfe:**  
Wenn der zur Vergleichsbasis (z. B. zur geladenen Konfiguration) passende Projektstand noch verfügbar ist

1. Eine Kopie dieses alten Projektstandes nach SILworX V9 bzw. in die gewünschte, neue Version konvertieren. Dort Code generieren und das Ergebnis über den Startdialog des Versionsvergleichs exportieren.
2. Das andere Projekt, in dem der Versionsvergleich stattfinden soll, in der neuen SILworX-Version öffnen, die eben exportierte Konfiguration im Startdialog des Versionsvergleichs importieren und als Vergleichsbasis verwenden.

## 4.8 Codegenerierung

- **Auswirkung:**  
Konflikt durch Änderung des Konstant-Attributs globaler Variablen nach Verwendung:  
Wird eine globale Variable als VAR\_EXTERNAL verwendet und danach von konstant auf änderbar gesetzt oder umgekehrt, führt dies bei der Codegenerierung zum Konflikt, wenn diese VAR\_EXTERNAL beschrieben wird und die globale Variable konstant ist.

**Bedingung:**

GV als VAR\_EXTERNAL in der Logik verwenden und den Konstant-Zustand in der GV ändern.

**Abhilfe:**

Globale Variable an allen Verwendungsstellen entfernen, so dass VAR\_EXTERNAL verschwindet. Anschließend an allen Stellen neu einfügen

## 4.9 Reload

- **Auswirkung:**

In einer Reload-Codegenerierung kommt es zu einer Reload-Warnung, wenn die Download-Info des Partners nicht zu finden ist:

"Safeethernet-Reload-Sequenz gestartet. Es wurde eine Dualkonfiguration angelegt. Es ist keine Download-Konfiguration für [Partner-Name] zum Abgleich der safeethernet-Signaturen vorhanden."

Das geschieht auch dann, wenn die Reload-Codegenerierung ohne vorherige Änderungen gestartet wurde.

**Bedingung:**

Safeethernet-Verbindung mit Proxy erstellen und laden. Reload-CG auf der realen Steuerung ohne irgendwelche Änderungen.

**Abhilfe:**

Nicht vorhanden.

- **Auswirkung:**

1. Es kann eine unverständliche Fehlermeldung erscheinen, wenn beim Zuweisen der Vergleichs-Config Abbruch gewählt wurde.

2. Die Benutzerführung in SILworX bei einer Reload-Codegenerierung mehrerer Ressourcen und Import einer Config-Datei macht erst nach einer Bestätigung der Config-Datei mit der Reload-Codegenerierung weiter.

**Bedingung:**

Eine Reload-Codegenerierung auf einer Konfiguration mit mehreren Ressourcen starten, die eine Backup- oder Import-Config enthalten und im Auswahldialog abbrechen.

**Abhilfe:**

Nicht vorhanden

## 4.10 Protokolle

- **Auswirkung:**  
In einem Projekt vor V6 eine safeethernet-Verbindung mit zwei Ressourcen (Res1, ID=1 und Res2, ID=2) erzeugen und in einer aktuellen Version die Verbindung auf "Ab V6" einstellen. Das kann zur Folge haben, dass Res2 zum Timing Master ernannt wird.

**Bedingung:**

Aus Anwendersicht ist keine Vorsage möglich, wer beim Übergang zu "Ab V6" zum Timing Master ernannt wird.

**Abhilfe:**

Timing Master explizit festlegen.

- **Auswirkung:**  
Bei Überschreitung der zulässigen Maximaldatenmengen von PROFIBUS DP Slave wird bei der Verifikation ein Fehler gemeldet.

**Bedingung:**

Die Prozessdaten für einen PROFIBUS DP Slave müssen die erlaubte Maximaldatenmengen überschreiten:

Eingangsdatengröße > 192 Bytes (bei HIQuad H41X bzw. 51X > 244 Bytes)

Ausgangsdatengröße > 240 Bytes (bei HIQuad H41X bzw. 51X > 244 Bytes)

Verifikation bzw. Codegenerierung durchführen um die Meldung zu erzeugen.

**Abhilfe:**

Grenzen einhalten

- **Auswirkung:**  
SILworX lehnt die Reload-Codegenerierung nicht ab, wenn ein COM-Protokoll von Klassisch auf Schreibanforderung umgestellt wird. Das Verhalten der Steuerung wird in HE 31070 weiter verfolgt.

**Bedingung:**

Auf einer COM z.B. den Modbus Slave von V1 nach V2 per Reload umrüsten, dabei den Namen des Protokolls jedoch beibehalten.

Damit entsteht ein Reload-Übergang, den SILworX ablehnen muss.

Ganz allgemein gilt das, wenn ein COM-Protokoll von Klassisch (Modbus Slave V1, PROFIBUS Slave, Modbus Master etc.) auf Schreibanforderungsprotokoll (aktuell Modbus Slave V2, aber auch künftige Protokolle) per Reload umgerüstet wird.

**Abhilfe:**

Protokoll umbenennen. Damit bekommt es eine neue Identität, der Reload ist als Cold-Reload möglich.

- Auswirkung:  
Bei der Codegenerierung entstehen unberechtigte Meldungen der Art: Unkritischer Fehler:  
Redundanter Master 'Master2' im redundanten Slave 'Modbus-Slave\_1' nicht gefunden.

Diese Meldung entsteht nicht bei der reinen Validierung.

Damit sind gegenseitige, namentliche Master-Referenzierungen nicht möglich.

Die Funktionalität dient ausschließlich der Validierungshilfe für den Anwender. Es gibt keinen Einfluss auf die Redundanz oder Konfiguration.

Bedingung:

1) Modbus Slave V2 mit zwei Slaves anlegen.

2) Die jeweiligen Master unterschiedlich benennen.

3) In den Mastern jeweils die "Explizite Prüfung redundanter Master" und die redundanten Master benennen.

In der Validierung des Modbus Slave Sets entsteht deswegen keine Meldung, jedoch in der Codegenerierung.

Abhilfe:

Es reicht, zur Validierung nur von einem Master zum anderen zu referenzieren. Daher die Referenz vom unberechtigt gemeldeten Problem entfernen. Es entsteht kein Validierungsverlust.

## 4.11 Projekt

- **Auswirkung:**  
Projekte können bei der Windowsinternen Synchronisierung mit Netzlaufwerken auf dem Netzlaufwerk verloren gehen. Sie sind dann unter Umständen nur noch lokal verfügbar.

**Bedingung:**

Ein Projekt in einem synchronisierten Ordner öffnen, bearbeiten und schließen.  
Anschließend ist das Projekt vom Netzlaufwerk gelöscht.

**Abhilfe:**

Projekte nicht öffnen, wenn sie synchronisiert werden, und eine Netzwerkverbindung besteht. Durch das Öffnen des Projektes wird die Netzversion von Windows gelöscht. Sollte es dennoch dazu gekommen sein, kann die lokale Kopie wieder auf das Netzlaufwerk kopiert werden.

- **Auswirkung:**  
Werden in Kombination ein geschütztes Trennzeichen und Leerzeichen im Programmname verwendet, kann SILworX terminieren.

**Bedingung:**

Im Strukturbaum unter der Ressource ein Programm "P 1" anlegen. Unter derselben Ressource dann ein Programm "P- 1" anlegen, wobei bei diesem Namen vor dem Leerzeichen ein durch Alt-0173 eingebautes geschütztes Trennzeichen steht. Dann stürzt SILworX ab.

**Abhilfe:**

Keine Abhilfe

- **Auswirkung:**  
In SILworX V4 konnte es bei Löschaktionen dazu kommen, dass in der Datenbank Objekte übrig blieben, die nicht mehr bearbeitet werden konnten. Diese Objekte hatten keinen Einfluss mehr auf das übrige Projekt, wurden aber bei Prüfe Projektintegrität gemeldet.

**Bedingung:**

Projekte der SILworX Versionen 4 und 5, die solche "übriggebliebenen" Objekte enthalten, sind mit großer Wahrscheinlichkeit nicht in die Versionen V6 und V7 konvertierbar. Die Wahrscheinlichkeit ist besonders groß, wenn die Projekte benutzerdefinierte Datentypen enthalten.

**Abhilfe:**

Die in der Integritätsprüfung gefundenen Objekte sind vor der Konvertierung zu entfernen. Die einfachste Vorgehensweise hierzu ist nachfolgend beschrieben und ist in der alten SILworX-Version durchzuführen:

- 1) Alle Knoten im Strukturbaum, die sich unterhalb des Projekt-Knotens im Projekt befinden, archivieren, mit Ausnahme des Knotens Programmiergerät.
- 2) In der alten SILworX Version ein neues Projekt erstellen.
- 3) Im neuen Projekt den Knoten Konfiguration löschen.
- 4) Die unter Punkt 1 archivierte Konfiguration und, falls vorhanden, weitere Knoten im neuen Projekt unter dem Projektknoten wiederherstellen.

Das so erzeugte Projekt sollte in die aktuelle SILworX Version konvertierbar sein.

- **Auswirkung:**  
Fehlermeldung aufgrund nicht auflösbarer Ziele nach Projekt-Konvertierung auf V10  
Verweise auf Objekte, die in einer vor-V10-Version gelöscht wurden, werden nach der Konvertierung wieder sichtbar und müssen manuell entfernt werden. Das kann z.B. eine gelöschte CPU- oder COM-Baugruppe sein, die in einem safeethernet-Interface Kanal genutzt wurde. Dieser Kanal zeigt in V10 den Verweis "?" auf die nicht mehr verfügbare Baugruppe. In anderen Fällen wird ein Pfad auf das verwiesene Objekt angezeigt.

**Bedingung:**

Mit einer Version vor V10 (beispielsweise V9.36.0) ein neues Projekt mit zwei Ressource anlegen und jeweils eine CPU und eine COM einfügen. Zwischen den Ressourcen eine Safeethernet-Verbindung erzeugen und beide Baugruppen verwenden. Dann COM oder CPU löschen.

Der Verweis auf den Kanal scheint verschwunden zu sein. Nun nach V10 konvertieren. Verweis ist wieder da (mit einer "?" als System-ID, da das Ziel nicht auflösbar ist).

**Abhilfe:**

Den Verweis von Hand korrigieren bzw. auf "Keine" setzen. Im Fall eines safeethernet-Interface: den zweiten Kanal manuell deaktivieren.

## 4.12 Projekthistorie

- **Auswirkung:**  
Beim Import eines mit SILworX V2 erstellten englischen Projekts interpretiert SILworX das Datum in der Projekthistorie falsch. Z. B. wird 1/11/2013 statt dem 11. Jan 2013 als 1. November 2013 interpretiert. 1/13/2013 wird als ungültiges Datum interpretiert und führt zum Ergebnis 1. Jan 2000 (Standardwert).

**Bedingung:**

SILworX V2 (oder kleinere Version) Projekt mit englischer Sprachversion erzeugen. Anschließend dieses Projekt mit V8.34 öffnen. Die Meldungen werden in der oben angegebenen Weise in die Projekthistorie eingelesen.

**Abhilfe:**

keine Abhilfe

## 4.13 Dokumentation

- **Auswirkung:**  
Querverweise von Struktur- oder Array-Elementen werden nicht ausgedruckt.

**Bedingung:**

Strukturelement oder Array-Element verwenden und Querverweis in Dokumentation ansehen.

**Abhilfe:**

keine Abhilfe

[HE 17921 t.koller]

## 4.14 Benutzerverwaltung

- **Auswirkung:**  
Beim Wiederherstellen eines Benutzerverwaltungs-Archivs, welches mit V9 erstellt wurde, wird ein enthaltener Standard-Benutzer nicht berücksichtigt. Der Anwender muss sich dabei mit Benutzernamen und Passwort explizit anmelden

**Bedingung:**

- Neues Projekt mit Benutzerverwaltung anlegen.
- Standardbenutzer festlegen.
- Benutzerverwaltung archivieren.
- vorhandene Benutzerverwaltung im Projekt entfernen.
- archivierte Benutzerverwaltung wiederherstellen.

**Abhilfe:**

keine Abhilfe

- **Auswirkung:**  
PES-Zugriffe von Benutzergruppen, in deren Name ein (oder mehrere) Anführungszeichen (") vorkommt, funktionieren nicht in der sonst üblichen Art:  
Beim Anmelden auf eine Steuerung, wird das Passwort der Benutzergruppe für diese Steuerung nicht automatisch vorgeschlagen. Stattdessen könnte manuell der Name der Gruppe ohne Anführungszeichen (") sowie das entsprechende Passwort eintragen werden. Dazu muss das Passwort allerdings bekannt sein.

Beim Umbenennen der Gruppe kann die Verknüpfung ganz verloren gehen, und die Zugriffsberechtigung wird auch nicht mehr angezeigt. Sie bleibt aber erhalten, und wird bei der Codegenerierung beachtet.

**Bedingung:**

Eine Benutzergruppe mit (mindestens einem) Anführungszeichen (") im Namen erzeugen.  
Ein Benutzer-Konto in dieser Gruppe anlegen. Einen PES-Ressourcenzugriff für diese Benutzergruppe anlegen. Code generieren und laden.  
Mit dem Benutzer-Konto anmelden und versuchen, sich auf der Steuerung zu verbinden.  
Die Anmeldedaten werden dabei nicht automatisch vorbelegt. Um sich mit der Steuerung zu verbinden, ist es möglich, manuell den Namen der Benutzergruppe ohne alle Anführungszeichen (") und das Passwort einzugeben.  
Beim Umbenennen der Benutzergruppe wird der PES-Zugriff nicht mehr angezeigt, bleibt aber erhalten. Das heißt, es ist nach wie vor möglich, sich mit dem alten Namen der Benutzergruppe auf der Steuerung anzumelden.

**Abhilfe:**

Keine Benutzergruppen mit Anführungszeichen (") im Namen verwenden.  
Wurde ein Zugriff mit Anführungszeichen (") angelegt, so kann die Situation bereinigt werden, indem die Benutzergruppe umbenannt wird, und zwar so, dass genau die Anführungszeichen (") entfernt werden. Nun werden beim Löschen der Benutzergruppe auch die Zugriffe entfernt.



## 5 Besonderheiten

Bei der Benutzung von SILworX sind die beschriebenen Besonderheiten zu beachten.

- **Auswirkung:**

Im HW-Editor werden Skalierungseinstellungen eines Analogwerts als REAL gelesen. SILworX liest die eingegebenen Werte für die Stützpunkte eines Analogwerts (bei 4 mA und 20 mA) als REAL. Die Weiterverarbeitung erfolgt dagegen als LREAL. Auch im Anwenderprogramm kann LREAL benutzt werden. Die Einschränkung hat aber nur bei sehr großen oder sehr kleinen Stützpunkt-Werten Auswirkungen.

**Bedingung:**

Durch die Verwendung von extrem kleinen oder großen Stützpunkt-Werten kann die Genauigkeit des Prozesswertes schlechter werden.

**Abhilfe:**

Rohwerte im Anwendungsprogramm verarbeiten.

- **Auswirkung:**

Logische Verknüpfungen von BOOL-Variablen, deren Werte von Fremdsystemen stammen, können andere als die erwarteten Ergebnisse liefern

**Bedingung:**

Die Ursache ist die Codierung der BOOL-Werte im externen System, die von der im HIMA System abweicht.

**Abhilfe:**

Es gibt zwei Möglichkeiten der Abhilfe:

- das externe System liefert definiert nur die Werte 0 für FALSE und 1 für TRUE
  - ins Anwenderprogramm wird für alle entsprechenden BOOL-Variablen eine Korrekturschaltung eingefügt, die den Wert auf 0 bzw. 1 normiert:
- unnormierte Variable -> Baustein AtoByte -> Baustein AtoBOOL -> normierte Variable

- **Auswirkung:**

Bei Berechnungen mit Variablen vom Datentyp REAL oder LREAL können die Zykluszeiten insbesondere bei Verwendung von trigonometrischen Funktionen extrem schwanken

**Bedingung:**

s.o.

**Abhilfe:**

Für die Bemessung der Watchdog-Zeit ist es notwendig, die Zykluszeit unter realistischen Bedingungen zu bestimmen. Siehe Sicherheitshandbuch: Kapitel: „Genaue Bestimmung der Watchdogzeit.“

- Auswirkung:  
Online-Test und Offline-Simulation zeigen den Wert von Systemvariablen des Anwenderprogramms nicht an:
  - OLT-Feld ist leer
  - Der Wert von digitalen Systemvariablen ist nicht durch die Farbe der entsprechenden Linien dargestellt
  - Die Spalte Prozess-Wert im Register System-Variablen der Objektauswahl ist leer
  - Der Force-Editor enthält keine Systemvariablen

Bedingung:

s. o.

Abhilfe:

Die meisten Informationen sind an anderer Stelle, z. B. im Control Panel, ersichtlich. Zur Anzeige im

OLT die Systemvariable mit einer Variablen verbinden und an die Variable ein OLT-Feld anschließen.

Forcen ist nur möglich, wenn die Systemvariable mit einer Variablen verbunden ist.

- Auswirkung:  
Wertänderungen von VAR\_INPUT-Variablen benutzerdefinierter Funktionsbausteine SILworX behandelt VAR\_INPUT-Variablen bei benutzerdefinierten Funktionsbausteinen je nach Beschaltung der Eingänge unterschiedlich:
  - Bei Beschaltung der Eingänge mit Variablen von Standard-Datentypen wird der Wert der Variablen an eine baustein-lokale Kopie übergeben (Call by value).
  - Bei Beschaltung der Eingänge mit Variablen von benutzerdefinierten Datentypen wird die Referenz auf die Variable übergeben (Call by reference).

Bedingung:

Dieses Verhalten kann zu Fehlern führen wenn alle folgenden Bedingungen zutreffen:

- Die Quelle einer VAR\_INPUT-Variable ist eine VAR\_EXTERNAL-Variable.
- Die selbe Quelle der VAR\_INPUT-Variable wird im aufgerufenen Funktionsbaustein gleichzeitig als VAR\_EXTERNAL-Variable verwendet.

Wird der Wert der VAR\_EXTERNAL-Variablen im Funktionsbaustein verändert, hat anschließendes Lesen der entsprechenden VAR\_INPUT-Variablen im Funktionsbaustein folgende Auswirkungen:

- Bei benutzerdefiniertem Datentyp werden die aktuellen Werte gelesen.
- Bei elementarem Datentyp werden die alten Werte gelesen, die bei Beginn der Abarbeitung der Bausteininstanz gültig waren.

Abhilfe:

VAR\_EXTERNAL-Variablen sollten nicht gleichzeitig als Quelle einer VAR\_INPUT von Instanzen dieser POE verwendet werden.

- **Auswirkung:**  
Das Dokumentenmanagement kann den Inhalt der Online-Hilfe einer benutzerdefinierten POE nicht ausdrucken.

**Bedingung:**

-

**Abhilfe:**

Anzeigen der Online-Hilfe und Ausdrucken der einzelnen Themen aus Windows.

- **Auswirkung:**  
Es kann nicht ausgeschlossen werden, dass sich zwischen SILworX Versionen Schlüsselbegriffe in den Export-/Import-Dateien (.CSV, .XML) ändern. Dadurch importiert SILworX die entsprechenden Daten als Standardwert und gibt eine Fehlermeldung aus.

**Bedingung:**

Beispiel: In Versionen vor V5.xx ist der Datentyp für die Spracheinstellung English als «Data Type» gekennzeichnet, ab V5.xx als «Data type». Beim Import einer Export-Datei einer Version vor V5.xx legt SILworX alle Variablen mit dem Standard-Datentyp BOOL an.

**Abhilfe:**

In den zu importierenden Dateien die entsprechenden Schlüsselwörter anpassen

- **Auswirkung:**  
Wird bei bestehendem System-Login die Diagnoseansicht geöffnet und die Verbindung getrennt, dann bietet SILworX beim Versuch, die Verbindung neu aufzubauen, das Modul-Login an

**Bedingung:**

Hardware-Login, Detailansicht zu Baugruppe öffnen.

Diagnose zu dieser Baugruppe in zweitem Fenster öffnen.

Anschließend Verbindung-Trennen:

Diagnose bietet nach Verbindungsverlust nur den Modul-Login an,  
Detailansicht bietet System-Login an.

**Abhilfe:**

Wenn Modul-Login-Dialog für Diagnose einmal geöffnet wurde, müssen alle Online-Ansichten dieses Moduls (Diagnose und Modul-Online-Ansicht) geschlossen und dann wieder geöffnet werden, damit sie wieder über das System ausgelesen werden können.

- **Auswirkung:**  
Für HIMatrix Geräte vor F\*03 (z. B. F30 01, F31 02, F35 01, F35 012 und F60 CPU 01) wird der Force-Status in Force-Ansichten für lokales Forcen (Oberhalb der Force-Tabelle) mit regulären Werten angezeigt, als wären die Informationen tatsächlich verfügbar, sie sind aber ohne Funktion. Im Einzelnen sind dies: Force-Zustand, Geforcte Variablen, Verbleibende Force-Dauer und Force-Timeout-Reaktion

**Bedingung:**

-

**Abhilfe:**

Verhalten ist in Online Hilfe beschrieben

## 6 Upgrade von einer Vorversion

Projektdaten aus vorherigen Versionen können in V10.64 weiterverwendet werden.

Dabei werden keine CRC-Änderungen auftreten, solange die Versionseinstellung minimale Konfigurationsversion einer Ressource nicht verändert wird. SILworX hält die CRCs kompatibel, sofern keine Änderungen erfolgen bzw. keine neuen Features verwendet werden.

Das Upgrade von einer Version ab V2.36 auf V10.64 ist folgendermaßen durchzuführen:

- Es sollte eine Sicherung des Projektes durchgeführt werden.
- Vor der Konvertierung für alle Ressourcen Code generieren. Durch den Export und Import der Konfiguration im Versionsvergleich lassen sich eventuelle Abweichungen bei der Konvertierung feststellen.
- Projekt in V10.64 öffnen und konvertieren.
- Da die Konvertierung sehr umfangreich ist, nach der Konvertierung die Projektintegrität prüfen.
- In V10.64 eine Codegenerierung durchführen, um festzustellen, ob Fehler auftreten und/oder sich CRCs ändern. Im Versionsvergleich kann dies mit der importierten Konfiguration festgestellt werden.
- Erkannte Fehler beseitigen und erneut Code generieren, um CRC-Änderungen festzustellen.
- Liegen keine CRC-Änderungen vor, ist der Übergang erfolgreich abgeschlossen.
- Liegen CRC-Änderungen vor, prüfen, ob diese akzeptabel sind.
- Sind die Änderungen akzeptabel, ist der Übergang erfolgreich abgeschlossen.
- Falls diese inakzeptabel sind, mit der entsprechenden Vorversion weiterarbeiten.

Hinweise zur Konvertierung:

- Die Konvertierung von Versionen vor V2.36 ist in den Release-Notes zu V2.36 beschrieben.
- Die Konvertierung kann bei sehr großen Projekten bis zu mehreren Stunden dauern.

### 6.1 Referenzen

- SILworX Online-Hilfe
- SILworX Erste-Schritte-Handbuch, HI 801 102 D
- Kommunikationshandbuch, HI 801 100 D
- HIPRO-S V2 Handbuch, HI 800 722 D
- ISOfast Handbuch, HI 801 464 D
- Modbus V2 Handbuch, HI 801 474 D
- X-OPC-Server Handbuch, HI 801 479 D

## Release-Notes

---

### HI 801 501 D

Für weitere Informationen kontaktieren Sie:

**HIMA Paul Hildebrandt GmbH**

Albert-Bassermann-Str. 28  
68782 Brühl, Germany

Telefon: +49 6202 709-0  
Fax +49 6202 709-107  
E-Mail: [info@hima.com](mailto:info@hima.com)

Erfahren Sie online mehr über HIMA-Lösungen:

 [www.hima.com/de//](http://www.hima.com/de//)



[www.hima.com](http://www.hima.com)