Manual

Communication

Configuration in SILworX

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

HIQuad®, HIQuad®X, HIMax®, HIMatrix®, SILworX®, XMR®, HICore® and FlexSILon® are registered trademarks of HIMA Paul Hildebrandt GmbH.

All of the technical specifications and information in this manual were prepared with great care and effective control measures were employed for their compilation. For questions, please contact HIMA directly. HIMA appreciates any suggestion on which information should be included in the manual.

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

All the current manuals can be obtained upon request by sending an e-mail to: documentation@hima.com.

Contact

HIMA Paul Hildebrandt GmbH
P.O. Box 1261
68777 Brühl, Germany
Phone: +49 6202 709-0
Fax: +49 6202 709-107
E-mail: info@hima.com

Document designation	Description
HI 801 100 D, Rev. 12.00 (2024)	German original document
HI 801 101 E, Rev. 12.00.00 (2027)	English translation of the German original document

Table of Contents

1 Introduction

The communication manual for safety-related HIMA systems provides an overview of the protocols available and the physical properties of the Ethernet and fieldbus interfaces. For protocols not described in this manual, separate manuals are available, see Table 2.

The following conditions must be met to safely install and start up the system and to ensure safety during their operation and maintenance:

- Knowledge of regulations.
- Proper technical implementation of the safety instructions detailed in this manual performed by qualified personnel.

HIMA will not be held liable for severe personal injuries, damage to property or the environment caused by any of the following:

- Unqualified personnel working on or with the systems.
- De-activation or bypassing of safety functions.
- Failure to comply with the instructions detailed in this manual.

HIMA develops, manufactures and tests the HIMA systems in compliance with the pertinent safety standards and regulations. The use of the systems is only allowed if the following requirements are met:

- They are only used for the intended applications.
- They are operated under the specified environmental conditions.

1.1 Structure and Use of This Manual

The manual contains the following chapters:

- Introduction
- Safety
- Product description
- safeethernet
- SNTP
- HART
- General

Additionally, the following documents must be taken into account:

Name	Content	Document no.
HIMax system manual	Hardware description HIMax system	HI 801 001 E
HIMax safety manual	Safety function HIMax systems	HI 801 003 E
HIMatrix safety manual	Safety function HIMatrix systems	HI 800 023 E
HIMatrix compact system manual	Hardware description HIMatrix compact system	HI 800 141 E
HIMatrix modular system manual	Hardware description HIMatrix modular F 60 system	HI 800 191 E
HIQuad X system manual	Hardware description HIQuad X system	HI 803 211 E
HIQuad X safety manual	Safety function HIQuad X system	HI 803 209 E
Automation security manual	Description of automation security aspects related to the HIMA systems	HI 801 373 E
SILworX first steps manual	Introduction to SILworX.	HI 801 103 E

Table 1: Additional Applicable Manuals

All the current manuals can be obtained upon request by sending an e-mail to: documentation@hima.com. The documentation is available for registered HIMA customers in the download area https://www.hima.com/en/downloads/.

1.2 Target Audience

This document is aimed at the planners, design engineers, programmers and the persons authorized to start up, operate and maintain the automation systems. Specialized knowledge of safety-related automation systems is required.

1.3 Writing Conventions

To ensure improved readability and comprehensibility, the following writing conventions are used in this document:

Bold To highlight important parts.
Names of buttons, menu functions and tabs that can be clicked and used in the programming tool.

Italics Parameters and system variables, references.

`Courier` Literal user inputs.

RUN Operating states are designated by capitals.

Chapter 1.2.3 Cross-references are hyperlinks even if they are not specially marked.
In the electronic document (PDF): When the mouse pointer hovers over a hyperlink, it changes its shape. Click the hyperlink to jump to the corresponding position.

Safety notices and operating tips are specially marked.

1.3.1 Safety Notices

Safety notices must be strictly observed to ensure the lowest possible risk.

The safety notices are represented as described below.

- Signal word: warning, caution, notice.
- Type and source of risk.
- Consequences arising from non-observance.
- Risk prevention.

The signal words have the following meanings:

- Warning indicates hazardous situations which, if not avoided, could result in death or serious injury.
- Caution indicates hazardous situation which, if not avoided, could result in minor or moderate injury.
- Notice indicates a hazardous situation which, if not avoided, could result in property damage.

⚠ SIGNAL WORD

Type and source of risk!

Consequences arising from non-observance.

Risk prevention.

NOTICE

Type and source of damage!

Damage prevention.

1.3.2 Operating Tips

Additional information is structured as presented in the following example:

i The text giving additional information is located here.

Useful tips and tricks appear as follows:

TIP The tip text is located here.

1.4 Safety Lifecycle Services

HIMA provides support throughout all the phases of the plant's safety lifecycle, from planning and engineering through commissioning to maintenance of safety and security.

HIMA's technical support experts are available for providing information and answering questions about our products, functional safety and automation security.

To achieve the qualification required by the safety standards, HIMA offers product or customer-specific seminars at HIMA's training center or on site at the customer's premises. The current seminar program for functional safety, automation security and HIMA products can be found on HIMA's website.

Safety Lifecycle Services:	
Onsite+ / On-Site Engineering	In close cooperation with the customer, HIMA performs changes or extensions on site.
Startup+ / Preventive Maintenance	HIMA is responsible for planning and executing preventive maintenance measures. Maintenance actions are carried out in accordance with the manufacturer's specifications and are documented for the customer.
Lifecycle+ / Lifecycle Management	As part of its lifecycle management processes, HIMA analyzes the current status of all installed systems and develops specific recommendations for maintenance, upgrading and migration.
Hotline+ / 24 h Hotline	HIMA's safety engineers are available by telephone around the clock to help solve problems.
Standby+ / 24 h Call-Out Service	Faults that cannot be resolved over the phone are processed by HIMA's specialists within the time frame specified in the contract.
Logistics+/ 24 h Spare Parts Service	HIMA maintains an inventory of necessary spare parts and guarantees quick, long-term availability.

Contact details:	
Safety Lifecycle Services	https://www.hima.com/en/about-hima/contacts-worldwide/
Technical Support	https://www.hima.com/en/products-services/support/
Seminar Program	https://www.hima.com/en/products-services/seminars//

2 Safety

All safety information, notes and instructions specified in this document must be strictly observed. The product may only be used if all guidelines and safety instructions are adhered to.

The product is operated with SELV or PELV. No imminent risk results from the product itself. Use in the Ex zone is only permitted if additional measures are taken.

2.1 Intended Use

To use the HIMA controllers, all pertinent requirements must be met, see additionally applicable manuals listed in Table 1.

2.2 Residual Risk

No imminent risk results from a HIMA system itself.

Residual risk may result from:

- Faults related to engineering.
- Faults in the user program.
- Faults related to the wiring.

2.3 Safety Precautions

Observe all local safety requirements and use the protective equipment required on site.

2.4 Emergency Information

A HIMA system is a part of the safety equipment of an overall system. If the controller fails, the system enters the safe state.

In emergencies, no action that may prevent the HIMA system from operating safely is permitted.

2.5 Automation Security for HIMA Systems

The objectives of automation security are data confidentiality, integrity and availability. Targeted attacks are to be expected for automation security. In particular, potential targets of attacks are interfaces such as described in this manual.

⚠ WARNING

Physical injury possible due to unauthorized manipulation of the controller!

Protect the controller against unauthorized access!

Users are responsible for implementing the necessary measures in a way suitable for the plant!

Careful planning should identify the measures to implement. The required measures are to be implemented after the risk analysis is completed. Such measures can include:

- Meaningful allocation of user groups.
- Maintained network maps help to ensure that secure networks are permanently separated from public networks and, if required, only a well-defined connection exists (e.g., via a firewall or a DMZ).
- Use of appropriate passwords.

A periodical review of the security measures is recommended, e.g., every year.

For further details, refer to the HIMA automation security manual (HI 801 373 E).

3 Product Description

Using the provided protocols, HIMA controllers can be connected to one another or to controllers from other manufacturers. The protocols are configured in the SILworX programming tool.

Manufacturer-independent standard protocols are available to ensure optimal integration of field devices and control systems into the HIMA systems. Both Ethernet and fieldbus protocols may be used. The standard protocols are interference-free with respect to the safe processor system of the HIMAsystems.

The following protocols are available for the HIMA systems:

Protocol	SIL[1]	HIMax	HIQuad X	HIMatrix	Chapter or manual
safeethernet	4	X	X	X	Chapter 4
SNTP	-	X	X	X	Chapter 5
HART Protocol	-	X	--	--	Chapter 6
HIMA X-OPC Server[2]	-	X	X	X	HI 801 480 E
HIMA OPC UA Server	-	X	X	X	HI 801 551 E
ISOfast	3	--	--	X	HI 801 465 E
Send/Receive TCP	-	X	--	X	HI 801 524 E
HIPRO-S V2	3	X	X	X	HI 800 723 E
PROFINET IO controller	-	X	--	X	HI 801 523 E
PROFINET IO device	-	X	--	X	
PROFIsafe host	3	X	--	X	
PROFIsafe F-device	3	X	--	X	
PROFIBUS DP master	-	X	--	X	
PROFIBUS DP slave	-	X	X	X	
Modbus master	-	X	X	X	HI 801 522 E
Modbus slave set	-	X	X	X	
Modbus slave set V2	-	X	X	X	HI 801 475 E
Synchronous serial interface (SSI)	-	X	--	X	HI 801 521 E
ComUserTask[3]	-	X	X	X	

[1] --: No SIL.
3: SIL 3 in accordance with IEC 61508-2:2010, IEC 61784-3:2019.
4: SIL 4 in accordance with IEC 61508-2:2010, IEC 61784-3:2019 and EN 50159:2010, see Chapter 4.

[2] The HIMA X-OPC Server is installed on a host PC and is used as a transfer interface for up to 255 HIMA controllers and third-party systems that have an OPC interface.

[3] In the ComUserTask, a C program of the user can be implemented and connected to various communication interfaces of the COM module.

Table 2: Protocols Available for the HIMA Systems

The safety-related protocols are operated on the corresponding processor module of the HIMA system. The amount of process data is limited by the available free memory for global process data on the processor module:

- HIMax, HIMatrix, HIQuad X = 512 kBytes.

i The memory for global process data is used for all variables of the HIMA system (e.g,. protocol, user program and system variables). If the available memory space is depleted, the HIMA system rejects a configuration during the download or reload and informs the user in the SILworX logbook.

Many standard protocols only ensure a non-safety-related data transmission. The non-safe data may only be used for safety-related functions under the responsibility of the user if sufficient additional measures have been taken.

⚠ WARNING

Use of non-safe import data in safety-related functions!

Physical injury possible due to usage of non-safe import data!

Do not use data imported from unsafe sources for the user program's safety-related functions.

3.1 HIMA System Quantity Structure for Non-Safety-Related Protocols

Non-safety-related protocols (NSIP) are operated on the corresponding communication module (COM module) of the HIMA systems.

Properties	HIMax	HIQuad X	Description
System view			The pictures are examples of the respective system family. The figures depict a HIMax and a HIQuad X H51X.
Communication modules per HIMA controller	With X-CPU 01: 1…20 X-COM 01 With X-CPU 31: 1…4 X-COM 01	H51X: 1…10 F-COM 01 H41X: 1…2 F-COM 01	NSIP are run on the communication modules.
Ethernet and fieldbus interfaces	On the X-COM 01	On the F-COM 01	For details, refer to Table 5.
Maximum number of NSIP	▪ 201) for each HIMax controller. ▪ 61) for each X-COM module.	▪ 20[1] for each HIQuad X. ▪ 5[1] for each F-COM 01.	Available NSIP, see Table 2.
Process data volume[1][2] of all NSIP within a controller	Send 128 kB Receive 128 kB	Send 64 kB Receive 64 kB	The maximum process data volume of the controller must not be exceeded. If it is exceeded, the controller configuration is rejected during the load process.

Properties	HIMatrix	Description
System view		The picture is an example of the respective system family. It shows an F30.
Communication modules per HIMA controller	Integrated communication module	NSIP are run on the communication modules.
Ethernet and fieldbus interfaces	On the controller	For details, see Table 5.
Maximum number of NSIP	6[1]	Available NSIP, see Table 2.
Process data volume[1][2] of all NSIP within a controller	Send 64 kB Receive 64 kB	The maximum process data volume of the controller must not be exceeded. If it is exceeded, the controller configuration is rejected during the load process.

[1] X-OPC Server, SNTP client and SNTP server are not taken into account in this calculation.

[2] The process data volume of non-safety-related protocols (NSIP) includes the exchanged data and the system variables of non-safety-related protocols and of PROFIsafe.

Table 3: HIMA System Quantity Structure for Non-Safety-Related Protocols

3.2 Protocol Registration and Activation

The protocols specified below are available for HIMA systems and can be activated as follows:

Protocol	Interfaces	HIMax	HIQuad X	HIMatrix
HIMA safeethernet	Ethernet	I	I	I
SNTP	Ethernet	I	I	I
HART protocol	Ethernet	I	--	--
HIMA X-OPC Server (runs on a host PC)	Ethernet	II	II	II
HIMA OPC UA Server	Ethernet	II	II	II
ISOfast	Ethernet	--	--	II
Send/Receive TCP	Ethernet	II	--	II
HIPRO-S V2	Ethernet	II	II	II
PROFINET IO controller	Ethernet	II	--	II
PROFINET IO device	Ethernet	II	--	II
PROFIsafe F-Host1)	Ethernet	II	--	II
PROFIsafe F-Device1)	Ethernet	II	--	II
PROFIBUS DP master	Fieldbus	III	--	III
PROFIBUS DP slave	Fieldbus	III	II	III
Modbus master Eth	Ethernet	II	II	II
Modbus slave Eth	Ethernet	II	II	II
Modbus master RS485	Fieldbus	IV	II	IV
Modbus slave RS485	Fieldbus	IV	II	IV
Synchronous serial interface (SSI)	Fieldbus	IV	--	IV
ComUserTask	Ethernet, Feldbus	IV	II	IV
I These protocols are activated by default. II A license (software activation code) must be purchased for these protocols. III These protocols are activated by installing a fieldbus submodule. IV A license (software activation code) and, if required, the corresponding fieldbus submodule must be purchased for these protocols.				
1) Additional PROFINET license needed				

Table 4: Protocol Registration and Activation

The software activation code with the required licenses is generated on the HIMA website using the system ID of the controller. To this end, follow the instructions provided on the HIMA website www.hima.com-> Products & Services -> Product Registration-> Options SILworX.

i The license is intrinsically bound to the system ID. A license can only be used once for a specific system ID. For this reason, only activate the code when the system ID has been uniquely defined.

A software activation code may include a maximum of 32 licenses. It is also possible to specify multiple activation codes in the license management. A maximum of 64 licenses may be loaded into one controller.

i If a Modbus master RS485 is operated on one COM through multiple interfaces, it is still considered a single Modbus master instance. It requires therefore only one license.

To enter the software activation code in SILworX

1. In the structure tree, select Configuration, Resource, License Management.
2. Right-click License Management and select New, License Key from the context menu.
   ☑ A new license key is created.
3. Right-click the license key and select Properties from the context menu.
4. Enter the new software activation code in the Activation Code field.

---

i

Order the license on time!
All functions requiring a license (e.g., protocols) can be tested without license for 5000 operating hours.
If functions are operated with no valid license, the Error LED (for HIMax/HIMatrix and HIQuad X) is lit.
After 5000 operating hours, the function (e.g., protocols) continues until the controller is stopped. Afterwards, the user program cannot be started without a valid license for the features used in the project (faulty configuration).

3.3 Ethernet Interfaces

The Ethernet interfaces of CPU and COM in the HIMA systems can be used for communication with external systems and programming. The Ethernet interfaces can simultaneously process multiple protocols, excepted from the system bus interfaces of the X-SB 01, X-CPU 31 and F-IOP 01 modules.

The use of these interfaces is described in the respective system manual.

Each CPU and COM module has a a freely configurable IPv4 address and an Ethernet switch.

To transfer data, the Ethernet switch establishes a targeted connection between two communication partners. This prevents collisions and reduces the load on the network.

For targeted data forwarding, a MAC/IP address assignment table (ARP cache) is generated in which the MAC addresses are assigned to specific IP addresses. From now on, data packets are only forwarded to the IP addresses specified in the ARP cache.

---

i

Replacement of CPU or COM module with identical IP address.
If a device has its ARP Aging Time set to 5 minutes and its MAC Learning set to Conservative, its communication partner does not adopt the new MAC address until a period of 5 to 10 minutes after the module is replaced. Until the new MAC address has been adopted, no communication is possible using the replaced device.
In addition to the configurable ARP Aging Time, the user must wait at least the non-configurable MAC Aging Time of the switch (approx. 10 seconds) before the replaced device is able to communicate again.

3.3.1 HIMax Ethernet Interfaces

The following table shows the HIMax Ethernet interfaces for communication with external system:

Property	HIMax X-CPU 01	HIMax X-CPU 31	HIMax X-COM 01
Ports	4	2 for protocols 2 for system bus UP/DOWN	4
Transmission standard	10/100/1000 Base-T, half and full duplex	10/100 Base-T Half and full duplex	
Autonegotiation	Yes		
Autocrossover	Yes		
Connection socket	RJ45		
IP address	Freely configurable[1]		
Subnet mask	Freely configurable[1]		
Supported protocols	safeethernet, X-OPC (DA & A+E), HIPRO-S V2 Programming and debugging tool (PADT), SNTP		
	--	--	Standard protocols[2]
[1] The general rules for assigning IP address and subnet masks must be adhered to. [2] In this manual, the term standard protocols designates protocols that are used to connect to external systems.			

Table 5: HIMax Ethernet Interfaces

3.3.2 HIQuad X and HIMatrix Ethernet Interfaces

The following table shows the HIQuad X and HIMatrix Ethernet interfaces for communication with external system.

Property	HIQuad X F-CPU 01	HIQuad X F-COM 01	HIMatrix Steuerung
Ports	2	2	4
Transmission standard	10BASE-T/ 100BASE-Tx, Half and full duplex		
Autonegotiation	Yes		
Autocrossover	Yes		
Connection socket	RJ45		
IP Address	Freely configurable[1]		
Subnet Mask	Freely configurable[1]		
Supported protocols	safeethernet, X-OPC (DA & A+E), HIPRO-S V2 Programming and debugging tool (PADT), SNTP		
	--	Standard protocols[2]	Standard protocols[2]
[1] The general rules for assigning IP address and subnet masks must be adhered to. [2] In this manual, the term standard protocols designates protocols that are used to connect to external systems.			

Table 6: HIQuad X and HIMatrix Ethernet Interfaces

3.3.3 Configuring the Ethernet Interfaces

The Ethernet interfaces are configured in SILworX in the detail view of the CPU or COM module.

For HIMA systems, the Speed Mode and Flow Control Mode parameters are set to Autoneg by default.

i Communication loss!

With an inappropriate Ethernet parameters setting, the device might no longer be reachable. Reset the device!

To open the CPU/COM module detail view

1. In the structure tree, select Configuration, Resource, Hardware.
2. Right-click and select Edit from the context menu to open the Hardware Editor.
3. Right-click CPU/COM Module and select Detail View from the context menu to open the detail view.

i The parameters set in the properties of the CPU/COM module are not available for the HIMA system communication, until they have been re-compiled with the user program and transferred to the controller.

3.3.3.1 The Module Tab

The Module tab contains the following parameters:

Designation	Description
Name	Module name.
Activating Max. µP Budget for HH Protocol	▪ Activated: Use CPU load limit from the *Max. µP Budget for HH Protocol [%]* field. ▪ Deactivated: Do not use the CPU load limit for safeethernet.
Max. µP Budget for HH Protocol [%]	Maximum CPU load of the module that can be used for processing the safeethernet protocol. i The maximum load must be distributed among all the implemented protocols that use this communication module.
Code Generation	This parameter can only be set for HIMax and HIMatrix systems since HIQuad X is only available as of V10. Prior to V6: Setting compatible with existing projects. V6 and higher: Recommended setting for new projects, especially if safeethernet connections are routed via this communication module. Changes to the safeethernet connection can be loaded by performing a reload.
IP Address	IP address of the Ethernet interface. Default value: 192.168.0.99
Subnet Mask	32-bit address mask to split up the IP address into network and host address.
Standard Interface	Activated: The interface is used as standard interface for system login. Default setting: Deactivated
Default Gateway	IP address of the default gateway. Default value: 0.0.0.0

Designation	Description
ARP Aging Time [s]	A processor or COM module stores the MAC addresses of the communication partners in a MAC/IP address assignment table (ARP cache). The MAC address remains stored in the ARP cache if messages from the communication partner are received within 1x…2x *ARP Aging Time*. The MAC address is erased from the ARP cache if no messages from the communication partner are received within 1x…2x *ARP Aging Time*. The typical value for the *ARP Aging Time* in a local network ranges from 5…300 s. The contents of the ARP cache cannot be read out. Range of values: 1…3600 s Default value: 60 s **Note:** If routers or gateways are used, the *ARP Aging Time* must be adjusted (increased) due to the additional time required for two-way transmission. If the *ARP Aging Time* is too low, the MAC address of the communication partner is erased from the ARP cache and communication is delayed or interrupted. For an efficient performance, the *ARP Aging Time* value must be greater than the receive timeout set for the protocols in use.
MAC Learning	*MAC Learning* and *ARP Aging Time* are used to set how quick the Ethernet switch should learn the MAC address. The following settings are possible: ▪ Conservative (recommended) If the ARP cache already contains MAC addresses of communication partners, these are locked and cannot be replaced by other MAC addresses for at least 1 *ARP Aging Time* and a maximum of 2 *ARP Aging Time* periods. ▪ Tolerant When a message is received, the IP address contained in the message is compared to the data in the ARP cache, and the MAC address stored in the ARP cache is immediately overwritten with the MAC address from the message. Tolerant must be used if the availability of communication is more important than the authorized access to the controller. Default setting: Conservative
ICMP Mode	The Internet Control Message Protocol (ICMP) allows the higher protocol layers to detect error states on the network layer and optimize the transmission of data packets. Message types of Internet Control Message Protocol (ICMP) supported by the CPU module: ▪ No ICMP Responses All ICMP commands are deactivated. This ensures a high degree of safety against potential sabotage that might occur over the network. ▪ Echo Response If Echo Response is activated, the node responds to a ping command. It is thus possible to determine if a node can be reached. Safety is still high. ▪ Host Unreachable Not important for the user. Only used for testing at the manufacturer's facility. ▪ All Implemented ICMP Responses All ICMP commands are activated. This allows a more detailed diagnosis of network malfunctions. Default setting: Echo Response

Table 7: Configuration Parameters

3.3.3.2 The Routings Tab

The Routings tab contains the routing table. This table is empty if the module is new. A maximum of 8 routing entries are possible.

Designation	Description
Name	Designation of the routing settings.
IP Address	Target IP address of the communication partner (with direct host routing) or network address (with subnet routing). Range of values: 0.0.0.0…255.255.255.255 Default value: 0.0.0.0
Subnet Mask	Define the target address range for a routing entry. 255.255.255.255 (with direct host routing) or subnet mask of the addressed subnet. Range of values: 0.0.0.0…255.255.255.255 Default value: 255.255.255.255
Gateway	IP address of the gateway to the addressed network. Range of values: 0.0.0.0…255.255.255.255 Default value: 0.0.0.1

Table 8: Routing Parameters

3.3.3.3 The Ethernet Switch Tab

The Ethernet Switch tab contains the following parameters:

Designation	Description
Name	Port number as printed on the housing; per port, only one configuration may exist. Range of values: 1…4
Speed [MBit/s]	10 Mbit/s: Data rate 10 Mbit/s 100 Mbit/s: Data rate 100 Mbit/s 1000 Mbit/s: Data rate 1000 Mbit/s (X-CPU 01 module only). Autoneg (10/100/1000): Automatic baud rate setting. Default value: Autoneg
Flow Control	Full duplex: Simultaneous communication in both directions. Half duplex: Communication in one direction. Autoneg: Automatic communication control. Default value: Autoneg
Autoneg also with Fixed Values	The Advertising function (forwarding the speed and flow control properties) is also performed if the parameters Speed and Flow Control have fixed values. This allows other devices whose ports are set to Autoneg to detect the port setting.
Limit	Limit the inbound multicast and/or broadcast packets. Off: No limitation. Broadcast: Limit broadcast (128 kbit/s). Multicast and Broadcast: Limit multicast and broadcast packets (1024 kbit/s). Default value: Broadcast

Table 9: Ethernet Switch Parameters

3.3.3.4 The VLAN Tab (Port-Based VLAN)

For configuring the use of port-based VLAN, see also Chapter 3.3.5.

i If VLAN is to be supported, port-based VLAN must be off to enable each port to communicate with the other switch ports.

For each port of a switch, the user can define to which other ports of the switch received Ethernet frames may be sent to.

The table in the VLAN tab contains entries through which the connection between two ports can be set to active or inactive.

Name	Eth1	Eth2	Eth3	Eth4
Eth1				
Eth2	Active			
Eth3	Active	Active		
Eth4	Active	Active	Active	
CPU	Active	Active	Active	Active

Table 10: VLAN Tab

Default setting: All connections between ports are set to Active

3.3.3.5 The LLDP Tab

LLDP (Link Layer Discovery Protocol) periodically sends information on the own device via multicast (e.g., MAC address, device name, port number) and receives the same information from the neighboring devices.

LLDP uses the following values depending on whether PROFINET is configured on the communication module:

PROFINET on the COM module	Chassis ID	TTL (Time to Live)
Used	Device name	20 s
Not used	MAC Address	120 s

Table 11: LLDP Values for Profinet

The processor and communication modules support LLDP on the Eth1, Eth2, Eth3 and Eth4 ports.

The following parameters define how a given port should work:

Off	LLDP is disabled on this port.
Send	LLDP sends LLDP Ethernet frames, received LLDP Ethernet frames are deleted without being processed.
Receive	LLDP sends no LLDP Ethernet frames, but received LLDP Ethernet frames are processed.
Send/Receive	LLDP sends and processes received LLDP Ethernet frames.

Default setting: Off.

3.3.3.6 The Mirroring Tab

Mirroring is used to configure whether the module should duplicate Ethernet packets on a given port such that they can be read from a device connected to that port, e.g., for test purposes.

The following parameters define how a given port should work:

Off	This port does not participate in the mirroring process.
Egress:	Outgoing data of this port are duplicated.
Ingress:	Incoming data of this port are duplicated.
Ingress/Egress:	Incoming and outgoing data of this port are duplicated.
Dest Port:	Duplicated data are sent to this port.

Default setting: Off.

3.3.4 Network Ports in Use for Ethernet Communication

UDP ports	Use
123	SNTP (time synchronization between controller and remote I/O, and external devices).
502	Modbus salve (can be changed by the user).
6010	safeethernet and OPC.
8000	Programming and operation with SILworX..
8001	Port on the remote I/O for configuring the remote I/O using the controller.
8004	Port on the controller for configuring the remote I/O using the controller.
34964	PROFINET endpoint mapper (required for establishing the connection).
49152	PROFINET RPC server.
49153	PROFINET RPC client.
Xxx	ComUserTask assigned by the user. May not be used with another protocol.

Table 12: Network Ports (UDP Ports) in Use

TCP ports	Use
502	Modbus salve (can be changed by the user).
Xxx	TCP SR assigned by the user.
Xxx	ComUserTask assigned by the user. May not be used with another protocol.

Table 13: Network Ports (TCP Ports) in Use

3.3.5 Separating Switch Ports via VLAN

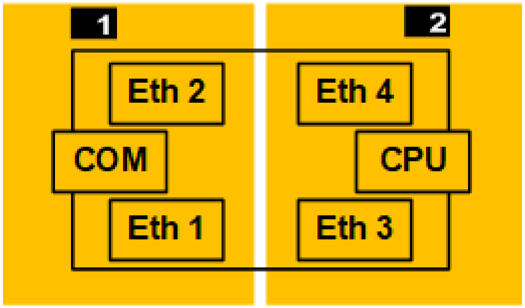
VLAN settings can be used to divide the available switch ports according to the required application. It is therefore possible in HIMatrix to establish a connection with two IP addresses or to separate safe communication through the CPU from non-safe communication through the COM.

The switch port is configured in SILworX in the detail view of the CPU or COM module, see Chapter 3.3.3.

For HIMatrix, HIMA recommends separating the CPU and COM. The example below can be adapted to the application-specific requirements.

	Eth1	Eth2	Eth3	Eth4	COM
Eth1					
Eth2	Active				
Eth3	Inactive	Inactive			
Eth4	Inactive	Inactive	Active		
COM	Active	Active	Inactive	Inactive	
CPU	Inactive	Inactive	Active	Active	Inactive

Table 14: VLAN Tab



1 Eth 1 and Eth 2 in the unprotected area via the COM for non-safety-related protocols.

2 Eth 3 and Eth 4 in the protected area via the CPU for safeethernet communication with the remote I/Os and other HIMA PES..

Figure 1: Example of Switch Ports Separated via VLAN

i If all the Ethernet port connections to the processor of the controller have been blocked by the VLAN configuration, the controller must be reset. Afterwards, the controller is once again accessible via the default IP address.

i Connection blockades in networks separated via VLAN if these networks are not completely separated, e.g.,connected through a common external switch.
The internal switch in HIMatrix controllers includes a common MAC<->switch port assignment table for the CPU and the COM. When Ethernet frames arrive from a network that is not completely separated, the MAC<->switch port assignment table of the internal switch must be continuously relearned. This results in alternating blockades of the corresponding Ethernet frames to the CPU and the COM.

3.4 Fieldbus Interfaces

The fieldbus submodules allow communication via the fieldbus interfaces of the HIMax X-COM 01, HIQuad X, F-COM 01, as well as the HIMatrix controllers F30, F35 and the F60 CPU 01.

For the HIMax and HIMatrix controllers, the fieldbus submodules are optional and must be installed by the manufacturer. Ex-factory, the FB3 fieldbus interface of the HIMatrix controllers includes an RS485 for Modbus (master or slave) or ComUserTask.

Users can configure the fieldbus interface transmission standards in SILworX for the HIQuad X controllers. The pins for the FB1and FB2 interfaces of the F-COM 01 module are automatically assigned once this configuration has been loaded into the HIQuad X controller.

The fieldbus protocols may only be used for safety-related functions under the responsibility of the user if sufficient additional measures have been taken.

The system does not support programming using these interfaces.

3.4.1 Registration and Activation

The communication options are activated in accordance with the protocol, see Chapter 3.2.

3.4.2 Installation of the Fieldbus Submodules

The fieldbus submodules are optional and must be installed by the manufacturer. Additionally, the protocols used must be activated. Additionally, the protocols used must be partially activated.

3.4.2.1 Part Number Structure

The following sections present how the part number for the HIMax X-COM 01 or a HIMatrix controller changes if fieldbus interfaces are used.

Numbers are allocated to the fieldbus to create the part numbers, see Table 15.

Options for FB1 and FB2	Designation	Fieldbus submodule description
0	--	No fieldbus submodule inserted.
1	RS485 module	RS485 for Modbus (master or slave) or ComUserTask.
2	PROFIBUS master	PROFIBUS DP master.
3	PROFIBUS slave	PROFIBUS DP slave.
5	RS232 module	RS232 for use with ComUserTask.
6	RS422 module	RS422 for use with ComUserTask.
7	SSI module	SSI for use with ComUserTask.
8	CAN module	CAN for use with ComUserTask. Only available for HIMatrix.

Table 15: Options for Fieldbus Interfaces FB1 and FB2

3.4.2.2 HIMax COM Module Part Number

When the X-COM 01 is equipped with one or multiple fieldbus submodules, in addition to the the part number, the module name changes from X-COM 01 to X-COM 010 XY.

Each COM module forms a functional unit with the X-CB 001 02 connector board. Note that the connector board must be separately purchased.

The following table specifies the available components:

Designation	Description
X-COM 01	Communication module without fieldbus submodules.
X-COM 010 **XY** [1)]	Communication module with fieldbus submodule.
X-CB 001 02	Connector board.
[1)] **X**: Option for fieldbus interface FB1 in accordance with Table 15. **Y**: Option for fieldbus interface FB2 in accordance with Table 15.	

Table 16: Available HIMax Components

The designation and part number (part no.) are printed on the type label of the module.

i HIMA recommends operating the PROFIBUS DP using the FB1 fieldbus interface (maximum transfer rate 12 Mbit). The maximum transfer rate permitted for the FB2 fieldbus interface is 1.5 Mbit.

3.4.2.3 HIMatrix Controller Part Numbers

The HIMatrix controllers can be equipped with fieldbus submodules in accordance with the following table:

Controller	FB1 and FB2	FB3
F30 03z **XY**[1)]	Freely equippable in accordance with Table 15.	Integrated RS485
F35 03z **XY**[1)]	Freely equippable in accordance with Table 15.	Integrated RS485
F60 CPU 03z **XY**[1)]	Freely equippable in accordance with Table 15.	---
[1)] **X**: Option for fieldbus interface FB1 in accordance with Table 15. **Y**: Option for fieldbus interface FB2 in accordance [1)]Table 15. z: Hardware variant.		

Table 17: Equipment of HIMatrix Controllers with Fieldbus Submodules

The part number changes when the appropriate fieldbus submodule is selected:

Example: The part number for F35 030 XY is 98 22**XY**497

X: Option for fieldbus interface FB1 according to Table 15.
Y: Option for fieldbus interface FB2 according to Table 15.

3.4.3 HIMax and HIMatrix Fieldbus Interfaces

Pin assignment of the HIMax and HIMatrix fieldbus interfaces depends on the selected communication option, see Chapter 3.4.

i

Wiring and bus termination!

Observe the corresponding fieldbus standard when connecting the fieldbus interfaces.

- These require a suitable grounding concept.
- The shielded cables should be connected on both sides over a large area. Use the bus terminations to terminate the fieldbuses on their physical ends.

3.4.3.1 RS485 for Modbus Master, Slave or ComUserTask

One RS485 cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RxD/TxD-A	Receive/send data A.
4	CNTR-A	Control signal A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RxD/TxD-B	Receive/send data B.
9	CNTR-B	Control signal B.

Table 18: Pin Assignment of D-Sub Connectors for RS485

3.4.3.2 PROFIBUS DP Master or Slave

One PROFIBUS DP cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	-	Not used.
2	-	Not used.
3	RxD/TxD-A	PROFIBUS DP receive/send data A.
4	RTS	Control signal.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RxD/TxD-B	PROFIBUS DP receive/send data B.
9	-	Not used.

Table 19: Pin Assignment of D-Sub Connectors for PROFIBUS DP

3.4.3.3 RS232 für ComUserTask

One RS485 (RS232) cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	-	Not used.
2	TxD	Send data.
3	RxD	Receive data.
4	-	Not used.
5	DGND	Data transmission potential (ground to 5 V).
6	-	Not used.
7	RTS	Request to send.
8	-	Not used.
9	-	Not used.

Table 20: Pin Assignment of D-Sub Connectors for RS232

3.4.3.4 RS422 for ComUserTask

One RS485 (RS422) cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RxA	Receive data A.
4	TxA	Send data A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RxB	Receive data B.
9	TxB	Send data B.

Table 21: Pin Assignment of D-Sub Connectors for RS422

3.4.3.5 SSI

One RS485 (SSI) cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	D2+	Data input channel 2+.
2	D1-	Data input channel 1-.
3	CL2+/D3+	Clock output channel 2+ or data input channel 3+.
4	CL1+	Clock output channel 1+.
5	GND	Reference potential.
6	D1+	Data input channel 1+.
7	D2-	Data input channel 2-.
8	CL2-/D3-	Clock output channel 2- or data input channel 3-.
9	CL1-	Clock output channel 1-.

Table 22: Pin Assignment of D-Sub Connectors for SSI

3.4.3.6 CAN

One CAN cable must be used, see Chapter 3.7.

Connection	Signal	Function
1	-	Not used.
2	CAN-L	CAN-Low.
3	GND	Reference potential.
4	-	Not used.
5	-	Not used.
6	-	Not used.
7	CAN-H	CAN-High.
8	-	Not used.
9	-	Not used.

Table 23: Pin Assignment of D-Sub Connectors for CAN

3.4.4 HIQuad X F-COM 01 Fieldbus Interfaces

Pin assignment of the F-COM 01 fieldbus interfaces FB1/FB2 depends on the selected communication option, see Chapter 3.4.

i

Wiring and bus termination!
Observe the corresponding fieldbus standard when connecting the fieldbus interfaces.

- These require a suitable grounding concept.
- The shielded cables should be connected on both sides over a large area. Use the bus terminations to terminate the fieldbuses on their physical ends.

3.4.4.1 RS422

One RS485 (RS422) cable must be used, see Chapter 3.7.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RxD-A	Receive data A.
4	TxD-A	Send data A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RxD-B	Receive data B.
9	TxD-B	Send data B.

Table 24: Pin Assignment of the FB1 Interface with RS422

3.4.4.2 RS485 with RTS

One RS485 cable must be used, see Chapter 3.7.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RXD/TXD-A	Receive/send data A.
4	CNTR-A	Control signal A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RXD/TXD-B	Receive/send data B.
9	CNTR-B	Control signal B.

Table 25: Pin Assignment of the FB1 Interface with RS485 (with RTS)

3.4.4.3 Twice RS485 (without RTS)

One (two) RS485 cable must be used, see Chapter 3.7.

The pin assignment does not comply with the standard, since two interfaces are connected to one plug.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RxD1/TxD1-A	First receive/send data A.
4	RxD2/TxD2-A	Second receive/send data A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RxD1/TxD1-B	First receive/send data B.
9	RxD2/TxD2-B	Second receive/send data B.

Table 26: Pin Assignment of the FB1 and FB2 Interface with two RS485 (without RTS)

i The assignment Table 25 is active upon reload completion on FB1 with RS485 (with RTS).

The assignment Table 27 is active upon reload completion on FB2 with RS485 (without RTS).

3.4.4.4 FB2 with RS485 (without RTS)

One RS485 cable must be used, see Chapter 3.7.

The pin assignment does not comply with the standard.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	-	-
4	RxD2/TxD2-A	Second receive/send data A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	-	-
9	RxD2/TxD2-B	Second receive/send data B.

Table 27: Pin Assignment of the FB2 Interface with RS485 (without RTS)

3.4.4.5 PROFIBUS DP Slave

One PROFIBUS DP cable must be used, see Chapter 3.7.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	RXD/TXD-A	PROFIBUS DP receive/send data A.
4	CNTR-A	Control signal A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	RXD/TXD-B	PROFIBUS DP receive/send data B.
9	CNTR-B	Control signal B.

Table 28: Pin Assignment of the FB1 Interface with PROFIBUS DP Slave

3.4.4.6 PROFIBUS DP Slave and RS485

The pin assignment does not comply with the standard, since two interfaces are connected to one plug.

One PROFIBUS DP cable must be used for PROFIBUS DP slaves. One RS485 cable must be used for RS485, see Chapter 3.7.

Pin	Signal	Description
1	-	Not used.
2	5 V	Fieldbus supply decoupled via diode.
3	PROFIBUS DP RXD/TXD-A	PROFIBUS DP receive/send data A.
4	RS485 RxD1/TxD1-A	Receive/send data A.
5	DGND	Data transmission potential (ground to 5 V).
6	5 V	Fieldbus supply.
7	-	Not used.
8	PROFIBUS DP RXD/TXD-B	PROFIBUS DP receive/send data B.
9	RS485 RxD1/TxD1-B	RS485 receive/send data B.

Table 29: Pin Assignment of the FB1/2 Interface with PROFIBUS DP Slave and RS485

3.5 Technical Characteristics of RS485 Transmission

The following table presents the basic technical features of the RS485 transmission that is used for the PROFIBUS DP.

Element	Description
Network topology	Linear bus, active bus termination on both ends.
Medium	Shielded, twisted pair wires
Connectors	9-pin D-sub connector, see Chapter 3.4.3 and Chapter 3.4.4.
Number of bus subscribers for each segment	32 subscribers in every segment, without repeaters[1].
Total number of bus subscribers for each bus	1 Modbus master, 3 repeaters[1]. 121 Modbus slaves.
Max. length of a bus segment	1200 m for each segment.
Max. length of the bus	4800 m, 4 segments with 3 repeaters[1].
Max. baud rate	115200 Bit/s
[1] The maximum number of bus subscribers in the segment decreases by 1 for each repeater used. This means that a maximum of 31 subscribers may be operated on the segment. According to the standard, a total of three repeaters may be used so that a maximum of 121 Modbus slaves may be connected per serial interface on a Modbus master. If several interfaces are available (HIMax and HIMatrix), up to 3 interface slaves or repeaters can be connected. Internally, the system behaves like a master. The minimum number of slaves is thus 254.	

Table 30: Properties of the RS485 Transmission

The cable length specified in Table 31 depends on the baud rate selected.

Baud rate	Cable length for each segment	RS485	PROFIBUS DP
300 Bit/s	1200 m	X	-
600 Bit/s	1200 m	X	-
1200 Bit/s	1200 m	X	-
2400 Bit/s	1200 m	X	-
4800 Bit/s	1200 m	X	-
9600 Bit/s	1200 m	X	X
19200 Bit/s	1200 m	X	X
38400 Bit/s	1200 m	X	-
45450 Bit/s	1200 m	-	X
57600 Bit/s	1200 m	X	-
62500 Bit/s	1200 m	X	-
76800 Bit/s	1200 m	X	-
93750 Bit/s	1200 m	-	X
115200 Bit/s	1200 m	X	-
187500 Bit/s	1000 m	-	X
500000 Bit/s	400 m	-	X
1.5 MBit/s	200 m	-	X
3 MBit/s	100 m	-	X
6 MBit/s	100 m	-	X
12 MBit/s	100 m	-	X

Table 31: Cable Length According to the Baud Rate for RS485 and PROFIBUS DP

The cable length may be increased by using bidirectional repeaters. A maximum of three repeaters may be connected between two subscribers. In doing so, a cable length of 4.8 km may be achieved.

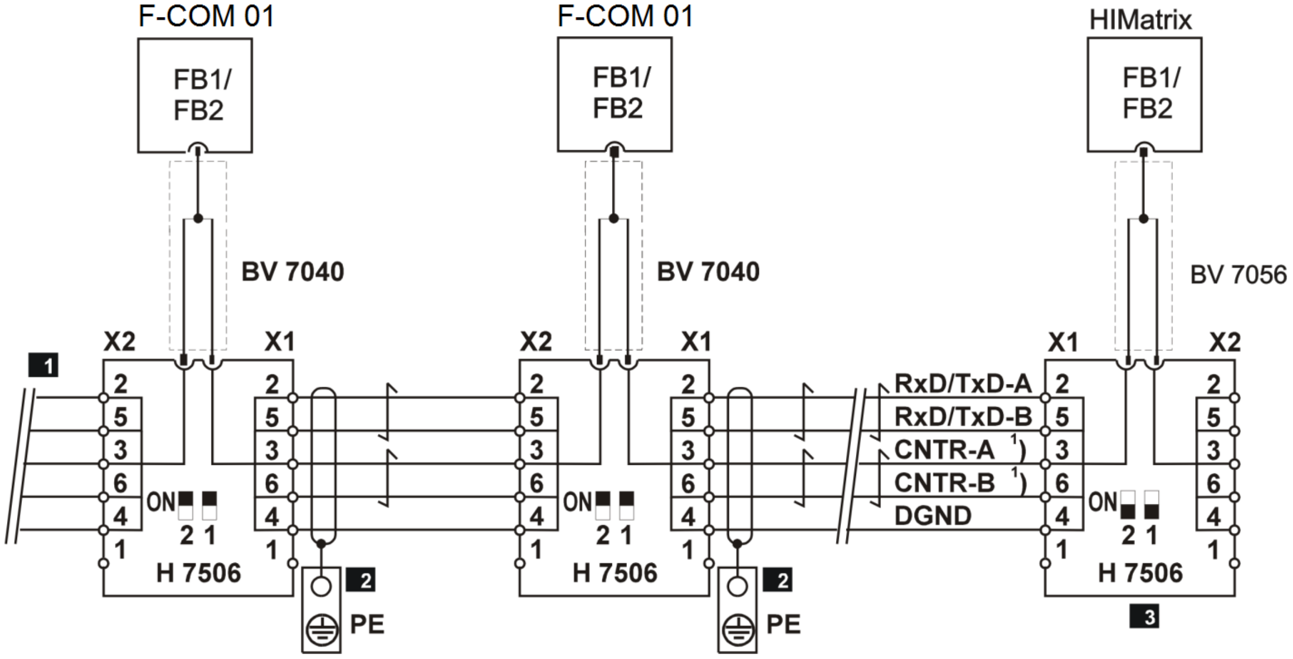
For time-critical applications, HIMA recommends connecting no more than 32 bus subscribers. For non-time-critical applications, up to 126 subscribers (with repeaters) may be used.

3.6 RS485 Bus Topology

The following picture shows a structure example of RS485 bus topology using HIMA components. H 7506 are used as bus terminals. The total bus length may not exceed 1200 m. A repeater such as the H 7505) must be used for long distances. A total of 3 repeaters may be used. The bus may thus have a maximum extension of 4800 m.

If fiber optic cable or RS485 converters are used in the bus, H 7505 must not be used (no automatic switching of data direction).

The time until the information from a slave is available on a master increases by the number of slaves on the bus. The more slaves are connected to the bus, the worse the system response time will become.



1) Only necessary in repeater operation mode

1 Additional controllers

2 Protective conductor terminal, USLKG4 YE/GN

3 H 7506 switch position (bus termination) (switch position: both white switches set to ON)

Figure 2: RS485 Bus Topology

Equipotential bonding should be used if the bus is extended over larger distances.

At transmission rates ≥ 1.5 MBit/s, branch lines must be strictly avoided. For this reason, use suitable bus connector plugs only.

3.6.1 H 7506 Terminal Assignment

The following table shows the terminal assignment of the HIMA H 7506 bus terminal. The HIMA BV 7040 cable connects the H 7506 to the FBx fieldbus interface of the controller.

X1/X2	Color	Description
1	-	-
2	WH	RxD/TxD-A, data cable.
3	GN	CNTR-A, control line for repeater.
4	GY	DGND
5	BN	RxD/TxD-B, data cable.
6	YE	CNTR-B, control line for repeaters.

Table 32: Terminal Assignment for H 7506

i For registered customers, the product documentation for this and other HIMA RS485 components is available at https://www.hima.com/en/downloads/.

3.6.2 Bus Connection and Bus Termination

The incoming and outgoing data cables can be directly connected in the bus connector plug. This avoids branch lines and the bus connector plug can be plugged in to and out from the field device at any time without interrupting the data traffic.

The IEC 61158 standard recommends using a 9-pole D-sub connector. Depending on the degree of protection of the field device, other slots, which are not occupied, may be used.

Figure 3 shows the pin assignment of the 9-pole D-sub connector. The bus connection to the field device is implemented as a socket.

The PROFIBUS DP bus connection includes a resistor combination that ensures a defined rest potential on the bus cable. The resistor combination is integrated in the PROFIBUS DP bus connector plugs and can be activated via bridges or switches.

Additionally, stations at which the bus terminates should provide 5 V at pin 6.

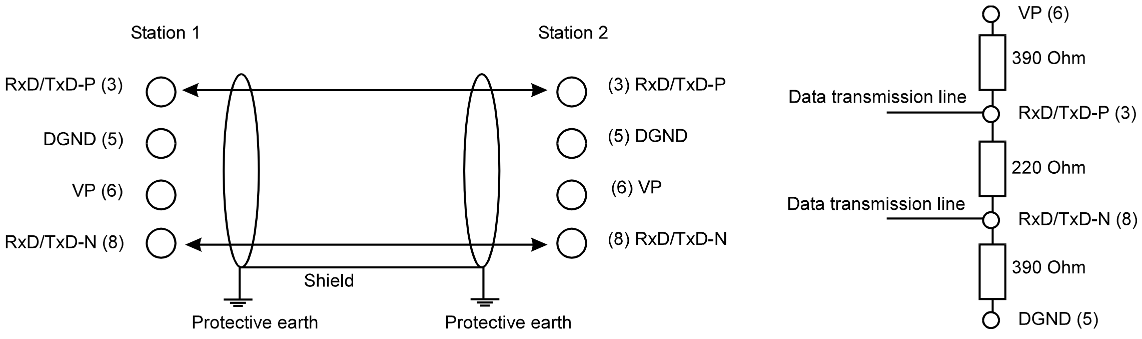


Figure 3: Bus Connection and Bus Termination, Pin Assignment of the Fieldbus Interface

3.7 Communication Cable Requirements

For communication connections running within a control cabinet, the minimum cable cross-section must be 0.2 mm².

For communication connections running outside a control cabinet, the minimum cable cross-section must be 0.5 mm². If necessary, installation cables with rigid cores must be used instead of cables with flexible cores.

Cables with the following characteristics are permitted for connecting to Ethernet or fieldbus interfaces:

- All the cables for Ethernet or fieldbus interfaces must withstand at least 500 bending cycles, if bending load is applied during the intended operating conditions.
- All the cables for Ethernet or fieldbus interfaces must withstand at least 25 bending cycles, if bending load is only applied during maintenance.
- All the cables for Ethernet or fieldbus interfaces must comply with UL94-V0.

3.7.1 Patch Cables

HIMA recommends using patch cables with the following minimum requirements: Cat.5e, RJ-45.

3.7.2 CAN Cables

As transmission medium, HIMA recommends only using CAN cables approved for CAN.

3.7.3 RS485 (RS422, RS232, SSI) Cables

As bus cables for RS485 (RS422, RS232 and SSI), HIMA recommends using shielded twisted pair wires with the following characteristics:

Element	Description
Cable type	LiYCY 3 x 2 x 0.25 mm^2 for RS485, RS422, RS232. LiYCY 6 x 2 x 0.25 mm^2 for SSI
Wire cross-section	> 0.25 mm^2
Impedance	100…120 Ω

Table 33: RS485 (RS422, RS232, SSI) Bus Cables

3.7.4 PROFINET Cables

As transmission medium, HIMA recommends only using PROFINET cables approved for PROFINET.

3.7.5 PROFIBUS DP Cables

As transmission medium, HIMA recommends only using PROFIBUS DP cables approved for PROFIBUS DP with the following parameters:

Parameters	Cable type A
Impedance	135…165 Ω
Capacitance	≤ 30 pF / m
Loop impendence	≤ 110 Ω / km
Wire diameter	> 0.64 mm
Wire cross-section	> 0.34 mm^2

Table 34: Parameters of the PROFIBUS DP Cable Type A

Cable type A can be used for all transfer rates up to 12 Mbit/s.

4 safeethernet

All HIMA can safely communicate via safeethernet.

i The safeethernet protocol meets all requirements for safety-related protocols in accordance with IEC 61508-2:2010, IEC 61784-3:2019 and EN 50159:2010. The TÜV has tested these features and verified the safe**ethernet** protocol as part of HIMA systems.

If the bit error probability of the transmission medium is 0.5, e.g., due to a disturbed network, the residual error rate λ_{SCL} of a safety-related function with 100 safe**ethernet** connections is less than 1 % of SIL 4 in accordance with IEC 61508-2:2010, IEC 61784-3:2019 and EN 50159:2010.
The residual error rate λ_{SCL} is applicable irrespective of the number of storing network elements, non-safety-related devices, the use of WLAN, compression and encryption.
This results in a residual error rate λ_{SCL} of less than 10^{-12}/h for the individual safe**ethernet** connection.

The corresponding Ethernet interfaces of the HIMA controllers can be also used for other protocols.

Various Ethernet network topologies can be used to ensure safe**ethernet** communication between controllers. To this end, so-called safe**ethernet** profiles suitable for the Ethernet network in use can be selected in SILworX to increase the data transmission speed and efficiency.
These safe**ethernet** profiles ensure safe**ethernet** communication, without requiring users to first become familiar with all the details involved in network configuration.

⚠ Warning

Manipulation of safety-related data transmission!

Physical injury

The plant manufacturer and the operator are responsible for ensuring that the Ethernet network used for safeethernet is sufficiently protected against manipulations (e.g., from hackers).
The type and extent of the measures must be agreed upon together with the responsible test authority.

4.1 General Information about safeethernet

Requirements as determinism, reliability, interchangeability, expandability and above all safety, are central issues within the process and automation technology.

safe**ethernet** is a protocol for transmitting safety-related data up to SIL 4 in accordance with IEC 61508-2:2010, IEC 61784-3:2019 and EN 50159:2010 when Ethernet technology is used.

safe**ethernet** implements mechanisms that can detect and safely respond to faults.

The transmission of safety-related data occurs via standard Ethernet (IEEE 802.3) and is based on UDP/IP.

safe**ethernet** uses "unsafe data transmission channels" (Ethernet) in accordance with the black channel approach and monitors the data correctness through safety-related protocol mechanism. This allows users to use normal Ethernet network components such as switches, routers and wireless LAN devices within a safety-related network.

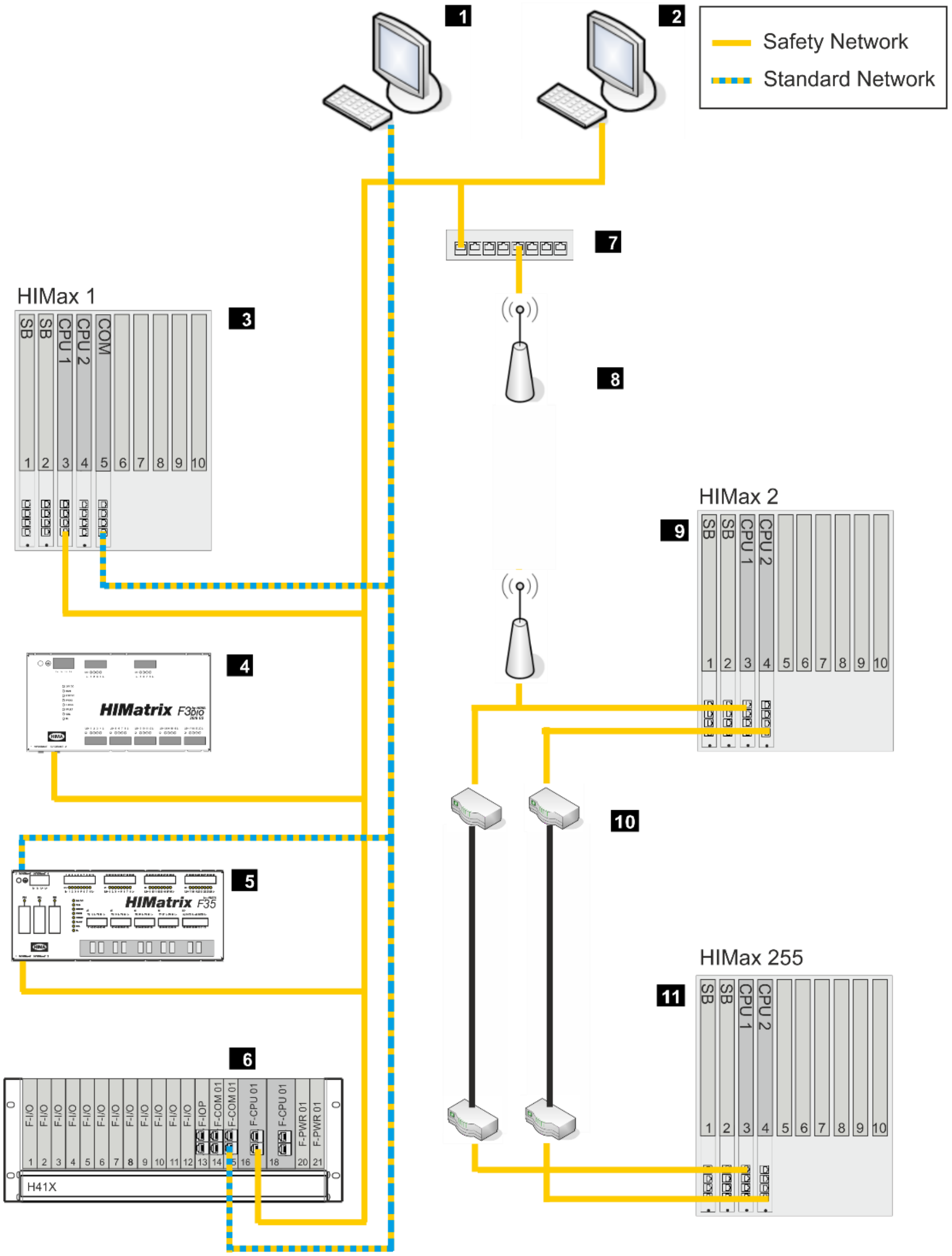
safe**ethernet** uses the abilities of standard Ethernet so that security and real-time capability are made possible. A special protocol mechanism ensures a deterministic behavior even if faults occur or new communication subscribers join the network. The system automatically integrates new components in the running system. All network components can be replaced during operation. Transmission times can be clearly defined using switches. If properly configured, Ethernet is thus real-time capable.

The possible transfer rate of up to 1 Gbit/s offers automation applications sufficient transmission capacity for safety-related data. Transmission media such as copper lines and fiber optic cables can be used.

safe**ethernet** data can be transmitted via the existing company-internal Ethernet network in addition to other data traffic on the Ethernet network. However, this could increase security risks.

i To reduce security risks, HIMA recommends setting up a safety network via the CPU modules and a separate standard network via the COM modules. The standard network is used to connect to non-safety components such as X-OPC Server, see Figure 4.

safe**ethernet** allows flexible system structures to be created with defined response times for decentralized automation. Depending on the requirements, the intelligence can be distributed to the network subscribers in a centralized or decentralized manner.



1 PC (X-OPC Server)

2 PC (PADT SILworX)

3 HIMax (address 1)

4 Remote I/O

5 Separating HIMatrix switch ports via VLAN

6 HIQuad X H41 X

7 Switch/router

8 Radio, satellite, WLAN

9 HIMax (address 2)

10 Fiber optic cable

11 HIMax (address 255)

Figure 4: Flexible System Structure with safeethernet

i A faulty network structure can cause a part of or the entire HIMA system to shut down.

The generally accepted regulations for developing Ethernet networks must be observed. no network loop may occur Data packets may only reach a controller over a single path, see also Chapter 4.7.

4.2 User Requirements for safeethernet in a Noisy Network

The plant manufacturer and the operator must include the effects of the noisy network on the application in their safety analysis.

To ensure that safe**ethernet** achieves sufficient availability for the respective application, users must comply with the following requirements.

- Users must select a suitable transmission system for the safety-related process data communication and set the safe**ethernet** parameters in such a way that sufficient availability is achieved for the application. In their safety analysis, users must consider the dangers of an unintentional shutdown by safe**ethernet** , for example. In case of doubt, the degree of availability required has to be agreed with the responsible test authority.
- Users must ensure that their communication system adheres to the configured response time and that this is less than or equal to half the value of the receive timeout. If not, the worst case response time must be suitable for the safety-related function even if the double the value of the receive timeout is included in the worst case response time calculation.
- If users cannot always ensure that their communication system meets the configured response time, they must monitor this response time and the response time measured by the system (system variable of the connection). Only in rare exceptional cases may the measured response time be exceeded by more than half the value of the receive timeout. Alternatively, users can also include double the value of the receive timeout in the worst case response time calculation of the safety-related function.
- If users operate a safe**ethernet** connection in a noisy network, or if the configured response time is not or frequently not observed and/or a cleanroom profile is used, HIMA does not recommend using the cleanroom profile due to potentially reduced availability!
  If safe**ethernet** needs to be used under these conditions, the *Receive Timeout* must be set so that the worst case response time for the safety-related function is still suitable if double the value of the *Receive Timeout* were to be used for calculating the worst case response time.
  The factor *n* in *Response Time ≤ Receive Timeout / n*, where *n* > 4, can for instance be configured to increase the availability of the safe**ethernet** connection. The value of n depends on the availability actually required or necessary. The characteristics of the transmission system must be considered.

4.3 HIMA System Quantity Structure for safeethernet

HIMA systems HIMax and HIQuad X support the safeethernet protocol with the following properties.

Element	HIMax	HIQuad X	Description
System view			The pictures are examples of the respective system family. The figures depict a HIMax and a HIQuad X H51X.
Module/controller	For each HIMax 1…4 X-CPU 01 1…2 X-CPU 31	For each H41X/H51X: 1…2 F-CPU 01	safeethernet is run on the safety-related CPU module.
Ethernet interfaces	X-CPU 01: 1 GBit/s X-CPU 31:100 Mbit/s X-COM 01: 100 Mbit/s	F-CPU 01: 100 Mbit/s F-COM 01: 100 Mbit/s	The Ethernet interfaces in use can simultaneously be used for additional protocols.
Connections	255	128	safeethernet connections to other controllers and remote I/Os.
Connections between two controllers	1 prior to CPU OS V6 64 as of CPU OS V6	64	safeethernet connections
Redundant connections	255	128	2-channel operation Redundant safeethernet connections between HIMA controllers can be configured in the safeethernet Editor.
Process data volume for each connection	1100 bytes	1100 bytes	For each safeethernet connection.
n.a.: not applicable			

Table 35: safeethernet Protocol for HIMax und HIQuad X

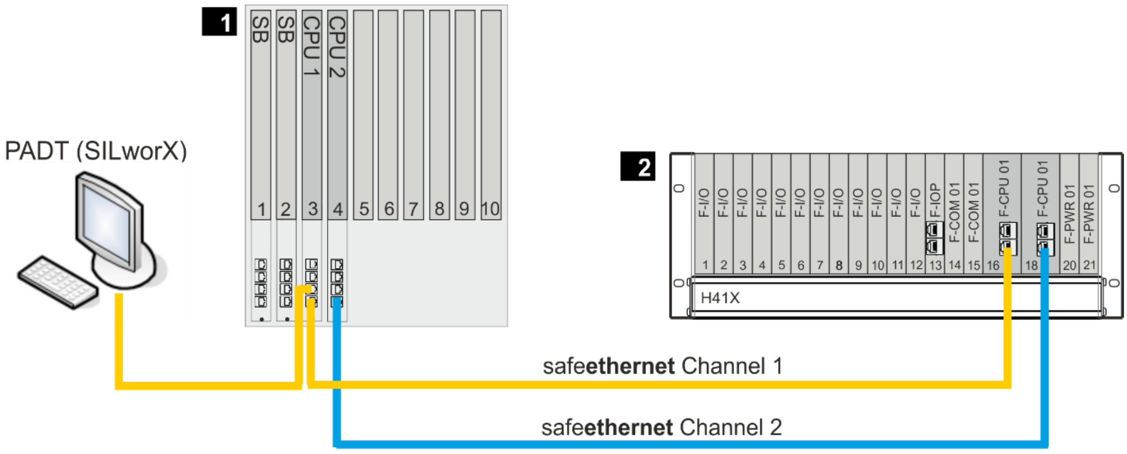
HIMA's HIMatrix systems support the safeethernet protocol with the following properties.

Element	HIMatrix	Description
System view		The picture is an example of the respective system family. It shows an F30.
Module/controller	Integrated CPU module of the controller	safe**ethernet** is run on the safety-related CPU module.
Ethernet interfaces	100 Mbit/s	The Ethernet interfaces in use can simultaneously be used for additional protocols.
Connections	128 prior to CPU OS V12 255 as of CPU OS V12	safe**ethernet** connections to other controllers and remote I/Os.
Connections between two controllers	1 prior to CPU OS V10 64 as of CPU OS V10	safe**ethernet** connections
Redundant connections	128 prior to CPU OS V12 255 as of CPU OS V12	2-channel operation. Redundant safe**ethernet** connections between HIMA controllers can be configured in the safe**ethernet** Editor.
Process data volume for each connection	1100 bytes	For each safe**ethernet** connection.
n.a.: not applicable		

Table 36: safeethernet Protocol for HIMatrix

4.4 Configuring a Redundant safeethernet Connection

This example shows how to configure a redundant safe**ethernet** connection between two HIMA controllers.



1 Resource A **2** Resource B

Figure 5: Structure for Configuring a Redundant Connection

i For redundant safe**ethernet** connections, HIMA recommends implementing the two transport paths (channel 1 and channel 2) via two Ethernet networks that are completely separated from one another. In doing so, the bandwidth and the delay on the respective transport paths must be nearly identical.

4.4.1 Establishing the safe**ethernet** Connection

In the safe**ethernet** Editor, create a safeethernet connection between Resource A and Resource B.

To open the safeethernet Editor of resource A

1. In the structure tree, open **Configuration**, **Resource**.
2. Right-click safe**ethernet** and select **Edit** from the context menu.

   ☑ *Resource B* is located in the Object Panel.

To create the safeethernet connection to resource B

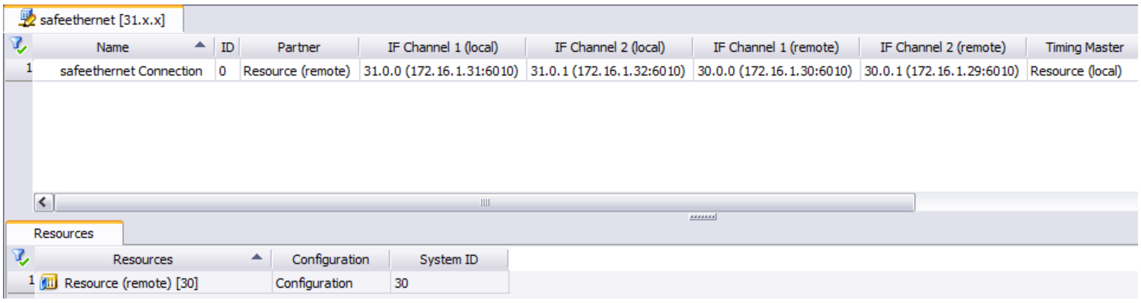
1. In the Object Panel, click **Resource B** and drag it onto a free space within the workspace of the safe**ethernet** Editor.

   ☑ A dialog box appears to enter a name for the safe**ethernet** connection. This name must be unique.

i The reciprocal communication path is automatically added to resource B in the safe**ethernet** Editor.

To configure the safeethernet connection

1. Select **Ethernet Interfaces Channel 1** for resource A and resource B.
2. Select **Ethernet Interfaces Channel 2** for resource A and resource B.
3. Select the **Network Profile** (e.g., Fast&Noisy) for the safeethernet connection.
4. Calculate and enter the **Receive Timeout** and **Response Time** (see Chapter 4.8).



safeethernet [31.x.x]

	Name	ID	Partner	IF Channel 1 (local)	IF Channel 2 (local)	IF Channel 1 (remote)	IF Channel 2 (remote)	Timing Master
1	safeethernet Connection	0	Resource (remote)	31.0.0 (172.16.1.31:6010)	31.0.1 (172.16.1.32:6010)	30.0.0 (172.16.1.30:6010)	30.0.1 (172.16.1.29:6010)	Resource (local)

Resources

	Resources	Configuration	System ID
1	Resource (remote) [30]	Configuration	30

Figure 6: View in the safe**ethernet** Editor

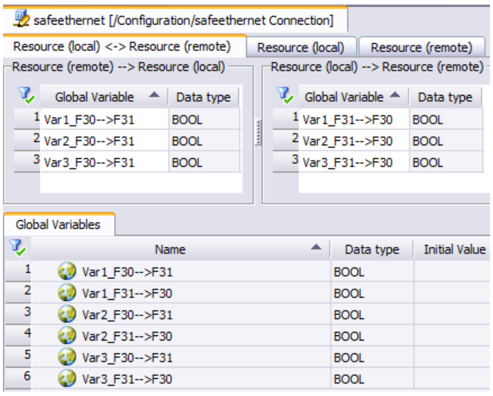
4.4.2 Configuring within the safe**ethernet** Connection Editor

To connect process variables in the safe**ethernet** connection editor.

i Only global variables from the configuration context may be used, and not from the resource or project context!

To open the connection editor

1. Right-click the created safe**ethernet** connection and open the context menu.
2. Select **Edit** from the context menu to open the connection editor of the safe**ethernet** connection.
3. Select the **Resource A<->Resource B** tab.
4. In the Object Panel, select a Global Variable and drag it onto the **Resource A --> Resource B** area or onto the **Resource B --> Resource A** area depending on the selected transport direction.
5. Repeat this step for additional global variables.



safeethernet [/Configuration/safeethernet Connection]

Resource (local) <-> Resource (remote) | Resource (local) | Resource (remote)

Resource (remote) --> Resource (local)

	Global Variable	Data type
1	Var1_F30-->F31	BOOL
2	Var2_F30-->F31	BOOL
3	Var3_F30-->F31	BOOL

Resource (local) --> Resource (remote)

	Global Variable	Data type
1	Var1_F31-->F30	BOOL
2	Var2_F31-->F30	BOOL
3	Var3_F31-->F30	BOOL

Global Variables

	Name	Data type	Initial Value
1	Var1_F30-->F31	BOOL	
2	Var1_F31-->F30	BOOL	
3	Var2_F30-->F31	BOOL	
4	Var2_F31-->F30	BOOL	
5	Var3_F30-->F31	BOOL	
6	Var3_F31-->F30	BOOL	

Figure 7: View in the safe**ethernet** Connection Editor

i Evaluate the system variables of the safe**ethernet** connection in the user program!
In the respective subtabs of Resource A and Resource B, global variables should at least be assigned to the system variables Connection State, Quality of Channel 1 and Quality of Channel 2 to evaluate these in the user program.

To verify the safeethernet connection

1. In the structure tree, select **Configuration, Resource**, safe**ethernet** .
2. Right-click and select **Verification** from the context menu.
3. Thoroughly verify the messages displayed in the logbook and correct potential errors.

i The configuration of the safe**ethernet** connection must be compiled with the user program of resource A and resource B and transferred to the controllers. The new configuration can only be used for communicating with the HIMA controller upon completion of this step.

4.4.3 Verifying safeethernet Communication

Reset the views in the Control Panel to zero with **Reset safeethernet Statistics**.

To verify that the redundant safe**ethernet** connection was established properly, disconnect and reconnect one redundant connection and then repeat this test for the other connection. During this test, there must be no faults in the safe**ethernet** communication. Also observe the values for Quality Channel 1 and Quality Channel 2 here.

i Additional causes for *Bad Messages and Resends*!

Verify the correct network design (e.g., wires, switches, PCs).
If the Ethernet network is not exclusively used for safe**ethernet** , also verify the network load (probable data collisions).

4.5 safeethernet Connection Overview

The safe**ethernet** connection overview for a resource lists all configured safe**ethernet** connections. The overview can also be used to create new safe**ethernet** connections.

To open the safe**ethernet** connection overview

1. In the structure tree, open **Configuration**, **Resource**.
2. Right-click safeethernet and select **Edit** from the context menu.

The safe**ethernet** connection overview displays the following safe**ethernet** protocol parameters:

Parameters	Description
Name	Name of the safe**ethernet** connection
ID	safe**ethernet** connection ID. Range of values: 0…63
Partner	Resource name of the link partner
IF Channel	Ethernet interfaces available on the (local) and (remote) resource, see also Chapter 3.3.
Timing Master	The timing master provides the value for *Receive Timeout*, *Resend Timeout* and the *Acknowledge Timeout* for this safe**ethernet** connection. The opposite controller is the timing slave, which adopts these values. If no timing master is selected, the controller with the smaller IP address determines these safe**ethernet** parameters.
Profile	Combination of matching safe**ethernet** parameters, see also Chapter 4.10.
Rsp t	*Response Time* is the time period expressed in milliseconds (ms) until the sender of the message receives acknowledgement from the recipient, see also Chapter 4.8.3. Default value: 500 ms
Receive Timeout	Monitoring time of controller 1 within which a valid response from controller 2 must be received, see also Chapter 4.8.2. Default value: 1000 ms

<table>
<tr><th>Parameters</th><th>Description</th></tr>
<tr><td>Resend Timeout</td><td>Monitoring time expressed in milliseconds (ms) and set in controller 1 within which controller 2 must have acknowledged the receipt of a data packet; upon expiration of this period, the data packet is sent again, see also Chapter 4.8.5.</td></tr>
<tr><td>Acknowledgment Timeout</td><td>Time period expressed in milliseconds (ms) within which the CPU must acknowledge the receipt of a data packet, see also Chapter 4.8.6.</td></tr>
<tr><td>Prod Rate</td><td>The production rate is the minimum time interval between two data packets, see also Chapter 4.8.7.</td></tr>
<tr><td>Memory</td><td>Number of data packets that can be sent without acknowledgment, see also Chapter 4.8.8.</td></tr>
<tr><td>Behavior</td><td>Behavior of the input variables for this safe<b>ethernet</b> connection if the connection is interrupted.
<table>
<tr><td>Use Initial Value</td><td>The initial data are used for the import variables.</td></tr>
<tr><td>Freeze Process Value Indefinitely</td><td>The import variables are frozen to the current value and used until a new connection is established.</td></tr>
<tr><td>Initial Value after [ms]</td><td>Input: Double-click the field and enter the time value in milliseconds.
The import variables are frozen to the current value and used until the configured timeout. Afterwards, the initial data are used for the input variables.
The timeout can be extended by up to a CPU cycle.</td></tr>
</table>
⚠ Caution
<b>For safety-related functions implemented via safeethernet</b>, <b><i>Use Initial Value</i> is the only setting which may be used.</b></td></tr>
<tr><td>Diag.Entry</td><td>The number of warnings that must occur in sequence within the <i>Warning Period [ms]</i> before the warnings are recorded in the diagnostics or communication fault statistic.</td></tr>
<tr><td>Prio A&E</td><td>The function is only activated for the connection to the X-OPC Server.
This defines the priority for events requested by the X-OPC Server from the controller.
Fragments with priority <b>n</b> and fragments with priority <b>m</b> are sent at a ratio of <b>n</b> to <b>m</b> times.</td></tr>
<tr><td>Prio Sync</td><td>The function is only activated for the connection to the X-OPC Server.
This defines the priority for state values requested by the X-OPC Server from the controller.
Fragments with priority <b>n</b> and fragments with priority <b>m</b> are sent at a ratio of <b>n</b> to <b>m</b> times.</td></tr>
<tr><td>Activate A&E</td><td>The function can only be used and changed for connections to the X-OPC Server.</td></tr>
<tr><td>Codegen</td><td>To operate safe<b>ethernet</b> connections with communication partners with SILworX versions prior to V6, set the code generation version to <b>Prior to V6</b>.
V6 and higher: Reload of safe<b>ethernet</b> connection is possible.
Prior to V6: Reload of safe<b>ethernet</b> connection is not possible.
Default value: <b>V6 and higher</b></td></tr>
</table>

Table 37: safe**ethernet** Protocol Parameters

Object Panel

The Object Panel contains all the project resources to which the current resource can be connected via safe**ethernet**.

i The archive function can be used for safe**ethernet** connections to resources outside the project (see Chapter 4.13).

4.6 Connection Editor of a safeethernet Connection

The safe**ethernet** Editor always refers to the local resource from which the safe**ethernet** Editor was started.

To open the safeethernet connection overview

1. In the structure tree, open **Configuration**, **Resource**.
2. Right-click safe**ethernet** and select **Edit** from the context menu.

To open the Connection Editor of a safeethernet Connection

1. Right-click the required safe**ethernet** connection to open the context menu.
2. Select **Edit**.
   ☑ The safe**ethernet** Editor includes the three tabs *Peer1<->Peer2, Peer1 and Peer2*.

4.6.1 The *Resource A<->Resource B* Tab

The *Resource A<->Resource B* tab is divided into two areas for the required transport direction: *Resource B-->Resource A* and *Resource A-->Resource B*.

Global Variables can be dragged onto these two areas from the Object Panel.

4.6.2 The *Resource B* Tab

The *Resource A* tab contains the tabs *System Variables* and *Fragment*
Definitions: *Resource B-->Resource A*, see Chapter 4.6.3.1 and Chapter 4.6.3.2.

4.6.3 The *Resource B* Tab

The *Resource B* tab contains the tabs *System Variables* and *Fragment*
Definitions: *Resource A-->Resource B*, see Chapter 4.6.3.1 and Chapter 4.6.3.2.

4.6.3.1 The System Variables Tab

The safeethernet connection can be controlled and evaluated by means of system variables.

<table>
<tr><th>Name</th><th>Data type</th><th>R/W</th><th>Description</th></tr>
<tr><td colspan="4">The following statuses and parameters can be assigned global variables and used in the user program.</td></tr>
<tr><td>Ack.Frame No.</td><td>UDINT</td><td>R</td><td>Receive counter (revolving).</td></tr>
<tr><td>Number of Faulty Messages</td><td>UDINT</td><td>R</td><td rowspan="2">Number of all faulty messages per channel (invalid CRC, invalid header, other faults).</td></tr>
<tr><td>Number of Faulty Messages for Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Number of Successful Connections</td><td>UDINT</td><td>R</td><td>Number of successful connections since statistics reset.</td></tr>
<tr><td>Number of Lost Messages</td><td>UDINT</td><td>R</td><td rowspan="2">Number of messages dropped out on one of the two transport paths since statistics reset.
The counter only continues to run until a channel completely fails.</td></tr>
<tr><td>Number of Lost Messages for Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Early Queue Usage</td><td>UDINT</td><td>R</td><td>Number of early messages since statistics reset. Early messages are stored in the <i>Early Queue</i>. See also Chapter 4.8.8.</td></tr>
<tr><td>Bad Messages</td><td>UDINT</td><td>R</td><td>Number of rejected messages since statistics reset.</td></tr>
<tr><td>Frame No.</td><td>UDINT</td><td>R</td><td>Send counter (revolving).</td></tr>
<tr><td>Channel State</td><td>USINT</td><td>R</td><td>Current state of Channel 1.
<table>
<tr><th>Status</th><th>Description</th></tr>
<tr><td>0</td><td>No message on the state of channel 1.</td></tr>
<tr><td>1</td><td>Channel 1 OK.</td></tr>
<tr><td>2</td><td>The last message was faulty, the current one is OK.</td></tr>
<tr><td>3</td><td>Error on Channel 1.</td></tr>
</table></td></tr>
<tr><td>Last Channel Latency</td><td>UDINT</td><td>R</td><td rowspan="8">Channel Latency specifies the delay between two redundant transport paths and the reception time of messages with identical SeqNo.
A statistic is kept specifying the average, minimum, maximum and last latency.
If the minimum value is greater that the maximum value, the statistics values are invalid.
The values of Last Channel Latency and <i>Avg. Channel Latency</i> are then 0.</td></tr>
<tr><td>Last Latency of Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Max. Channel Latency</td><td>UDINT</td><td>R</td></tr>
<tr><td>Maximum Latency of Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Min. Channel Latency</td><td>UDINT</td><td>R</td></tr>
<tr><td>Minimum Latency of Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Avg. Channel Latency</td><td>UDINT</td><td>R</td></tr>
<tr><td>Average Latency of Redundant Channel</td><td>UDINT</td><td>R</td></tr>
<tr><td>Monotony</td><td>UDINT</td><td>R</td><td>User data send counter (revolving).</td></tr>
<tr><td>Quality Channel 1</td><td>BYTE</td><td>R</td><td>Quality of the main transport path.
<table>
<tr><th>Bit no.</th><th>Bit = 0</th><th>Bit = 1</th></tr>
<tr><td>0</td><td>Transport path not enabled</td><td>Transport path enabled.</td></tr>
<tr><td>1</td><td>Transport path not used</td><td>Transport path actively used.</td></tr>
<tr><td>2</td><td>Transport path not connected</td><td>Transport path connected.</td></tr>
<tr><td>3</td><td>-</td><td>Transport path first provides message.</td></tr>
<tr><td>4 … 7</td><td>Reserved</td><td>Reserved.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Data type</th><th>R/W</th><th>Description</th></tr>
<tr><td>Quality Channel 2</td><td>BYTE</td><td>R</td><td>Quality of the redundant transport path, see Quality of Channel 1 (main transport path).</td></tr>
<tr><td>Receive Timeout</td><td>UDINT</td><td>R</td><td>Time in milliseconds (ms) on controller1 within which a valid response must be received from controller2, see also Chapter 4.8.2.</td></tr>
<tr><td>Response Time</td><td>UDINT</td><td>R</td><td>Time in milliseconds (ms) until the acknowledgment of the last message is received by the sender.</td></tr>
<tr><td>Reset safeethernet Statistics</td><td>BYTE</td><td>W</td><td>In the user program, reset the statistical values for the communication connection (e.g., number of faulty messages, channel state, timestamp for the last fault on the red. channel [s], resends).
<table>
<tr><th>Value</th><th>Function</th></tr>
<tr><td>0</td><td>No reset.</td></tr>
<tr><td>1 … 255</td><td>Reset the safe<b>ethernet</b> statistics.</td></tr>
</table></td></tr>
<tr><td>Signatur N</td><td>UDINT</td><td>R</td><td>Changing the safe<b>ethernet</b>configuration results in a dual configuration.
Old signature of the safe<b>ethernet</b> configuration.</td></tr>
<tr><td>Signatur N+1</td><td>UDINT</td><td>R</td><td>New signature of the safe<b>ethernet</b>configuration.</td></tr>
<tr><td>Transmission Control for Channel 1</td><td>BYTE</td><td>W</td><td>Transmission control of channel 1.
<table>
<tr><th>Bit 0</th><th>Function</th></tr>
<tr><td>FALSE</td><td>Transport path enabled.</td></tr>
<tr><td>TRUE</td><td>Transport path locked.</td></tr>
</table>
<table>
<tr><th>Bit 1</th><th>Function</th></tr>
<tr><td>FALSE</td><td>Transport path enabled for tests.</td></tr>
<tr><td>TRUE</td><td>Transport path locked.</td></tr>
</table>
Bits 2...7 reserved.</td></tr>
<tr><td>Transmission Control for Channel 2</td><td>BYTE</td><td>W</td><td>Transmission control of channel 2, see Transmission Control for Channel 1.</td></tr>
<tr><td>Connection Control</td><td>WORD</td><td>W</td><td>Use this system variable to control the safeethernet connection from within the user program.
<table>
<tr><th>Command</th><th>Description</th></tr>
<tr><td>AUTOCONNECT (0x0000)</td><td>Default value:
After a safe<b>ethernet</b> communication loss, the controller attempts to re-establish the connection in the following CPU cycle.</td></tr>
<tr><td>Toggle Mode 0(0x0100) Toggle Mode 1(0x0101)</td><td>After a communication loss, the user program can change the toggle mode to re-establish the connection.
▪ TOGGLE MODE 0 (0x100) set: Set to TOGGLE MODE 1 (0x101) to re-establish the connection.
▪ TOGGLE MODE 1 (0x101) set: Set to TOGGLE MODE 0 (0x100) to re-establish the connection.</td></tr>
<tr><td>Disabled (0x8000)</td><td>safe<b>ethernet</b> communication is disabled.</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>Name</th><th>Data type</th><th>R/W</th><th>Description</th></tr>
<tr><td>Connection State</td><td>UINT</td><td>R</td><td>The connection state evaluates the status of the communication between two controllers from within the user program.
<table>
<tr><th>Status/Value</th><th>Description</th></tr>
<tr><td>Closed (0)</td><td>The connection is closed and no attempt is made to open it.</td></tr>
<tr><td>Try_open (1)</td><td>An attempt is made to open the connection, but it is still closed. This state applies for both the active and the passive sides.</td></tr>
<tr><td>Connected (2)</td><td>The connection is established and functioning (active time monitoring and data exchange)</td></tr>
</table></td></tr>
<tr><td>Reload State</td><td>UINT</td><td>R</td><td>Reload state of this safe<b>ethernet</b> connection, see also status of <i>Reload</i> in Chapter 4.11.1.
<table>
<tr><td>unknown</td><td>0x0000</td></tr>
<tr><td>up-to-date</td><td>0x0001</td></tr>
<tr><td>updated</td><td>0x0002</td></tr>
<tr><td>outdated</td><td>0x0003</td></tr>
</table></td></tr>
<tr><td>Resends</td><td>UDINT</td><td>R</td><td>Number of resends since statistics reset [UDINT].</td></tr>
<tr><td>Timestamp of Last Fault on Red. Channel [ms]</td><td>UDINT</td><td>R</td><td>Millisecond fraction of the timestamp (current system time).</td></tr>
<tr><td>Timestamp of Last Fault on Red. Channel [s]</td><td>UDINT</td><td>R</td><td>Second fraction of the timestamp (current system time).</td></tr>
<tr><td>Timestamp of Last Error [ms]</td><td>UDINT</td><td>R</td><td>Millisecond fraction of the timestamp (current system time).</td></tr>
<tr><td>Timestamp of Last Error [ms]</td><td>UDINT</td><td>R</td><td>Second fraction of the timestamp (current system time).</td></tr>
<tr><td>Redundant Channel State</td><td>USINT</td><td>R</td><td>Current state of channel 2.
It is the current state of channel 2 when a message with Seq. no. X is being received (Seq. no X-1).
<table>
<tr><th>Status</th><th>Description</th></tr>
<tr><td>0</td><td>No message on the state of channel 2.</td></tr>
<tr><td>1</td><td>Channel 2 OK.</td></tr>
<tr><td>2</td><td>The last message was faulty, the current one is OK.</td></tr>
<tr><td>3</td><td>Error on channel 2.</td></tr>
</table></td></tr>
</table>

Table 38: System Variables Tab in the safe**ethernet** Editor

4.6.3.2 The *Fragment Definitions* Tab

The *Fragment Definitions* tab contains the statuses and parameters of the fragments sent by the opposite controller.

The refresh rate of the received fragments from all connected controllers required for this controller (or X-OPC Server) can be set here. The priority setting is primarily intended for the X-OPC Server, which processes a large volume of data from various controllers.

<table>
<tr><th>Name</th><th>Data type</th><th>R/W</th><th>Description</th></tr>
<tr><td colspan="4">The following statuses and parameters can be assigned global variables and used in the user program.</td></tr>
<tr><td>Fragment Definition</td><td>-</td><td>-</td><td>The Priority column is used to define how often this fragment should be received compared to the other fragments.
A fragment in the HIMax, HIQuad X and HIMatrix is ≤ 1100 bytes.
Default setting: Priority 1.
Range of values: Priority 1 (highest) to 65535 (lowest).</td></tr>
<tr><td>Fragment Version State</td><td>UINT</td><td>R</td><td>Reload version state of this safe<b>ethernet</b> fragment, see also the <i>Reload</i> status in Chapter 4.11.1.
unknown: 0x0000
up-to-date 0x0001
updated 0x0002
outdated 0x0003</td></tr>
<tr><td>Timestamp [ms]</td><td>UDINT</td><td>R</td><td>Millisecond fraction of the timestamp (current system time).</td></tr>
<tr><td>Timestamp [s]</td><td>UDINT</td><td>R</td><td>Second fraction of the timestamp (current system time).</td></tr>
<tr><td>Fragment State</td><td>UINT</td><td>R</td><td><table>
<tr><th>Status</th><th>Description</th></tr>
<tr><td>0</td><td>CLOSED: Connection is closed.</td></tr>
<tr><td>1</td><td>TRY_OPEN: An attempt is made to open the connection, but it is still closed.</td></tr>
<tr><td>2</td><td>CONNECTED: The connection exists and the current fragment data has been received (cf. timestamp). As long as no fragment data has been received, the fragment state is set to TRY_OPEN while the connection is being established.</td></tr>
</table>
The connection state of the safe<b>ethernet</b> Editor is set to CONNECTED as soon as the connection is open. Unlike the Fragment state, no data needs to have been exchanged yet.</td></tr>
</table>

Table 39: The Fragment Definitions Tab

4.7 Network Structures for safeethernet Connections

This chapter lists a number of combinations for safeethernet connections.

i To reduce security risks, HIMA recommends setting up a safety network via the CPU modules and a separate standard network via the COM modules. The standard network is used to connect to non-safety components such as X-OPC Server.

Logically, a safe**ethernet** connection is always a connection between two HIMA systems that can be configured as 1-channel or 2-channel.
The Ethernet interfaces available for a safe**ethernet** connection are always displayed related to the resource for which the safe**ethernet** Editor was opened.
All Ethernet interfaces available for a controller are shown in the drop-down menu for the respective **IF Channel**... parameter.

Element	Description
IF Channel 1 (local)	Ethernet interface of the resource for which the safe**ethernet** Editor has been opened.
IF Channel 2 (local)	
IF Channel 1 (remote)	Ethernet interface of the partner resource
IF Channel 2 (remote)	

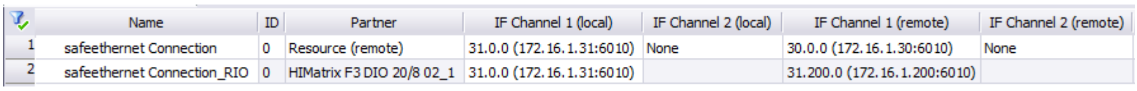
Table 40: Available Ethernet Interfaces

i When designing the network structure and calculating the maximum latency, HIMA recommends consulting a network expert.
A faulty network structure can cause a part of or the entire HIMA system to shut down.

In accordance with the generally accepted regulations for developing Ethernet networks, no network loop may occur. Data packets may only reach a controller over a single path.

4.7.1 Mono safe**ethernet** Connection (Channel 1)

For a mono connection, configure the Ethernet interfaces *IF Channel 1 (local)* and *IF Channel 1 (remote)* within the connection. Remove any automatically entered *IF Channel 2.*



	Name	ID	Partner	IF Channel 1 (local)	IF Channel 2 (local)	IF Channel 1 (remote)	IF Channel 2 (remote)
1	safeethernet Connection	0	Resource (remote)	31.0.0 (172.16.1.31:6010)	None	30.0.0 (172.16.1.30:6010)	None
2	safeethernet Connection_RIO	0	HIMatrix F3 DIO 20/8 02_1	31.0.0 (172.16.1.31:6010)		31.200.0 (172.16.1.200:6010)	

Figure 8: safe**ethernet** Overview of the Example in Figure 9

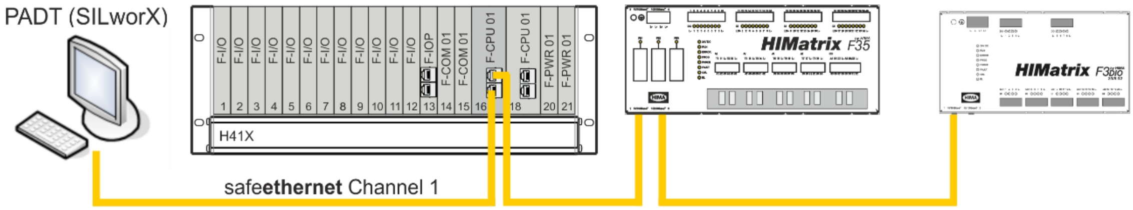


Figure 9: Mono safeethernet Connection (Channel 1)

All HIMA systems programmed with SILworX are suitable for mono safe**ethernet** connection.

4.7.2 Redundant safe**ethernet** Connection (Channel 1 and Channel 2)

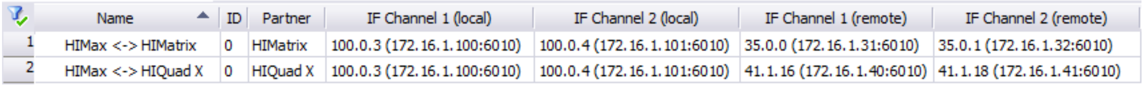
Redundant safe**ethernet** transport paths between two HIMA controllers are possible.
For a redundant connection, the following Ethernet interfaces can be used:

- Ethernet interfaces *IF Channel 1 (local)* and *IF Channel 1 (remote)* for channel 1.
- Ethernet interfaces *IF Channel 2 (local)* and *IF Channel 2 (remote)* for channel 2.

i The redundant transport paths must be sufficiently homogeneous to ensure that the bandwidth and the delay on the two transport paths are nearly identical.
Once the offset of the received messages becomes too large or the messages arrive delayed by more than the response time, the diagnostic function for the transport path no longer operates as intended and considers these delays as a fault of the transport path.
To evaluate the transport path diagnostics, refer to the system variables *State of the Red. Channel* and *Channel State*.

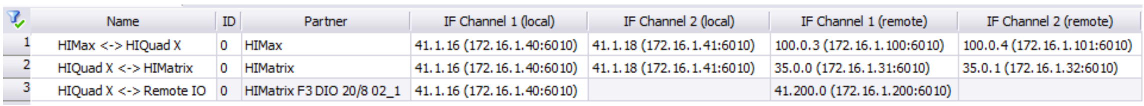
4.7.2.1 Redundant safe**ethernet** Connection to Multiple Systems

A redundant connection to two separate logical and physical transmission paths (channel 1 and channel 2) can be established with HIMA controllers. To allow all three controllers to exchange safe**ethernet** data with one another, at least one safe**ethernet** connection each must be configured between them. In the safe**ethernet** overview, this looks similar to the one in the following figures.



	Name	ID	Partner	IF Channel 1 (local)	IF Channel 2 (local)	IF Channel 1 (remote)	IF Channel 2 (remote)
1	HIMax <-> HIMatrix	0	HIMatrix	100.0.3 (172.16.1.100:6010)	100.0.4 (172.16.1.101:6010)	35.0.0 (172.16.1.31:6010)	35.0.1 (172.16.1.32:6010)
2	HIMax <-> HIQuad X	0	HIQuad X	100.0.3 (172.16.1.100:6010)	100.0.4 (172.16.1.101:6010)	41.1.16 (172.16.1.40:6010)	41.1.18 (172.16.1.41:6010)

Figure 10: HIMax Resource: safe**ethernet** Overview of the Example in Figure 12



	Name	ID	Partner	IF Channel 1 (local)	IF Channel 2 (local)	IF Channel 1 (remote)	IF Channel 2 (remote)
1	HIMax <-> HIQuad X	0	HIMax	41.1.16 (172.16.1.40:6010)	41.1.18 (172.16.1.41:6010)	100.0.3 (172.16.1.100:6010)	100.0.4 (172.16.1.101:6010)
2	HIQuad X <-> HIMatrix	0	HIMatrix	41.1.16 (172.16.1.40:6010)	41.1.18 (172.16.1.41:6010)	35.0.0 (172.16.1.31:6010)	35.0.1 (172.16.1.32:6010)
3	HIQuad X <-> Remote IO	0	HIMatrix F3 DIO 20/8 02_1	41.1.16 (172.16.1.40:6010)		41.200.0 (172.16.1.200:6010)	

Figure 11: HIQuad X Resource: safe**ethernet** Overview of the Example in Figure 12

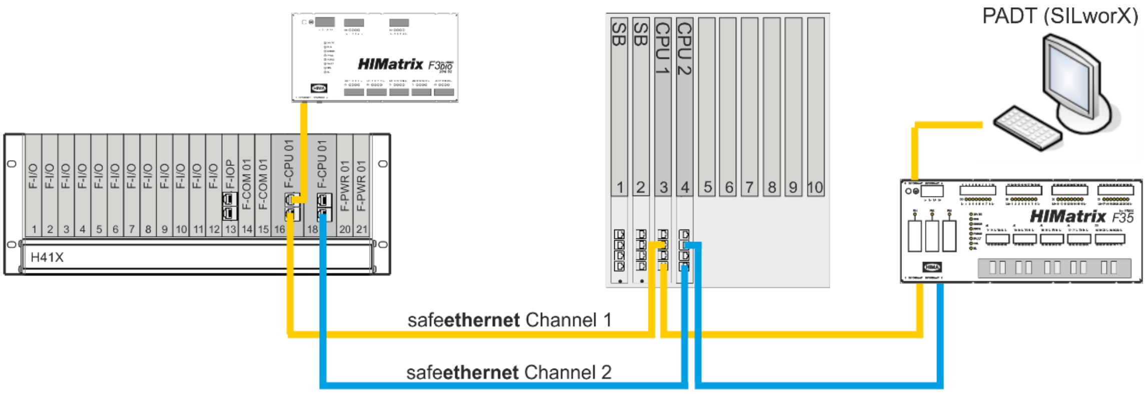


Figure 12: Parallel safe**ethernet** Redundancy

If HIMatrix controllers are used, the switch ports are separated from one another via VLAN, see Chapter 4.6.3.4. Remote I/Os are not suitable for parallel safe**ethernet** connection.

4.7.2.2 Redundancy via safe**ethernet** Ring

A redundant connection in accordance with IEC 62439-3 is also possible with a ring topology. In a ring network, data packet transmission is doubled, i.e., in both directions. Even if the transmission path is interrupted at any point in the safe**ethernet** ring, transmission is ensured.

The safe**ethernet** connection must be established via a ring switch in the ring topology. To this end, a suitable switch with ring management must be used.

In a safe**ethernet** ring, HIMax, HIQuad X and HIMatrix can be interconnected. These controllers only use one IP address each for safe**ethernet** communication.

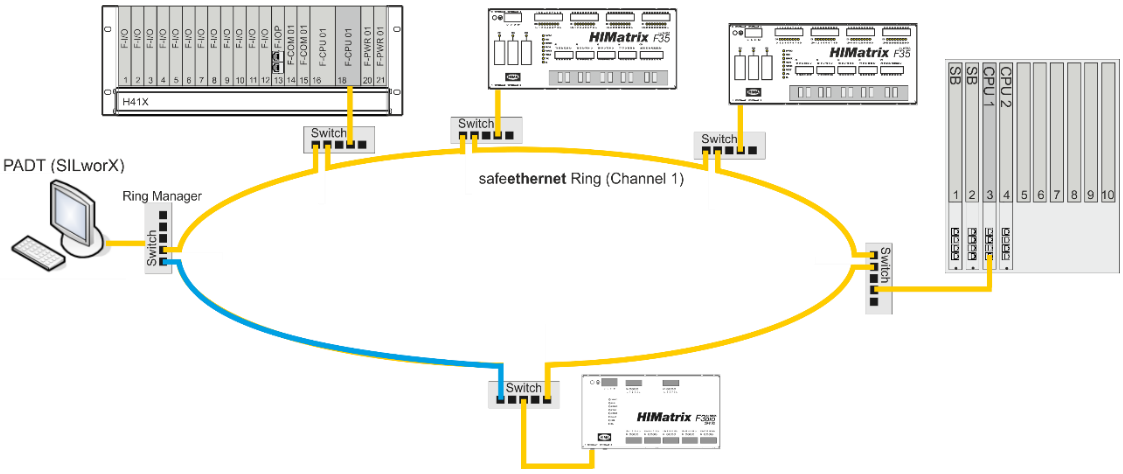


Figure 13: safe**ethernet** Ring Topology

i

Contact HIMA technical support for recommended switches and media converters!

4.8 safeethernet Parameters

Safety-related communication is configured in the safe**ethernet** Editor. The parameters described in this chapter must be set.

For determining the *Receive Timeout* and *Response Time* safe**ethernet** parameters, the following condition applies:

The communication time slice must be sufficiently high to allow all the safe**ethernet** connections to be processed within one CPU cycle, see Chapter 7.1.

4.8.1 Calculating a Suitable Watchdog Time (Max. Cycle Time)

A conservative calculation of the watchdog time for the system in use (HIMax, HIMatrix or HIQuad X) is described in the safety manual for the respective controller.

The maximum cycle time values during the reload depend on the configured watchdog time. If the system should be optimized to the lowest possible watchdog time, the value of the **configured** watchdog time must be gradually reduced in a series of measurements.

In the following cases, contact HIMA technical support:

- If the prerequisites for the strategy described in the safety manual for determining the watchdog time cannot be complied with.
- If the result is not satisfying.

HIMA systems allow settings that ensure an even better performance. In-depth knowledge in several areas is required to identify these settings.

4.8.2 Receive Timeout

Receive Timeout is the monitoring time in milliseconds (ms) within which a valid response from the communication partner must be received.

If a correct response is not received from the communication partner within *Receive Timeout*, safety-related communication is terminated. The import variables of this safe**ethernet** connection behave in accordance with the preset *Freeze Data on Lost Connection [ms]* parameter.

For safety-related functions implemented via safe**ethernet** , *Use Initial Value* is the only setting which may be used.

Since *Receive Timeout* is a safety-relevant component of the worst case response time TR (see Chapter 4.9.1 et seqq.), its value must be determined as described below and entered in the safe**ethernet** Editor.

Receive timeout ≥ 4 * delay + 5 * max. cycle time

Condition: The communication time slice must be sufficiently high to allow all the safe**ethernet** connections to be processed within one CPU cycle.

Delay:	Delay on the transport path, e.g., due to switch or satellite.
Max. Cycle Time	Maximum cycle time of both controllers.

i The availability of the safe**ethernet** communication can be increased by incrementing the *Receive Timeout* value (e.g., by doubling it), provided that the configured time is still sufficient to perform the safety-related function (worst case response time).

The plant manufacturer and the operator are responsible for ensuring that the safe**ethernet** connection complies at least with the following condition: Receive Timeout ≥ 2*Response Time.

4.8.3 Response Time

Response Time is the time period expressed in milliseconds (ms) until the sender of the message receives acknowledgement from the recipient.

When configuring parameters using a safe**ethernet** profile, the Response Time expected to result from the physical circumstances of the transmission path must be set.

The preset Response Time affects the configuration of all the safeethernet connection parameters and is calculated as follows:

Response Time ≤ Receive Timeout / n

n = 2, 3, 4, 5, 6, 7, 8 …

The ratio between Receive Timeout and Response Time influences the capability of tolerating faults, e.g., when packets are lost (resending lost data packets) or delays occur on the transport path.

In networks where packets can be lost, the following condition must be given:

[2.5*Max. Cycle Time + 2*Delay] ≤ **Min. Response Time** ≤ [Receive Timeout / 2]

If this condition is met, the loss of at least one data packet can be intercepted without interrupting the safe**ethernet** connection.

i If this condition is not met, the availability of a safe**ethernet** connection can only be ensured in a collision and noise-free network. However, this is not a safety problem for the processor module!

i It must be ensured that the transport path complies with the configured *Response Time*!

If this cannot always be ensured, a corresponding safe**ethernet** connection system variable for monitoring the *Response Time* is available. If the configured *Response Time* is frequently exceeded, HIMA urgently recommends increasing the its value.

The receive timeout must be adjusted according to the new value configured for response time.

The plant manufacturer and the operator are responsible for ensuring that the safeethernet connection complies at least with the following condition: *Receive Timeout* ≥ *2*Response Time*.

4.8.4 Sync/Async

Sync	Function is not currently supported.
Async	This is the default setting. If Async is set, data is received by the safe**ethernet** protocol instance during the CPU input phase and are sent during the CPU output phase in accordance with the sending rules.

4.8.5 Resend Timeout

Resend Timeout cannot be set manually, but it is calculated based on the profile and *Response Time*.

Monitoring time expressed in milliseconds (ms) and set in controller 1, within which controller 2 must have acknowledged the receipt of a data packet; upon expiration of this period, the data packet is sent again.

Automatic calculation in accordance with the following rule:
Resend Timeout ≤ Receive Timeout

If the *Resend Timeout* set in the communication partners differ, the active protocol partner (with the lowest system ID) determines the *Resend Timeout* for the protocol connection.

4.8.6 Acknowledge Timeout

Acknowledge Timeout cannot be set manually, but it is calculated based on the profile and Response Time.

Acknowledge Timeout is the time period within which the CPU must acknowledge the receipt of a data packet.

In a rapid network, *Acknowledge Timeout* is zero, i.e., the receipt of a data packet is acknowledged immediately. In a slow network (e.g., a telephone modem line), *Acknowledge Timeout* is greater than zero. In this case, the system attempts to transmit the acknowledgment message together with the process data to reduce the network load by avoiding addressing and security blocks.

Automatic calculation in accordance with the following rules:

- **Acknowledge Timeout must be ≤ Receive Timeout.**
- **Acknowledge Timeout must be ≤ Resend Timeout if Production Rate > Resend Timeout.**

4.8.7 Production Rate

Production Rate cannot be set manually, but it is calculated based on the profile and *Response Time*.

Minimum time interval in milliseconds (ms) between two data packets.

The *Production Rate* is used to limit the volume of data packets and prevent a (slow) communication channel from being overloaded. This ensures a uniform load of the transmission medium and prevents the receiver from receiving obsolete data.

Automatic calculation in accordance with the following rules:

- **Production Rate ≤ Receive Timeout**
- **Production Rate ≤ Resend Timeout, if Acknowledge Timeout > Resend Timeout.**

i A zero production rate means that data packets are transferred in every user program cycle.

4.8.8 Queue

Queue cannot be set manually, but it is calculated based on the profile and *Response Time*.

Queue is the number of data packets that can be sent with no need to wait for their acknowledgement. The value depends on the network's transfer capacity and potential network delays.

All safeethernet connections share the available message queue space in the CPU.

4.9 Worst Case Response Time for safeethernet

In the examples from Chapter 4.9.3 on, the formulas for calculating the worst case response time only apply for a connection to HIMatrix controllers if the parameter Safety Time = 2 * Watchdog Time is set. These formulas always apply to HIMaxand HIQuad X controllers.

i The allowed worst case response time depends on the process and must be agreed upon together with the competent test authority.

The following table describes the parameters and conditions that must be taken into account in SILworX to calculate the worst case response time:

Terms	Description
Receive Timeout	Monitoring time of controller 1 (PES 1) within which a valid response from controller 2 (PES 2) must be received. Otherwise, safety-related communication is terminated after the time has expired.
Production Rate	Minimum interval between two data transmissions.
Watchdog time	Maximum duration permitted for a controller's RUN cycle. The duration of the RUN cycle depends on the complexity of the user program and the number of safe**ethernet** connections. The watchdog time (WDT) must be entered in the resource properties.
Worst Case Response Time	The worst case response time is the time between a change in a physical input signal (in) of PES 1 and a change in the physical output signal (out) of PES 2.
Delay	Delay of a transport path, e.g., when a modem or satellite connection is used. For direct connections, an initial delay of 2 ms can be assumed. The responsible network administrator can measure the actual delay on a transport path.

Table 41: safe**ethernet** Parameter Description and Conditions

The following conditions apply to the calculations of the maximum response times specified below:

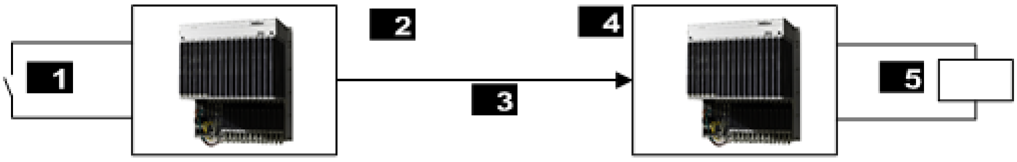
- The signals transmitted over safe**ethernet** must be processed in the corresponding controllers within one CPU cycle.
- The response times of the sensors and the actuators must also be added up.

The calculations also apply to signals in the opposite direction.

i HIMA systems allow settings that ensure an even better performance. In-depth knowledge in several areas is required to identify these settings.

4.9.1 Worst Case Response Time of 2 HIMax Controllers

The worst case response time T_R is the time between a change on the sensor input signal (in) of controller 1 and a response on the corresponding output (out) of controller 2. It is calculated as follows:



1 Input
2 HIMax controller 1
3 Safety-related protocol
4 HIMax controller 2
5 Output

Figure 14: Response Time with Interconnection of 2 HIMax Controllers

$$\mathbf{T_R = t_1 + t_2 + t_3}$$

T_R: Worst case response time.

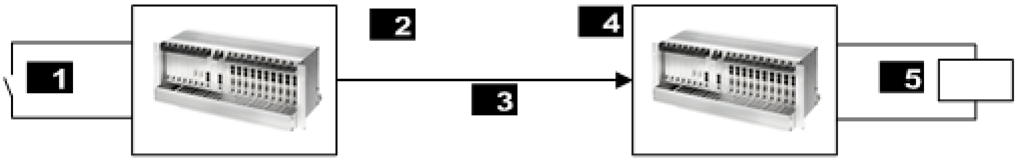
t_1: Safety time of HIMax controller 1.

t_2: Receive timeout.

t_3: Safety time of HIMax controller 2.

4.9.2 Worst Case Response Time of 2 HIQuad X Controllers

The worst case response time T_R is the time between a change on the sensor input signal (in) of controller 1 and a response on the corresponding output (out) of controller 2. It is calculated as follows:



1 Input
2 HIQuad X controller 1
3 Safety-related protocol
4 HIQuad X controller 2
5 Output

Figure 15: Response Time with Interconnection of 2 HIQuad X Controllers

$$\mathbf{T_R = t_1 + t_2 + t_3}$$

T_R: Worst case response time.

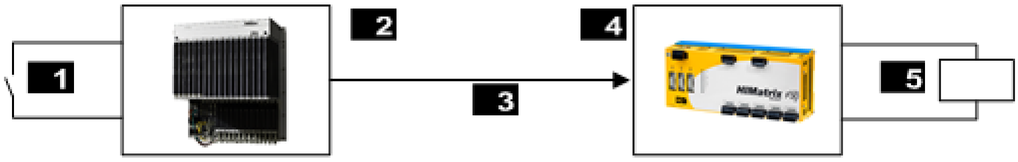
t_1: Safety time of HIQuad X controller 1.

t_2: Receive timeout.

t_3: Safety time of HIQuad X controller 2.

4.9.3 Worst Case Response Time of 1 HIMax Connected to 1 HIMatrix Controller

The worst case response time T_R is the time between a change on the sensor input signal (in) of the HIMax controller and a response on the corresponding output (out) of the HIMatrix controller. It is calculated as follows:



1 Input
2 HIMax controller
3 Safety-related protocol
4 HIMatrix controller
5 Output

Figure 16: Response Time for a HIMax Connected to a HIMatrix Controller

$\mathbf{T_R = t_1 + t_2 + t_3}$

T_R: Worst case response time.

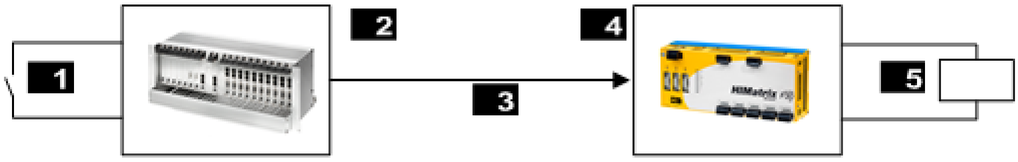
t_1: Safety time of the HIMax controller.

t_2: Receive timeout.

t_3: 2 * watchdog time of the HIMatrix controller.

4.9.4 Worst Case Response Time of 1 HIQuad X Connected to 1 HIMatrix Controller

The worst case response time TR is the time between a change on the sensor input signal (in) of the HIQuad X controller and a response on the corresponding output (out) of the HIMatrix controller. It is calculated as follows:



1 Input
2 HIQuad X controller
3 Safety-related protocol
4 HIMatrix controller
5 Output

Figure 17: Response Time for a HIQuad X Connected to a HIMatrix Controller

$\mathbf{T_R = t_1 + t_2 + t_3}$

T_R: Worst case response time.

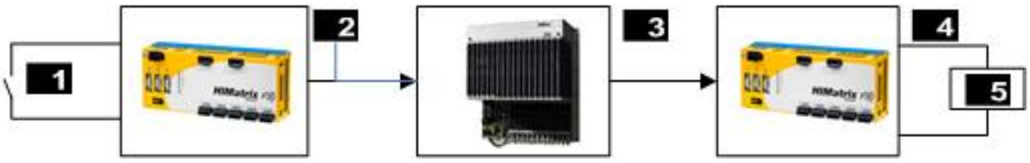
t_1: Safety time of the HIQuad X controller.

t_2: Receive timeout.

t_3: 2 * watchdog time of the HIMatrix controller.

4.9.5 Worst Case Response Time of 1 HIMax Connected to 2 HIMatrix Controllers or Remote I/Os

The worst case response time T_R is the time between a change on the sensor input signal (in) of the first HIMatrix controller or remote I/O (e.g., F3 DIO 20/8 01) and a response on the output (out) of the second HIMatrix controller or remote I/O (out). It is calculated as follows:



1 Input
2 Remote I/O 1
3 HIMax controller
4 Remote I/O 2
5 Output

Figure 18: Response Time with 2 Remote I/Os and 1 HIMax Controller

$$\mathbf{T_R = t_1 + t_2 + t_3 + t_4 + t_5}$$

T_R: Worst case response time.

t_1: 2 * watchdog rime of remote I/O 1.

t_2: Receive timeout1

t_3: 2 * watchdog time of the HIMax controller.

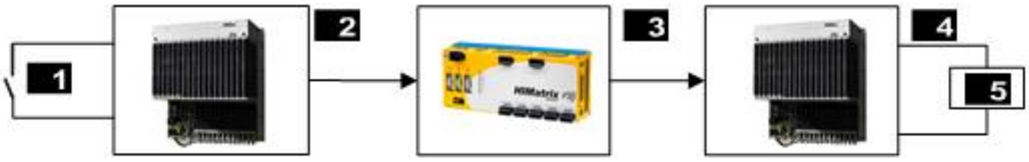
t_4: Receive timeout2

t_5: 2 * watchdog time of remote I/O 2.

Remote I/O 1 and remote I/O 2 can also be identical. The time values still apply if a HIMatrix controller is used instead of a remote I/O.

4.9.6 Worst Case Response Time of 1 HIMatrix Connected to 2 HIMax Controllers

The worst case response time T_R is the time between a change on the sensor input signal (in) of the first HIMax controller and a response on the corresponding output (out) of the second HIMax controller. It is calculated as follows:



1 Input
2 HIMax controller 1
3 HIMatrix controller
4 HIMax controller 2
5 Output

Figure 19: Response Time with 2 HIMax Controllers and 1 HIMatrix Controller

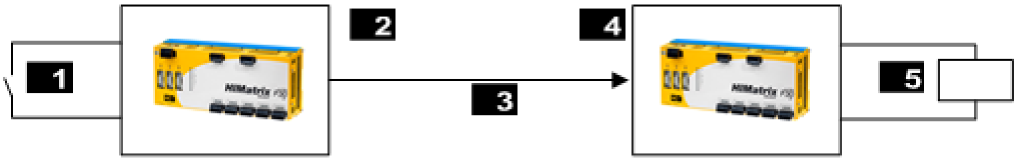
$T_R = t_1 + t_2 + t_3 + t_4 + t_5$

T_R: Worst case response time.
t_1: Safety time of HIMax controller 1.
t_2: Receive timeout1
t_3: 2 * watchdog time of the HIMatrix controller.
t_4: Receive timeout2
t_5: Safety time of HIMax controller 2.

Both HIMax controllers, 1 and 2, can also be identical.
The HIMatrix controller can also be a HIMax controller.

4.9.7 Worst Case Response Time of 2 HIMatrix Controllers

The worst case response time T_R is the time between a change on the sensor input signal of controller 1 and a response on the corresponding output of controller 2. It is calculated as follows:



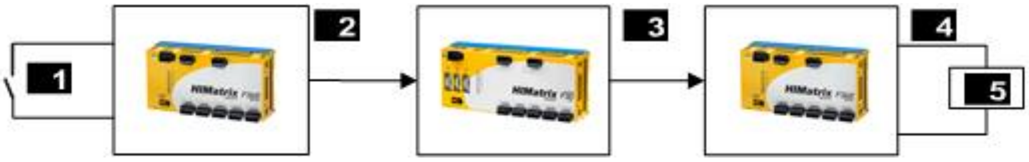
1 Input
2 HIMatrix controller 1
3 Safety-related protocol
4 HIMatrix controller 2
5 Output

Figure 20: Response Time with Interconnection of 2 HIMatrix Controllers

$T_R = t_1 + t_2 + t_3$
T_R: Worst case response time.
t_1: 2 * watchdog time of the HIMatrix controller 1.
t_2: Receive timeout.
t_3: 2 * watchdog time of the HIMatrix controller 2.

4.9.8 Worst Case Response Time of 1 HIMatrix Controller connected to 2 Remote I/Os

The worst case response time T_R is the time between a change on the sensor input signal (in) of the first HIMatrix controller or remote I/O (e.g., F3 DIO 20/8 01) and a response on the corresponding output of the second HIMatrix controller or remote I/O (out). It is calculated as follows:



1 Input
2 Remote I/O 1
3 HIMatrix controller
4 Remote I/O 2
5 Output

Figure 21: Response Time with Remote I/Os

$T_R = t_1 + t_2 + t_3 + t_4 + t_5$

T_R: Worst case response time.
t_1: 2 * watchdog rime of remote I/O 1.
t_2: Receive timeout1.
t_3: 2 * watchdog time of the HIMatrix controller.
t_4: Receive timeout2.
t_5: 2 * watchdog time of remote I/O 2.

Note: Remote I/O 1 and remote I/O 2 can also be identical. The time values still apply if a HIMatrix controller is used instead of a remote I/O.

4.10 safe**ethernet Profile**

safe**ethernet** profiles are combinations of parameters compatible with one another that are automatically set when one of the safe**ethernet** profiles is selected.
When configuring, only the receive timeout and the expected response time parameters must be set individually.
A safe**ethernet** profile is used to optimize the data throughput in a network in light of the physical circumstances.

To ensure that the optimization is effective, the following conditions must be met:

- The communication time slice value must be sufficiently large to allow all the safe**ethernet** connections to be processed within one CPU cycle.
- Average CPU cycle time < response time.
- Average CPU cycle time < ProdRate or ProdRate = 0.

i Unsuitable combinations of CPU cycle, communication time slice, response time and production rate are not rejected during code generation and download/reload. However, these combinations can cause communication disturbances up to and including a failure of the safe**ethernet** communication.

In the Control Panels of the two controllers, verify the *Faulty Messages* and *Resends* values.

6 safe**ethernet** profiles are available. The safe**ethernet** profile most suitable for the transmission path can be selected from these.

For a safe**ethernet** connection with high availability, HIMA recommends using the *Fast&Noisy*, *Medium&Noisy* or *Slow&Noisy* profile.

Fast & Cleanroom	Only recommended for network free from interference.
Fast & Noisy	Recommended for high availability of the safe**ethernet** connection.
Medium & Cleanroom	Only recommended for network free from interference.
Medium & Noisy	Recommended for high availability of the safe**ethernet** connection.
Slow & Cleanroom	Only recommended for network free from interference.
Slow & Noisy	Recommended for high availability of the safe**ethernet** connection.
Fixed	Starting with V4, a modified calculation applies to all Cleanroom profiles. If a project created with a SILworX version prior to V4 should be converted, the configured profile must be set to Fixed to ensure that the CRC does not change.

4.10.1 Profile I (Fast&Cleanroom)

i For a safe**ethernet** connection with high availability, HIMA recommends using the *Fast&Noisy*, *Medium&Noisy* or *Slow&Noisy* profile.
The use of the Cleanroom profile is only recommended for networks free from interference, see Chapter 4.2.

Use

The *Fast&Cleanroom* profile is suitable for applications in ideal environments such as laboratories!

- For the fastest data throughput.
- For applications that require fast data transmission.
- For application that require a worst case response time as low as possible.

Network requirements:

- Fast: 100 Mbit technology (100Base Tx), 1 Gbit technology.
- Clean: Noise-free network.
  Data loss due to network overload, external influences or network manipulation must be prevented.
- LAN switches are necessary!

Communication path characteristics:

- Minimum delays.
- Expected Response Time ≤ Receive Timeout
  (otherwise ERROR during configuration).

4.10.2 Profile II (Fast & Noisy)

Use

The *Fast&Noisy* profile is the SILworX standard profile for communicating via safe**ethernet**.

- For fast data throughput.
- For applications that require fast data transmission.
- For applications that require a worst case response time as low as possible.

Network requirements:

- Fast: 100 Mbit technology (100Base Tx), 1 Gbit technology.
- Noisy: Interference within the network.
  Low probability of data packet loss. Time for ≥ 1 resend(s).
- LAN switches are necessary!

Communication path characteristics:

- Minimum delays.
- Expected Response Time ≤ Receive Timeout / 2
  (otherwise ERROR during configuration).

4.10.3 Profile III (Medium&Cleanroom)

i For a safe**ethernet** connection with high availability, HIMA recommends using the *Fast&Noisy*, *Medium&Noisy* or *Slow&Noisy* profile.
The use of the Cleanroom profile is only recommended for networks free from interference, see Chapter 4.2.

Use

The *Medium&Cleanroom* profile is only suitable for applications in a network free from interference and requiring a moderately fast data transmission rate.

- For medium data throughput.
- Suitable for VPN (virtual private networks) in which data is exchanged slowly, but without faults since intermediate safety devices (e.g., firewalls, encryption) are used.
- Suitable for applications in which the worst case response time is not a critical factor.

Network requirements

- Medium: 10 Mbit (10BASE-T), 100 Mbit (100BASE-Tx), 1 Gbit technology.
- LAN switches are necessary!
- Clean: Noise-free network.
  Data loss due to network overload, external influences or network manipulation must be prevented; time for ≥ 0 resends.

Communication path characteristics

- Moderate delays.
- Expected Response Time ≤ Receive Timeout (otherwise ERROR during configuration).

4.10.4 Profile IV (Medium&Noisy)

Use

The *Medium&Noisy* profile is suitable for applications that require moderate fast data transmission.

- For medium data throughput.
- For applications that require moderate fast data transmission.
- Suitable for applications in which the worst case response time is not a critical factor.

Network requirements

- Medium: 10 Mbit (10BASE-T), 100 Mbit (100BASE-Tx), 1 Gbit technology.
- LAN switches are necessary!
- Noisy: Interference within the network.
  Low probability of data packet loss. Time for ≥ 1 resend(s).

Communication path characteristics

- Moderate delays.
- Expected Response Time ≤ Receive Timeout / 2 (otherwise ERROR during configuration).

4.10.5 Profile V (Slow&Cleanroom)

i For a safe**ethernet** connection with high availability, HIMA recommends using the *Fast&Noisy*, *Medium&Noisy* or *Slow&Noisy* profile.
The use of the Cleanroom profile is only recommended for networks free from interference, see Chapter 4.2.

Use

The *Slow&Cleanroom* profile is suitable for applications in a network free from interference and requiring a slow data transmission rate.

- For slow data throughput.
- For applications that only require a slow data transmission rate to controllers (potentially located far away) or if the communication path conditions cannot be defined in advance.

Network requirements

- Slow: Data transmission via ISDN, dedicated line or radio relay.
- Clean: Network free from interference.
  Data loss due to network overload, external influences or network manipulation must be prevented; time for ≥ 0 resends.

Communication path characteristics

- Moderate delays.
- Expected Response Time = Receive Timeout (otherwise ERROR during configuration)

4.10.6 Profile VI (Slow&Noisy)

Use

The *Slow&Noisy* profile is suitable for applications that only require a slow data transmission rate to the controllers (potentially located far away).

- For slow data throughput.
- Generally for applications and data transfer via bad telephone lines or disturbed radio relays.

Network requirements

- Slow: Data transmission via telephone, satellite, radio etc.
- Noisy: Interference within the network.
  Low probability of data packet loss. Time for ≥ 1 resend(s).

Communication path characteristics

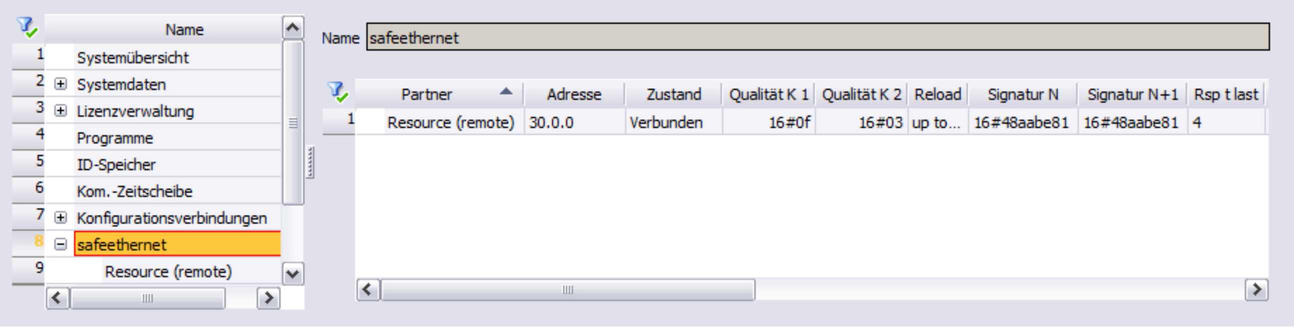
- Moderate to significant delays.
- Expected Response Time ≤ Receive Timeout / 2 (otherwise ERROR during configuration).

4.11 Control Panel (safe**ethernet**)

The Control Panel can be used to verify the safe**ethernet** connection settings. It also displays details about the current status of the safe**ethernet** connection (e.g., cycle time, bus state, etc.).

To open Control Panel for monitoring the safeethernet connection

1. In the structure tree, select Resource.
2. Select Online from the resource context menu.
3. In the **System Log-in** window, enter the access data to open the Control Panel for the resource.
4. In the structure tree associated with the Control Panel, select safe**ethernet** .



	Name
1	Systemübersicht
2	⊞ Systemdaten
3	⊞ Lizenzverwaltung
4	Programme
5	ID-Speicher
6	Kom.-Zeitscheibe
7	⊞ Konfigurationsverbindungen
8	⊟ safeethernet
9	Resource (remote)

Name: safeethernet

	Partner	Adresse	Zustand	Qualität K 1	Qualität K 2	Reload	Signatur N	Signatur N+1	Rsp t last
1	Resource (remote)	30.0.0	Verbunden	16#0f	16#03	up to...	16#48aabe81	16#48aabe81	4

Figure 22: Control Panel for safeethernet Connection Overview

To reset the statistical data of the safeethernet connection

This context menu function is used to reset the statistical data (e.g., min./max. cycle time etc.) to zero.

1. Select safeethernet connection in the structure tree.
2. In the context menu of the safeethernet connection, select Reset safe**ethernet** Statistics.

4.11.1 View Box (safe**ethernet** Connection)

The view box displays the following values of the selected safe**ethernet** connection.

Element	Description
Partner	Resource name of the communication partner
Address	System ID
State	State of the safe**ethernet** connection. (See also Chapter 4.6).
Quality Ch 1	Quality of transport path, Channel 1. (See also Chapter 4.6).
Quality Ch 2	Quality of transport path, Channel 2. (See also Chapter 4.6).

Element	Description
Reload	safeethernet reload status unknown: State of the loaded partner signatures is unknown: - There is no connection. - The partner has an old operating system without the safe**ethernet** reload function. updated: The current code has been loaded into this controller; it still has to be loaded into the partner. outdated: The current code has been loaded into the partner; it still has to be loaded into this controller. up-to-date: Both partners have an identical N+1 signature.
Signature N	Changing the safe**ethernet** configuration results in a dual configuration. Old signature of the safe**ethernet** configuration.
Signature N+1	New signature of the safe**ethernet** configuration.
Rsp t last	Actual response time as minimum, maximum, last and average value. See also Chapter 4.8.3.
Rsp t avg	
Rsp t min	
Rsp t max	
Error	Bad Messages Number of rejected messages since statistics reset.
Rsnd	Number of resends since statistics reset.
Succeeded	Number of successful connections since statistics reset.
Early	Early Queue Usage Number of early messages since statistics reset. Early messages are stored in the Early Queue.
Frame	Frame No. Revolving send counter.
Ack Frame	Ack.Frame No. Revolving receive counter.
Monotony	Revolving user data send counter
Receive Timeout	Receive Timeout [ms] (See also Chapter 4.8.2)
Resend Timeout	Resend timeout [ms] (See also Chapter 4.8.5)
Acknowledgment Timeout	Acknowledge Timeout [ms] (See also Chapter 4.8.6)
Conn Ctrl	Connection Control
Ctrl Ch 1	Transmission Control for Channel 1 (See also Chapter 4.6).
Ctrl Ch 2	Transmission Control for Channel 2 (See also Chapter 4.6).
Protocol	0-1: Protocol version for ELOP II Factory resources. 2: Protocol version for SILworX resources.

Table 42: View Box of the safe**ethernet** Connection

4.12 safeethernet Reload

Thanks to this feature, changes performed to a safe**ethernet** configuration can be loaded during operation by performing a reload to the controller while the safeethernet connection continues to run with no interruptions.

4.12.1 Requirements

safe**ethernet** reload is possible for HIMax, HIMatrix and HIQuad X. The following system requirements apply to all controllers participating in safe**ethernet** the connection:

- HIMax as of CPU OS V6 and COM OS V6.
- HIQuad X ab CPU BS V10 und COM BS V10.
- HIMatrix as of CPU OS V10 and COM OS V15.

The above-mentioned COM OS versions or higher are required to ensure that safe**ethernet** connections are properly routed via the COM module, see Chapter 4.12.7.

In the properties of the safe**ethernet** connection, set the *Codegen* parameter to V6 and higher.

If a redundant module is available, the operating systems of HIMax modules can be updated during operation. This ensures that the conversion to safe**ethernet** reload is performed without interruptions, even in HIMax plants using previous operating systems.

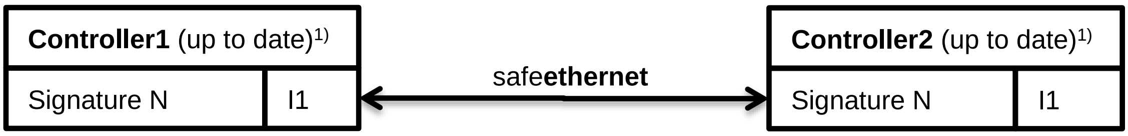
4.12.2 Technical Concept

The safe**ethernet** signature is a CRC code used to uniquely identify the safe**ethernet** configuration. The safe**ethernet** signature is created during the code generation and is part of the loaded configuration.

safe**ethernet** communication between 2 communication partners can only occur if both partners have the same safe**ethernet** configuration with identical signature.

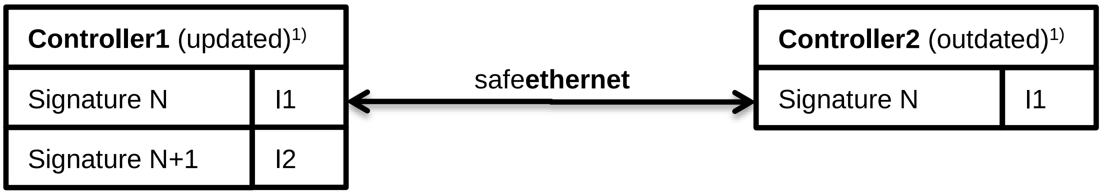
To use reload to perform changes to a safe**ethernet** connection, the controller must be provided with 2 safe**ethernet** configurations and corresponding signatures (N and N+1). This is supported for SILworX as of V6.

In the two controllers, configuration I1 is connected to a safe**ethernet** signature



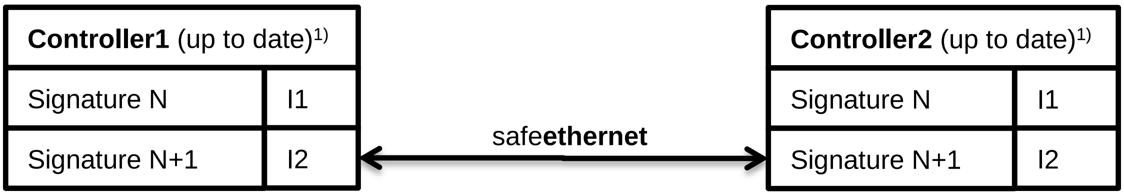
After changing the connection and reload for controller 1, configurations I1 and I2 are available. The previous safe**ethernet** configuration I1 with signature N is still active in Controller1.

Changing the safe**ethernet** configuration results in a dual configuration (in the example: I1+I2). The safe**ethernet** reload state of Controller1 is `updated` and the version state of Controller2 is `outdated`, which signalizes that a reload must be performed in Controller2.



1) safe**ethernet** version state, see Chapter 4.12.5

Upon completion of the reload process for controller 2, the new safe**ethernet** configuration I2 is active with signature N+1. The dual configuration (I1+I2) is now available for both controllers and should be deleted as recommended by performing an additional reload, see Chapter 4.12.3.1.



Controller1 (up to date)[1)]

Signature N	I1
Signature N+1	I2

Controller2 (up to date)[1)]

Signature N	I1
Signature N+1	I2

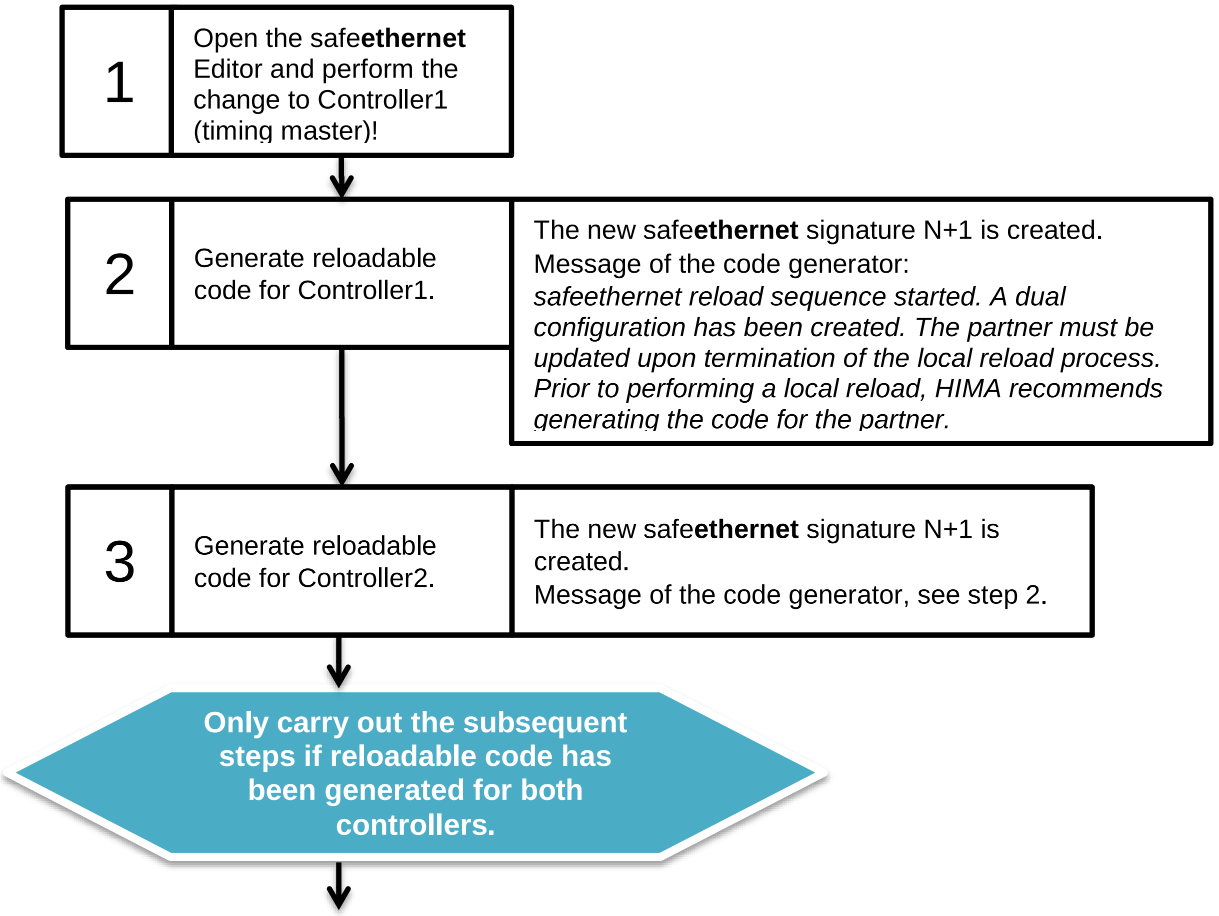
1) safe**ethernet** version state, see Chapter 4.12.5

i With respect to reload, HIMA recommends always starting the process for the controller that is configured as Timing Master of the safeethernet connection. The new safeethernet connection becomes active after the reload procedure is complete for both controllers.

4.12.3 Procedure to Be Observed

safe**ethernet** connections are to be considered holistically, i.e., changes should always be performed on both partners and in direct succession to ensure the consistency of the safe**ethernet** .

The previous safe**ethernet** configuration is active up to step 5. The new safe**ethernet** configuration becomes active after a successful reload in step 5.



1 Open the safe**ethernet** Editor and perform the change to Controller1 (timing master)!

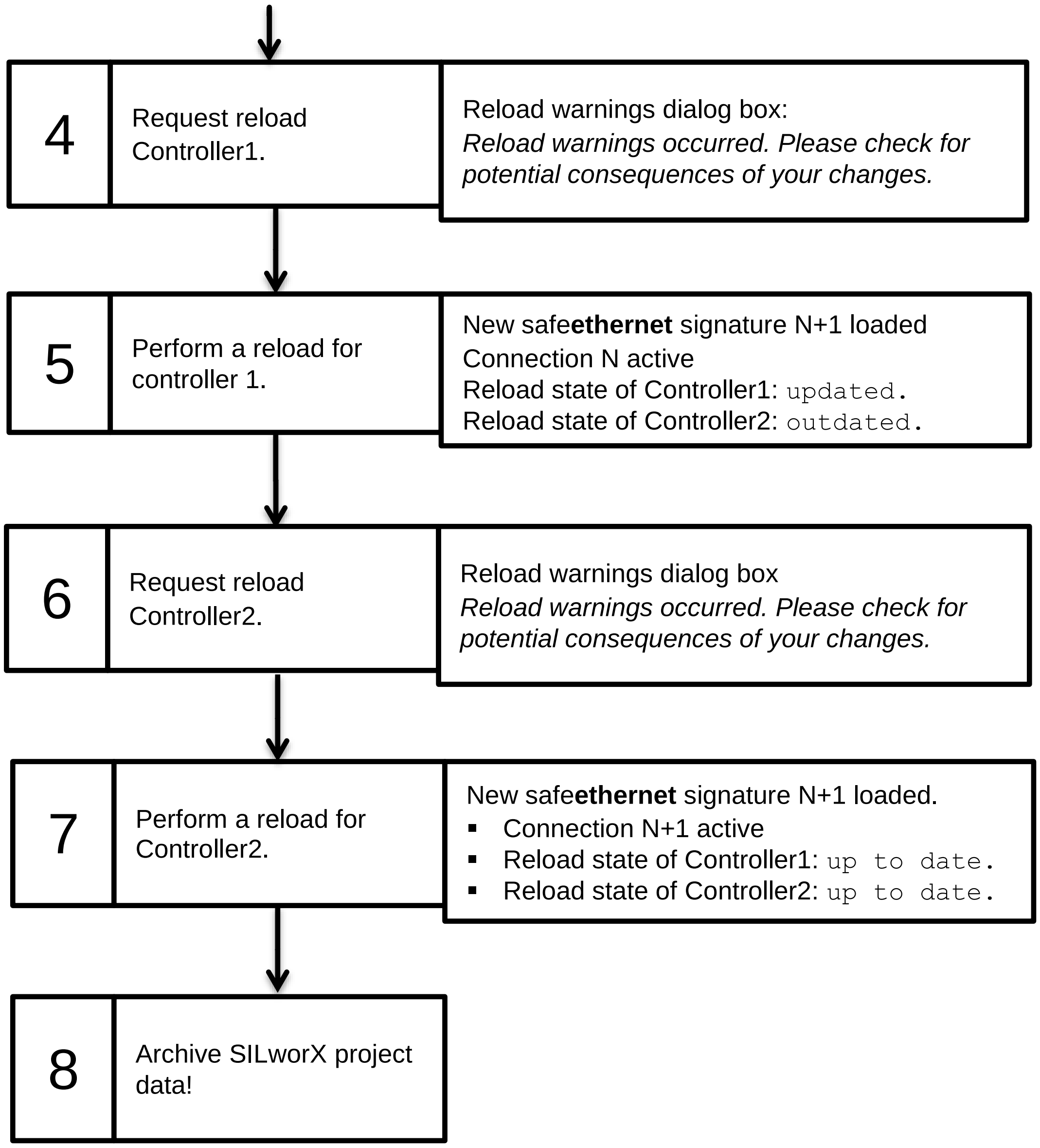
2 Generate reloadable code for Controller1.

The new safe**ethernet** signature N+1 is created.
Message of the code generator:
safeethernet reload sequence started. A dual configuration has been created. The partner must be updated upon termination of the local reload process. Prior to performing a local reload, HIMA recommends generating the code for the partner.

3 Generate reloadable code for Controller2.

The new safe**ethernet** signature N+1 is created.
Message of the code generator, see step 2.

Only carry out the subsequent steps if reloadable code has been generated for both controllers.



4 Request reload Controller1.

Reload warnings dialog box:
Reload warnings occurred. Please check for potential consequences of your changes.

5 Perform a reload for controller 1.

New safe**ethernet** signature N+1 loaded
Connection N active
Reload state of Controller1: `updated`.
Reload state of Controller2: `outdated`.

6 Request reload Controller2.

Reload warnings dialog box
Reload warnings occurred. Please check for potential consequences of your changes.

7 Perform a reload for Controller2.

New safe**ethernet** signature N+1 loaded.

- Connection N+1 active
- Reload state of Controller1: `up to date`.
- Reload state of Controller2: `up to date`.

8 Archive SILworX project data!

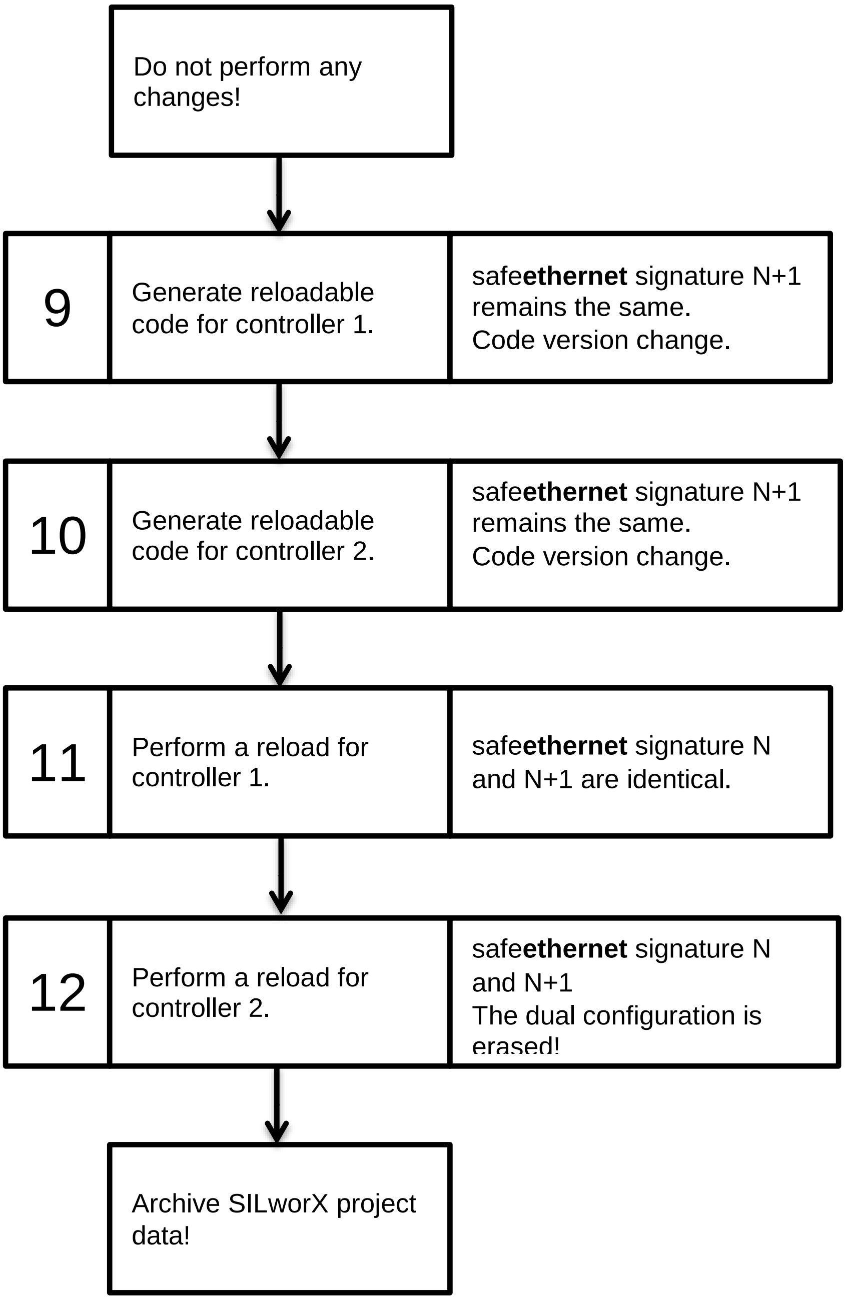
4.12.3.1 Align Signatures N and N+1

Changes to the safe**ethernet** configuration such as described in Chapter 4.12.3 result in a dual configuration. The controllers contain the following 2 configurations:

- The previous configuration with safe**ethernet** signature N through which safe**ethernet** communication is running, remains active until both controllers have been updated.
- The new configuration with safe**ethernet** signature N+1 through which safe**ethernet** communication is running, becomes active after both controllers have been updated.

i After completion of an additional code generation with no safe**ethernet** change, the dual configuration is deleted. This means that the same CRC code is set in the *Signature N* and *Signature N+1* system variables). HIMA recommends always erasing the dual configuration. This must be performed for both resources.

To erase a dual configuration, perform steps 9 through 12 as described below!



Do not perform any changes!

9 Generate reloadable code for controller 1. safe**ethernet** signature N+1 remains the same. Code version change.

10 Generate reloadable code for controller 2. safe**ethernet** signature N+1 remains the same. Code version change.

11 Perform a reload for controller 1. safe**ethernet** signature N and N+1 are identical.

12 Perform a reload for controller 2. safe**ethernet** signature N and N+1 The dual configuration is erased!

Archive SILworX project data!

4.12.4 Integrated Protective Mechanisms

The protective mechanisms integrated in SILworX and in the controller's operating system ensure early detection of unintended interruption or resumption of a safe**ethernet** connection and generate a warning message.

4.12.4.1 Automatic Test during Code Generation

The following table contains the messages output during a code generation and connected to safe**ethernet** reload, informing the user about the current safe**ethernet** reload state.

Information in the code generator dialog	Description
Reload Warning *safeethernet reload sequence started. A dual configuration has been created. The partner must be updated upon termination of the local reload process. Prior to performing a local reload, HIMA recommends generating the code for the partner.*	**Procedure OK!** This information is provided after a change performed to the safe**ethernet** connection and the code generation! Follow the recommended procedure, see Chapter 4.12.2.
Reload Info *Dual configuration safeethernet reload for connection of "safeethernet V1" to "controller2" has been removed.*	**Procedure OK!** A reload has been performed after completion of a new code generation without any safeethernet change. The dual configuration has been erased, i.e., there is once again only one configuration with one safe**ethernet** signature, see Chapter 4.12.3.1.
The safeethernet connection of "safeethernet V1" to "controller2" could be interrupted. Please update this partner. No matching connection version could be found in the partner's download configuration.	**Caution!** Do not perform any reload to ensure that the connection will not be interrupted! Please contact HIMA technical support! With the partner controller, there is no longer a common configuration with identical signature so that no safe**ethernet** reload can be performed.

Table 43: Messages from the Code Generator

4.12.4.2 Automatic Test during the Controller's Reload

Warning messages are only issued before a safe**ethernet** reload if suitable CPU operating systems are loaded in the controllers.

- HIMax CPU OS as of V6.
- HIQuad X ab CPU BS V10.
- HIMatrix CPU OS as of V10.

Prior to performing a reload, the operating system checks the safe**ethernet** reload state to ensure that is suitable for a reload. If a controller detects that a reload could result in the interruption of the safe**ethernet** connection, it generates a corresponding warning message displayed in SILworX. In this scenario, reload can be aborted by the user. After an aborted reload, the controllers continue to operate with the last suitable safe**ethernet** configuration.

Information in the Dialog Box	Description
A reload is to be performed although a safeethernet connection reports the safeethernet reload state updated, partner's safeethernet address: x/x/x. The connection might be lost by activating the configuration. Check for potential consequences.	**Caution!** Do not perform a reload. Please contact HIMA technical support! If reload is performed anyway, the safe**ethernet** connection may be interrupted!
A reload is to be performed although a safeethernet connection reports the safeethernet reload state unknown (i.e., no connection exists to the partner), partner's safeethernet address: x/x/x. If a connection is established before the configuration has been activated, the configuration activation could cause the connection to be lost again. Check for potential consequences.	**Caution!** The unknown safe**ethernet** reload state is reported, if a safe**ethernet** connection is interrupted, see Chapter 4.12.5. Prior to performing a new reload, check the physical connection, e.g., if all the Ethernet cables are properly plugged in.

Table 44: Messages from the Operating System

4.12.5 safe**ethernet** Reload State

The reload state provides information on the current state of the safe**ethernet** connection and about whether suitable safe**ethernet** configurations are loaded or are to be loaded. The consistent procedure applied to safe**ethernet** reload is a requirement for ensuring that the reload status is properly displayed, see Chapter 4.12.3.

The following safe**ethernet** reload state is displayed:

unknown	State of the loaded partner signatures is unknown: ▪ There is no connection. ▪ The partner has an old operating system without the safe**ethernet** reload function.
updated	The current code has been loaded into this controller; it still has to be loaded into the partner.
outdated	The current code has been loaded into the partner; it still has to be loaded into this controller.
up-to-date	Both partners have an identical N+1 signature.

If no suitable configuration is available after a reload, a warning is issued informing the user that the reload can be aborted.

If, however, the reload process is continued in spite of the warning messages described in Chapter 4.12.4, a suitable configuration might no longer be present in the partner controller. The safe**ethernet** connection to the partner controller could be interrupted (CLOSED)!

The safe**ethernet** version state is called Reload in the SILworX Online View of the safeethernet connection. The same information is provided by the *Version State* system variable, which can be assigned a global variable and used as such in the user program.

4.12.6 Maximum Number of safe**ethernet** Connections during Reload

The number of safe**ethernet** connections contained in the controller during reload can be greater than configured. Not only the added safe**ethernet** connections are held, but also the deleted safeethesafe**ethernet** rnet connections since they must remain enabled until the reload is completed.

The maximum number of simultaneous safe**ethernet** connections during reload is as follows:

- HIMax = 300 (max. 255 safe**ethernet** connections + 45 (reload buffer)).
- HIMatrix = 277 (max. 255 safe**ethernet** connections + 22 (reload buffer)).
- HIQuad X = 150 (max. 128 safe**ethernet** connections + 22 (reload buffer)).

These limits are defined to restrict the maximum storage space required during a reload.

i

If during the reload code generation, the maximum number of safe**ethernet** connections allowed for reload is exceeded, the reload code generation is aborted and an error message is issued.
For the maximum number of safe**ethernet** connections between two controllers refer to Chapter 4.3.

If multiple changes are required, these must be performed through multiple consecutive reloads.

4.12.7 safe**ethernet** Connection via the Communication Module

HIMA recommends setting the *Code Generation* parameter of the communication module to V6 and higher to prevent, as far as possible, a cold reload of the communication module. In doing so, the safeethernet connections routed through this communication module are not interrupted, even if changes are performed to variables or parameters, e.g., profiles.

For further details on the safe**ethernet** reload behavior in connection with the communication module and additional changes, refer to the following Chapter 4.12.8.

4.12.8 Changes to the safe**ethernet** Configuration

The following table provides an overview of the changes to the safe**ethernet** configuration and their effects on the safe**ethernet** reload.

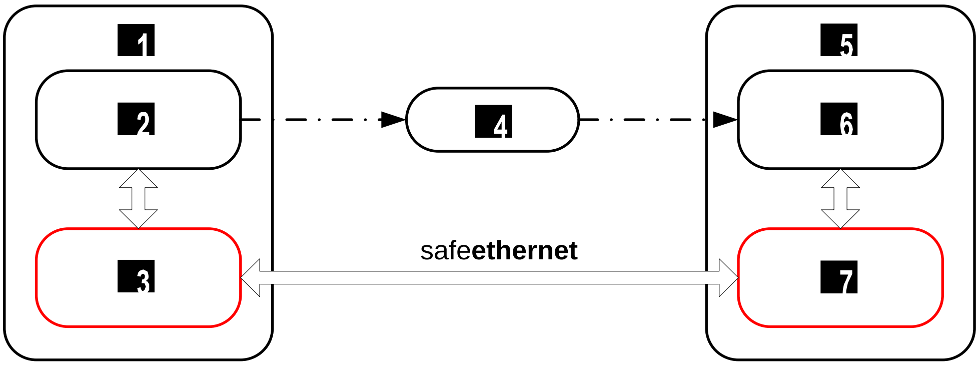
Changes to	CPU	COM
To add or delete global variables for:		
safeethernet	•	•
X-OPC (DA)	•	•
X-OPC (events)	•	•
To change the number of views (X-OPC).	•	•
To add or delete a new safe**ethernet** connection.	•	•[1)]
safe**ethernet** parameters (e.g., *Timing Master*, *Receive Timeout*).	•	•
IP addresses (changed transport path).	•	•[1)]
safe**ethernet** parameter (*Profile*).	-	n. a.
safe**ethernet** parameter (*Behavior on Connection Loss*).	-	n. a.
• safe**ethernet** reload possible. - safe**ethernet** reload not possible. n.a.: not applicable [1)] Only in connection with *Cold Reload*, i.e., with stopped communication module.		

Table 45: safe**ethernet** Reload after Changes

4.13 Cross-Project Communication

Cross-project safety-related communication is used to connect resources from various projects.

The connection between the two projects is established via proxy resource. A proxy resource substitutes a resource from the other project.



1	Project A	5	Project B
2	Resource A	6	Resource B
3	Proxy resource B	7	Proxy resource A
4	Configuration file		

Figure 23: safe**ethernet** Connection Between Resource A in Project A and Resource B in Project B

In project A, the safe**ethernet** connection is configured and the configuration file is created and archived.

In project B, the configuration file is restored. Proxy resource A is automatically created with the data of resource A from project A.

4.13.1 Configuration in SILworX

An example is given to illustrate the basic procedure. The names used for the projects, configurations and resources are only examples.

For clarity, configuration A and configuration B are created in both SILworX projects.

Project A

├ Configuration A
│ └ Resource A
├ Configuration B
│ └ Resource B (as proxy)
└ Global Variables

Project B

├ Configuration B
│ └ Resource B
├ Configuration A
│ └ Resource A (as proxy)
└ Global variables (Global variables can be created by restoring an archive or importing an Excel list.)

4.13.1.1 Creating Configuration B in Project A

Create a separate configuration B for proxy resource B in project A.

To create configuration B

1. Open project A in which configuration B should be created.
2. Right-click **Project A**, and then select **New**, **Configuration**.
   ☑ A new configuration (Configuration B) is created.

4.13.1.2 Creating Proxy Resource B in Project A

Proxy resource B serves as placeholder for a resource from an external project B and is used for configuring the process data exchange via safe**ethernet** .

To create Proxy Resource B

1. Right-click **Configuration B**, and select **New**, **Proxy Resource** SILworX.
   ☑ A new proxy resource (Proxy Resource B) is created.

To configure Proxy Resource B

1. Right-click Proxy Resource B, and select **Properties**.
2. Enter a unique name in the **Name** field.
   Use the name of resource B in project B for proxy resource B in project A.
3. Read the **System ID** from project B and enter it in proxy resource B.
4. Click **OK** to confirm.

To open the structure tree for Proxy Resource B

1. Right-click **Hardware** and select **Edit**.
2. Select the resource type used in project B:
   - **HIMatrix 03 proxy**
   - HIMatrix proxy
   - HIMax system proxy
   - H41X system proxy
   - H51X system proxy
3. Click **OK** to confirm. The Hardware Editor for proxy resource B appears.
4. For proxy HIMatrix 03, successively double-click the CPU and COM module to be used for establishing the redundant connection to proxy resource B.

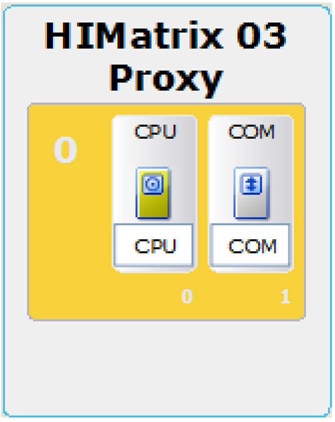


Figure 24: HIMatrix Proxy Resource

5. Enter the IP addresses and click **Save**.
6. Repeat these steps for every further proxy resource in project A.

4.13.1.3 Creating and Archiving Global Variables for safeethernet Connection

To create global variables for the safeethernet connection

1. Right-click **Project A**, and then select **New**, **Global Variables**.
   ☑ The Global Variables structure tree object is created at project level.
2. Right click **Global Variables**, and select **Edit** from the context menu to open the Variable Editor.
3. Right click a free space within the workspace of the Variable Editor and select **New Global Variable** from the context menu to create a new global variable.
4. Repeat these steps for each additional new global variable for the safe**ethernet** connection.
5. Additionally, create the global variables *Connection State*, *Quality Channel 1* (and possibly *Quality Channel 2* for redundant connections). Create each system variable twice, once from the perspective of resource A and once from the perspective of resource B.

To archive the global variables

TIP If at project level the *Global Variable* object already exists in project B, the SILworX function for importing or exporting contents as CSV can be used as an option. With the appropriate filter settings, the required global variables can be exported selectively, refer to the online help for more details.
Even with multiple proxy connections to various projects, export is more useful than archiving. In such a case, for each variable, enter a connection ID in the Additional Comment field.

1. In the structure tree, select **Project A**, **Global Variables**.
2. Select **Archive** from the context menu. The SILworX dialog box to archive an object appears.
3. In the dialog box, enter an archive name for the global variables object. The archive is saved with the ***.A3** file extension in the selected archive folder.
   ☑ The archived Global Variables object contains all the global variables created at project-level in project A.

4.13.1.4 Creating a Connection between Resource A and Proxy Resource B.

In the safe**ethernet** Editor, create a safe**ethernet** connection between resource A and proxy resource B.

To open the safeethernet Editor for Resource A

1. In the structure tree, select **Configuration A**, **Resource A**, safe**ethernet** .
2. Right-click safe**ethernet** , then select Edit.
   ☑ The new proxy resource B is created in the Object Panel.

To create the safeethernet connection to the proxy resource

1. Drag Proxy Resource B from the Object Panel onto a free space within the workspace of the safe**ethernet** Editor.
   Select an appropriate name for this connection, immediately.
2. Select proper Ethernet interfaces **IF Chx** of the resource and proxy resource.

4.13.1.5 Connecting Process Variables

Add the process variables in the editor of the safe**ethernet** connection.

To open the connection editor

The safe**ethernet** Editor of resource A is open.

1. Right-click the Proxy Resource B row and open the context menu.
2. Select Edit from the context menu to open the connection editor of the safe**ethernet** connection.
3. Select the Resource A<->Proxy Resource B tab.
4. In the Object Panel, select a **Global Variable** and drag it onto the **Resource A --> Resource B (Proxy)** area or onto the **Resource B (Proxy) --> Resource A** area depending on the selected transport direction.
5. Repeat this step for further variables.

4.13.1.6 Connecting System Variables

Connect the system variables Connection State, *Quality Channel 1* (and possibly *Quality Channel 2* for redundant connections) with global variables. For further information on the system variables, see Chapter 4.6.3.1.

To open the connection editor

The safe**ethernet** Editor of resource A is open.

1. Right-click the **Proxy Resource** row and open the context menu.
2. Select **Edit** from the context menu to open the connection editor of the safe**ethernet** connection.
3. Select the **System Variables** subtab in the **Resource A** tab.
4. In the Object Panel, select a suitable **Global Variable** for this system variable and drag it onto the **Global Variable** column.
5. Repeat this step for additional system variables.
6. Select the **System Variables** subtab in the **Resource B** tab.
7. In the Object Panel, select a suitable **Global Variable** for this system variable and drag it onto the **Global Variable** column.
8. Repeat this step for additional system variables.

4.13.1.7 Archiving the safe**ethernet** connection in Project A

The safeethernet connection configured in project A must be archived and then restored in project B.

To verify the safe**ethernet** connection

1. In the structure tree of **Project A**, select safe**ethernet** and open the context menu.
2. Select **Verification** from the context menu, and click **OK** to confirm.
3. Thoroughly verify the messages contained in the logbook and correct potential errors.

To archive the safeethernet connection

1. In the structure tree of **Project A**, select safe**ethernet** and open the context menu.
2. Select **Archive** from the context menu. The dialog box to archive an object appears.
3. In the dialog box, enter an archive name for the safeethernet object. The archive is saved with the ***.A3** file extension in the selected archive folder.
   ☑ All the safe**ethernet** connections contained in the safe**ethernet** object are now archived. safe**ethernet** connections can also be archived individually.
4. Close project A.

i The configuration of the safe**ethernet** connection must be recompiled with the user program of the resource and transferred to the controller. The new configuration can only be used for communicating with the controller upon completion of this step.

4.13.2 Configuration A in Project B

Create a separate Configuration A for proxy resource A in project B.

Project B is configured in SILworX like the first project. The resource from the first project is now the proxy resource.

4.13.2.1 Creating Proxy Resource A in Project B

Proxy resource A serves as placeholder for Resource A from an external project A and is used for exchanging process data via safeethernet.

To create a proxy resource

1. Open project B in which Proxy resource A should be created.
2. Right-click **Project B**, and then select **New**, **Configuration**.
   ☑ A new configuration (configuration A) is created.
3. Right-click **Configuration A**, and select **New**, **Proxy Resource** SILworX.
   ☑ A new proxy resource (proxy resource A) is created.

To configure a proxy resource

1. Right-click Proxy Resource A, and select **Properties**.
2. Enter a unique name in the **Name** field.
   Use the name of resource A in project A for proxy resource A in project B.
3. Read the **System ID** from project A and enter it in proxy resource A.
4. Click **OK** to confirm.

To open the structure tree for the proxy resource

1. Right-click **Hardware** and select **Edit**.
2. Select the resource type that is used in the first project:
   - H41X System proxy
   - H51X System proxy
   - HIMatrix 03 proxy
   - HIMatrix proxy
   - **HIMax system proxy**
3. Click **OK** to confirm. The Hardware Editor for the proxy resource appears.
4. Select **Generic Module** for HIMax system proxy and drag it to the base module on the proper slot, that is corresponding to the slot of the CPU / COM in the project A.

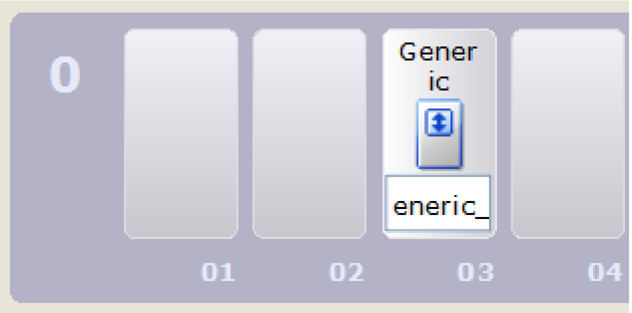


Figure 25: HIMax Proxy Resource

5. Double-click **Generic Module**, and enter the IP Address of the CPU and/or COM modules.
6. Click **Save**.
7. Repeat these steps for every further proxy resource in project B.

4.13.2.2 Creating Global Variables for safe**ethernet** Connection

The same global variables as in the first project must be created in project B.

TIP If at project level the Global Variable object already exists, the SILworX function for importing or exporting contents as CSV can be used as an option, see the online help.

To restore global variables in Project B

1. Right-click **Project**, and select **Restore** from the context menu.
   ☑ The SILworX dialog box to restore an object appears.
2. Open the archive folder and select the archived *Global Variables* object with the ***.A3** file extension that was previously created in project A.
   ☑ The restored global variables object contains all the global variables archived in project A.

4.13.2.3 Restoring the safe**ethernet** connection in Project B

To restore the safeethernet connection in Project B

1. Right-click **Project**, and select **Restore** from the context menu.
   ☑ The SILworX dialog box to restore an object appears.
2. Open the archive folder and select the archived safe**ethernet** object with the ***.A3** file extension that was previously created in project A.
   ☑ The restored safe**ethernet** object contains all the connections between resource A and proxy resource B in project B, including all assigned variables, process and system variables.

5 SNTP Protocol

The SNTP (Simple Network Time Protocol) is a simplified version of the NTP (Network Time Protocol).

The SNTP protocol is used by the SNTP server to synchronize the time of the SNTP clients via Ethernet.

HIMA systems can be configured and used as SNTP server and/or as SNTP client. The SNTP standard in accordance with RFC 2030 (SNTP V4) applies with the limitation that only the unicast mode is supported.

5.1 Equipment and System Requirements

Element	Description
Controller	HIMax HIQuad X HIMatrix
Activation	This function is activated by default in all HIMA systems.
Interface	Ethernet 10/100/1000BaseT

Table 46: Equipment and System Requirements for the SNTP Protocol

5.2 SNTP Client

To synchronize its time settings, the SNTP client only uses the SNTP server that is available and has the highest priority. If the priority is the same, the SNTP server that (randomly) first transmitted data to the SNTP client, is selected. This SNTP server is maintained until it is no longer available. Only then it is changed.

If the time difference is < 128 ms, the clock runs faster or slower by 0.5 ms per cycle until the time difference is compensated. If the time difference is ≥ 128 s, the clock is changed immediately.

For time synchronization, one SNTP client can be configured in each HIMA controller.

To create a new SNTP client

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**.
2. Right-click **Protocols**, then click **New**, **SNTP Client**.
   ☑ An SNTP server Info is added by default subordinate to the SNTP client.
3. Right-click the SNTP client, and click **Properties** and select the COM module.

The dialog box for the SNTP client contains the following parameters:

Element	Description
Type	SNTP client.
Name	Name for the SNTP client.
Module	Selection of the COM or processor module within which the protocol is processed.
Activate Max. µP Budget	Not taken into account by the operating system. Parameter was retained due to CRC and reload stability.
Max. µP Budget in [%]	Not taken into account by the operating system. Parameter was retained due to CRC and reload stability.
Description	Any unique description for the SNTP client.
Current SNTP version	The current SNTP version is displayed.

Element	Description
Reference stratum	The stratum of an SNTP client specifies the precision of its local time. The lowest the stratum, the more precise its local time. Zero means an unspecified or not available stratum (not valid). The SNTP server currently used by an SNTP client is the one that can be reached and has the highest priority. If the stratum of the current SNTP server is lower than the stratum of the SNTP client, the resource adopts the time of the current SNTP server. If the stratum of the current SNTP server is higher than the stratum of the SNTP client, the resource does not adopt the time of the current SNTP server. If the stratum of the current SNTP server is identical to the stratum of the SNTP client, two different cases result: ▪ If the SNTP client (resource) only operates as SNTP client, the resource adopts the time of the current SNTP server. ▪ If the SNTP client (resource) also operates as SNTP server, the resource adopts half the value of the time difference to the current SNTP server per SNTP client request (time adapts slowly). Range of values: 2…15 Default value: 15
Client Time Request Interval [s]	Time needed by the current SNTP server to perform time synchronization. The value set in the SNTP client for Client Time Request Interval must be greater than the timeout in the SNTP server. Range of values: 16…16384 s Default value: 16 s

Table 47: SNTP Client Properties

5.2.1 SNTP Server Info

The connection to the SNTP server (time server) is configured in the SNTP Server Info.

An SNTP server Info is subordinated to the SNTP client by default. A maximum of 4 SNTP Server Infos can be subordinated to an SNTP client.

i If several SNTP clients are configured in a HIMA system, only the (one) SNTP client whose active remote SNTP server has the highest priority may be used for time synchronization at any time.

To create a new SNTP Server Info

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**, **SNTP Client**.
2. Right-click **Protocols**, and then select **New**, **SNTP Server Info**.
   ☑ A new **SNTP Server Info** is created.

The dialog box for the SNTP Server Info contains the following parameters:

Element	Description
Type	SNTP Server Info
Name	Name for the SNTP server.
Description	Description for the SNTP server.
IP Address	IP address of the resource or PC in which the SNTP Server is configured. Default value: 0.0.0.0
SNTP Server Priority	Priority with which the SNTP client addresses this SNTP server. The SNTP servers configured for the SNTP client should have different priorities. Range of values: 0 (lowest priority) to 4294967295 (highest priority). Default value: 1
SNTP Server Timeout[s]	The timeout in the SNTP server must be set lower than the value for the *Time Request Interval* in the SNTP client. Range of values: 1…16384 s Default value: 1 s

Table 48: SNTP Server Info Properties

5.3 SNTP Server

The SNTP server on a HIMA system allows external systems to synchronize their date and time to the date and time of the HIMA system.

The SNTP server responds to SNTP requests when the system time is synchronized, otherwise SNTP requests are discarded.

The SNTP server of a HIMA system accepts the requests from an SNTP client (e.g., remote I/O) and sends its current time back to this SNTP client.

To create a new SNTP server

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**.
2. Right-click **Protocols**, and then select **New**, **SNTP Server**.
   ☑ A new SNTP server is created.
3. Right-click the SNTP server **Properties** and then select the **Module** used to connect the SNTP client.

The dialog box for the SNTP server contains the following parameters:

Element	Description
Type	SNTP Server
Name	Name of the SNTP server.
Module	Selection of the COM or processor module within which the protocol is processed.
Activate Max. µP Budget	Activated: Use the µP budget limit from the Max. µP Budget in [%] field. Deactivated: Do not use the µP budget limit for this protocol. Default value: Activated
Max. µP Budget in [%]	Maximum module's µP load that can be used for processing the protocol. Range of values: 1…100% Default value: 10%
Description	Description for the SNTP.
Current SNTP Version	The current SNTP version is displayed.
Stratum of Timeserver	The stratum of an SNTP server specifies the precision of its local time. The lowest the stratum, the more precise the local time. Zero means an unspecified or not available stratum (not valid). The value for the SNTP server stratum must be lower or equal to the stratum value of the requesting SNTP client. Otherwise, the SNTP client does not accept the SNTP server time. Range of values: 1…15 Default value: 14

Table 49: SNTP Server Properties

5.4 Configuration of Time Synchronization via SNTP

In the network structure shown, a HIMatrix is configured as SNTP server for time synchronization of the subordinate remote I/O. The HIMatrix is additionally configured as an SNTP client and gets the time synchronization from the network time server.

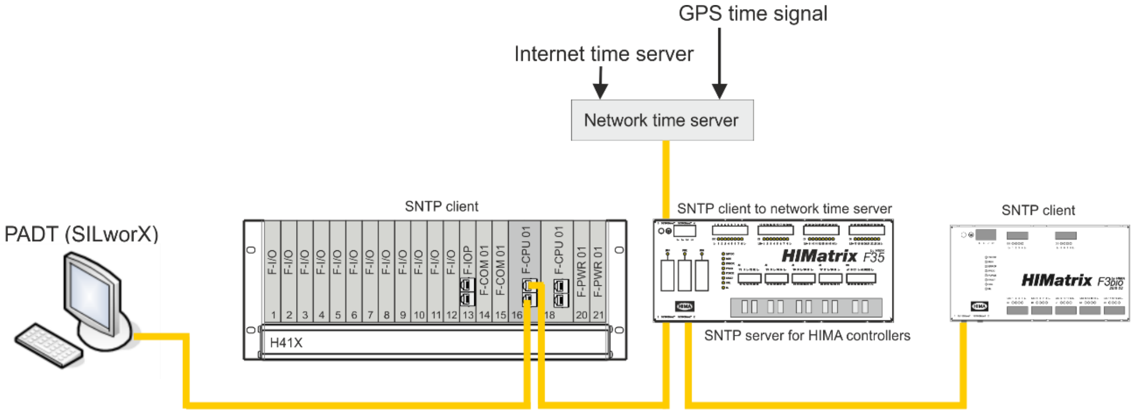


Figure 26: Time Synchronization of HIMA Systems via the SNTP Time Server

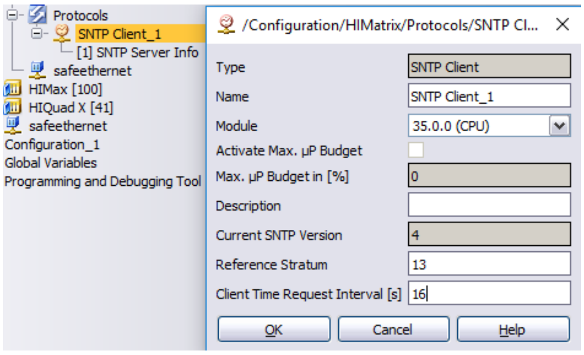
5.4.1 Creating an IP Connection to a Network Time Server

For time synchronization, one SNTP client can be configured in each resource.

To create a new SNTP client

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**.
2. Right-click **Protocols**, then click **New**, **SNTP Client**.
3. Right-click the SNTP client, click **Properties** and select the COM module connected with the PC.

   ☑ Standard reference stratum ‘15‘ can be retained if the COM module does not additionally function as an SNTP server.
   Rule: Value for the SNTP server stratum ≤ stratum value of the requesting SNTP client. Otherwise, the SNTP client does not accept the SNTP server time.

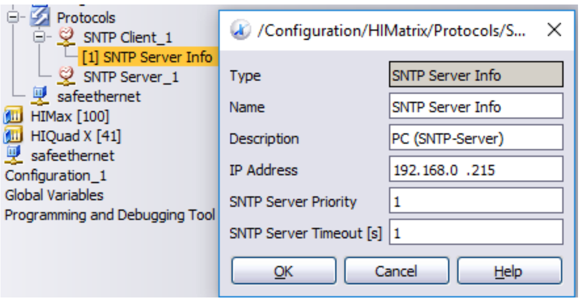


Type	SNTP Client
Name	SNTP Client_1
Module	35.0.0 (CPU)
Activate Max. µP Budget	□
Max. µP Budget in [%]	0
Description	
Current SNTP Version	4
Reference Stratum	13
Client Time Request Interval [s]	16

Figure 27: Configuring the SNTP Client for Time Synchronization

To configure the SNTP Server Info subordinate to the SNTP client

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**, **SNTP Client**.
2. Right-click **SNTP Server Info** and select **Properties** from the context menu.
3. In **Properties**, select the **IP Address** of the SNTP server (PC).



Type: SNTP Server Info
Name: SNTP Server Info
Description: PC (SNTP-Server)
IP Address: 192.168.0 .215
SNTP Server Priority: 1
SNTP Server Timeout [s]: 1

Figure 28: Configuring the IP Connection to the SNTP Server (PC)

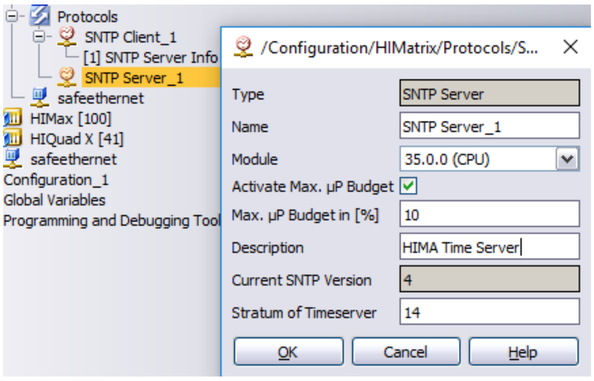
5.4.2 SNTP Time Synchronization of a Remote I/O by a HIMA Resource

The remote I/O is synchronized via SNTP. To do so, an SNTP server must be created in the superordinate HIMA resource, whose time is synchronized via an Internet time server or a GPS clock.

To create an SNTP server for SNTP time synchronization

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**.
2. Right-click **Protocols**, and then select **New**, **SNTP Server**.

   ☑ A new SNTP server is created.
3. Right-click the SNTP server, click **Properties** and select the **Module** connected with the remote I/O. This has to be the identical module used for the safeethernet connection to the remote I/O. The stratum value must not exceed 14.



Type: SNTP Server
Name: SNTP Server_1
Module: 35.0.0 (CPU)
Activate Max. µP Budget: ☑
Max. µP Budget in [%]: 10
Description: HIMA Time Server
Current SNTP Version: 4
Stratum of Timeserver: 14

Figure 29: Creating an SNTP server for SNTP time synchronization

6 HART

The HART (Highway Addressable Remote Transducer) protocol allows digital fieldbus communication during which the HART signal is superimposed onto the (4…20 mA) analog current signal. The data rate of the HART protocol is 1200 bit/s. The HART signal is used to transfer measuring and device data of connected HART-capable sensors or actuators.

The X-HART 32 01 module establishes digital HART fieldbus communication between a maximum of 32 HART-capable field devices and the HIMax system.

The HART signal is used to transfer measuring and device data of connected HART-capable sensors or actuators. Within the system, the X-HART 32 01 module transmits the measuring and device data to the assigned X-COM 01 communication module. The X-COM 01 transfers the measuring and device data via the HART-IP protocol to an asset management system or a HART OPC Server.

To reduce security risks, HIMA recommends preventing unauthorized changes to the HART field devices be implementing write-protection.

6.1 System Requirements

Equipment and system requirements for HART protocol:

Element	Description
Controller	HIMax with X-COM module and X-HART module.
X-CPU module	The Ethernet interfaces are not used for HART-IP. For setting the parameters on a per module basis: CPU operating system as of V5. For setting the parameters on a per channel basis: CPU operating system as of V11.
X-COM module	Ethernet 10/100BaseT are used for HART-IP. COM operating system as of V7.24.
X-HART 32 01	For setting the parameters on a per module basis: I/O operating system as of V5. For setting the parameters on a per channel basis: I/O operating system as of V7.48.
Analog module	Analog input or output module.
Activation	The HART-IP protocol is activated by default for HIMax systems.

Table 50: Equipment and System Requirements for the HART Protocol

6.1.1 HART Protocol Features

The HART hast the characteristics specified in the following table.

Properties	Description
Safety-related	No
Transfer rate	HART fieldbus communication: 1200 Bit/s. HART-IP via Ethernet: 100 Mbit/s full duplex.
Transport path	HART fieldbus communication 32-channel HART interface of the X-HART module. HART-IP via Ethernet Ethernet interfaces of the X-COM module. The Ethernet interfaces in use can simultaneously be employed for other protocols.
Max. number of X-HART modules	100 for each HIMax system. Based on dimensioning, refer to the system manual (HI 801 001 E).

Properties	Description
Max. number of I/O points	3200 for each HIMax system. Depending on the module type; in this case, for 100 analog modules with 32 inputs each.
Max. number of HART-IP protocol instances	1 for each X-COM module. 2 for each HIMax system (with 2 X-COM modules).
Max. number of HART-IP sessions via UDP	2 on each X-COM module.
Max. number of HART-IP sessions via TCP	2 on each X-COM module.

Table 51: HART Protocol Features

6.2 HART Communication for Safety-Related Applications

HART communication enables read and write access to the transmitters, including the option to change the transmitter configuration.

Since the HART-IP protocol was not developed in accordance with the requirements of IEC 61508, the data supplied via HART must not be used as a reliable source for safety-related functions.

However, the information provided via the HART protocol can be used within asset management systems, e.g., for diagnostics.

The safety-related analog values are processed in the HIMA safety-related controller and the HART data is processed in the asset management system (layers of protection in accordance with IEC 61511).

6.2.1 Safety Function

The safety function of HART communication via the HIMax system includes the following points:

- HART Deactivation: If the module is shut down, the HART channels are safely deactivated in accordance with SIL 3.
- HART Filtering: HART access to transmitters or sensors is locked in accordance with SIL 3.
- HART communication influences the analog metrological accuracy by 1 %. Further repercussions on the analog HIMax modules are excluded.
- HART parameter setting: The parameters of the X-HART module can be set on a per module basis for all 32 channels, or on a per channel basis for each individual channel.

⚠ WARNING

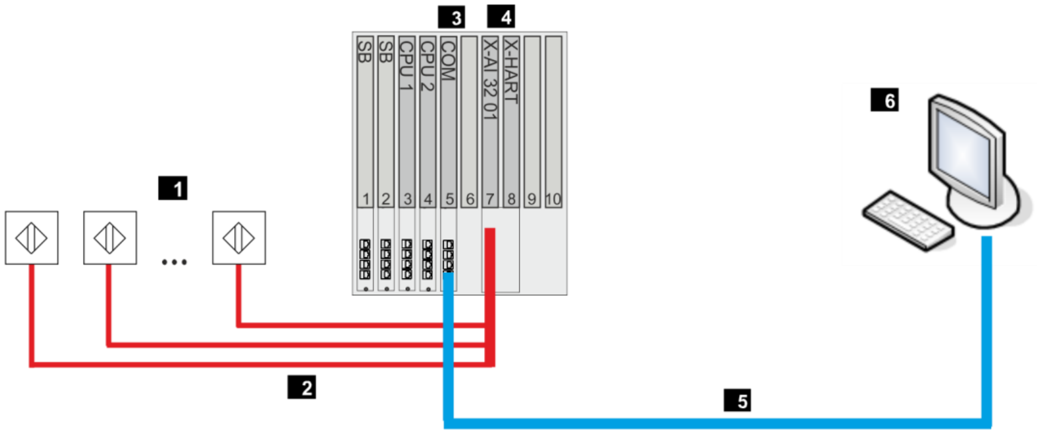
Manipulation of analog sensors and actuators!

If the HART filtering function is deactivated on the HART module, the corresponding analog sensor or actuator can be reprogrammed.

The operator is responsible for ensuring that the HART field devices used for the HART protocol are sufficiently protected against manipulations (e.g., from hackers).
The type and extent of the measures must be agreed upon together with the responsible test authority, see also Chapter 2.5.

6.3 Configuring a HART-IP Protocol Instance

This chapter provides an overview of the configuration of HART-IP instances and the interaction between HART field device, HIMax controller, engineering too and a HART OPC Server or FDT/DTM asset management system.



1 HART field device

2 HART via 4…20 mA

3 X-COM 01

4 X-HART 32 01 and analog I/O module

5 HART-IP (Ethernet)

6 HART OPC Server or FDT/DTM asset management system

Figure 30: Structure of the HART-IP Installation

6.3.1 HART OPC Server or FDT/DTM Asset Management System

A HART OPC Server or an FDT/DTM asset management system can be used to configure and monitor the HART field devices.

Supported asset management systems include:

- PACTWARE.
- FIELDCARE.
- Honeywell FDM.
- Yokogawa FieldMate.
- Other systems on request.

A suitable HART OPC Server can be obtained from the HART Foundation.

i The two device drivers *CommDTM* and *DeviceDTM* for the HIMax system can be obtained from HIMA.

6.3.2 HART Field Devices

The HART field devices must be connected to the analog input or output modules (e.g., X-AI 32 01, X-AO 16 01). If short-circuits or open-circuits occur, no HART communication is possible.

i HIMA recommends setting the polling address of all connected HART field devices to *zero*. The same sub-device address for each connected device is possible, since the HIMax system only provides one field device per HART channel (no multi-drop operation). The *search* for connected HART field devices starts with polling address *zero*. A device with an address of *zero* will be "found" fastest after a power-on.

6.3.3 Configuring the X-HART Module, X-COM Module and analog I/O Modules

The X-COM communication module and the assigned X-HART module form an I/O system in accordance with the HART specification.

The X-HART 32 01 module is a communication module with 32 channels. It is combined with an analog input or output module and connected through a connector board. The corresponding connector board occupies 2 slots for mono applications and 3 slots for redundant applications.

The HIMax modules are configured in the Hardware Editor of the SILworX programming tool.

To add the required modules in the Hardware Editor

1. In the structure tree, select **Configuration**, **Resource**, **Hardware**.
2. Right-click and select **Edit** from the context menu to open the Hardware Editor.
3. From the Object Panel, drag the modules **X-COM 01**, **X-AI 32 01** and **X-HART 32 01** to a suitable position on the base rack.

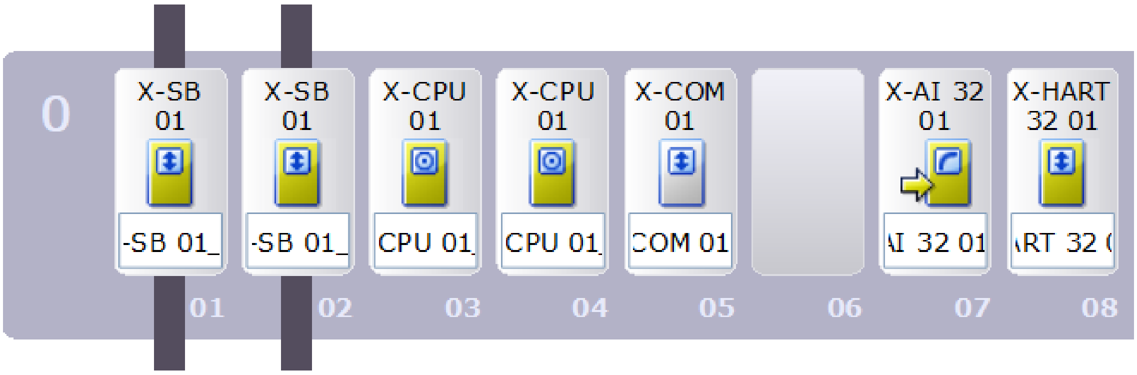


Figure 31: HART-IP Configuration in the SILworX Hardware Editor

6.3.3.1 X-AI 32 01 Analog Input Module

The X-AI 32 01 module is configured in the detail view of the Hardware Editor.

To open the detail view of the X-AI 32 01 module in the Hardware Editor

1. Right-click **X-AI 32 01** and select **Detail View** from the context menu.
   ☑ The detail view contains its own workspace, which sometimes includes additional tabs for configuring object parameters and analog inputs.

i For further information on how to configure the X-AI 32 01, refer to the corresponding manual (HI 801 021 E).

6.3.3.2 X-HART Module

The X-HART module is configured in the detail view of the Hardware Editor.

To open the detail view of the X-HART module in the Hardware Editor

1. Right-click **X-HART** and select **Detail View** from the context menu.
   ☑ The detail view contains its own workspace, which sometimes includes additional tabs for configuring object parameters.

Observe the following points when configuring the module:

- Configure the corresponding analog input or output module (e.g., X-AI 32 01).
- To diagnose the X-HART module and HART channels, the system parameters can be combined with global variables and evaluated within the user program.
- In case of analog output modules with redundant wiring, the Module Status parameter must be additionally taken into account, see the module-specific X-AO 16 01 manual (HI 801 111 E).

i For further information on how to configure the X-HART module, refer to the corresponding manual (HI 801 307 E).

6.3.3.3 Configuring the X-COM Module in the Detail View

The X-COM 01 module is configured in the detail view of the Hardware Editor.

To open the detail view of the X-COM 01 module in the Hardware Editor

1. Right-click **X-COM 01** and select **Detail View** from the context menu.
   ☑ The detail view contains its own workspace, which sometimes includes additional tabs for configuring object parameters.
2. Enter the **IP-Address** for connecting to the HART OPC Server or FDT/DTM asset management system.

i For further information on how to configure the X-COM 01, refer to the corresponding manual (HI 801 011 E).

6.3.4 Configuring the HART-IP Protocol Instance

The protocol and the required HIMax modules are configured in the SILworX programming tool.

To create a HART-IP protocol instance

1. In the structure tree, select **Configuration**, **Resource**, **Protocols**.
2. Select **New**, **HART-IP Protocol** from the context menu of protocols to add a new HART-IP protocol.
3. Right-click the HART protocol and select **Properties** of the **X-Com Module**.
   The default settings may be retained for the first configuration.

6.3.4.1 Properties

The Properties dialog box for the HART-IP protocol contains the following parameters:

Element	Description
Name	Name for the HART-IP protocol.
Module	Selection of the COM module within which the HART-IP protocol is processed.
Activate Max. µP Budget	Activated: Use the µP budget limit from the *Max. µP Budget in [%]* field. Deactivated: Do not use the µP budget limit for this protocol. Default value: Activated
Max. µP Budget in [%]	Maximum µP budget of the module that can be used for processing the protocols. Range of values: 1…100% Default value: 30%
Polling address	Polling address of X-COM Range of values: 0…63 Default value: 0
Use Standard HART TCP Port (5094)	Activated The TCP connection is enabled. Deactivated The TCP connection is disabled. Default value: Activated, TCP port 5094 is used.
Use Second HART TCP Port	Activated The TCP connection is enabled. Deactivated The TCP connection is disabled. Default value: Deactivated
Second HART TCP port	The second port number can be used as an alternative or in addition to the standard port. Range of values: 1…65535 Default value: 20004
Use Standard HART UDP Port (5094)	Activated The UDP connection is enabled. Deactivated The UDP connection is disabled. Default value: Activated, UDP port 5094 is used.
UseUse Second HART UDP Port	Activated The UDP connection is enabled. Deactivated The UDP connection is disabled. Default value: Deactivated
Second HART UDP port	The second port number can be used as an alternative or in addition to the standard port. Range of values: 1…65535 Default value: 20004

Table 52: HART-IP Protocol Properties

6.4 Online View of the X-COM Module

The settings for the HART protocol can be controlled and verified from within the online view of the X-COM module. Details about the current status of the field devices and X-COM module are displayed.

To open the online view of the Hardware Editor for the HART protocol:

1. Right-click the **Hardware** structure tree element and select **Online** from the context menu.
2. In the **System Login** window, enter the access data to open the online view for the hardware.
3. Double-click the **X-COM Module** and select the **HART Protocol** structure tree node.

6.4.1 View Box (HART Protocol)

The view box displays the following values of the selected HART protocol.

Element	Description
Name	Name for the HART-IP protocol.
Planned µP Budget [%]	Value displayed for the planned maximum µP budget of the X-COM module, which may be produced during the protocol's processing.
Current µP Budget [%]	Value displayed for the current µP budget of the X-COM module, which is being produced during the protocol's processing.
Polling Address	The polling address of X-COM is displayed. Range of values: 0…63
Unique COM Address	The 5-byte address of the X-COM (unique address) is displayed.
Standard HART TCP Port Number	The standard TCP port used for HART-IP is displayed online.
Second HART TCP Port Number	The TCP port additionally or alternatively used for HART-IP is displayed online.
Standard HART UDP Port Number	The standard UDP port used for HART-IP is displayed online.
Second HART UDP Port Number	The UDP port additionally or alternatively used for HART-IP is displayed online.
Number of HART Devices	The number of HART devices currently detected as connected is displayed. The X-COM 01, which is also a part of the HART configuration, is not included in this number.
Number of X-HART Modules	The number of X-HART 32 01 modules (I/O cards) detected and belonging to this X-COM 01 module is displayed.
Status Device Lock	It displays the device interlock status. The device interlock (interlock for the HART I/O subsystem) is triggered by the host through HART command 71. Range of values: See HCF_SPEC-183 (Common Tables) Table 25 Zero – The device is not locked Not equal to zero – Lock device status code Default value: 0
Device Lock through Host with IP	With the device interlock (device interlock status is not zero), this field displays the IP address of the host that triggered the interlock (i.e., which sent HART command 71). Range of values: IP address Default value: 0

Table 53: Online View of the HART Protocol

6.4.2 Online View of the Device List

To update the device list

1. In the structure tree, select **HART Protocol**, **Device List**.
2. Right-click and select **Update Device List**.

The **Device List** view box displays the following values.

Element	Description
Device Index	The device index is displayed online Range of values: 0…65535 (decimal, 2 bytes)
I/O Card Number	The I/O card number to which the device is connected is displayed online. Range of values: 0…249 (decimal, 1 byte) for connected devices. Value: 251 (None) for the X-COM itself.
Channel number	The channel number to which the device is connected is displayed online. Range of values: 1…31 (decimal, 1 byte) for connected devices, see Chapter 6.4.2.1. Range of values: 251 (None) for the X-COM itself.
Manufacturer ID	The ID of the device manufacturer is displayed online. Range of values: 0x00…0xFFFF (hexadecimal, 2 bytes)
Expanded Device Type Code	The expanded device type code of the device is displayed online. Range of values: 0x00…0xFFFF (hexadecimal 2 bytes)
Device ID	The ID of the device is displayed online. Range of values: 0x00 0x00 0x00…0xFF 0xFF 0xFF (hexadecimal 3 bytes)
HART Version	The HART version (Universal Command Revision Level) of the device is displayed online. Range of values 0…255 (decimal 1 byte)
Long Tag	The device long tag is displayed online. Range of values 32 characters (Latin-1)
Rack.Slot I/O Card	The I/O card slot (Rack.Slot) is displayed online. Format: Rack.Slot Range of values (Rack): 0…15 Range of values (Slot): 0…15
Telegram Counter STX	The telegram counter for the device's commands (Stx) is displayed online. Range of values 0…65535 (decimal 2 bytes revolving)
Telegram Counter ACK	The telegram counter for the device's acknowledgements (Ack) is displayed online. Range of values 0…65535 (decimal 2 bytes revolving)
Telegram Counter BACK	The telegram counter for the device's burst acknowledgements (Back) is displayed online. Range of values 0…65535 (decimal 2 bytes revolving)

Table 54: Online View of the Device List

6.4.2.1 HART Field Device Addressing

Channel number (X-HART 32 01 front plate)	Channel address (decimal)	Channel address (hexadecimal)
1…32	0…31	0x00…0x1f

Table 55: HART Field Device Addressing

The channel numbers displayed in the X-COM 01 online view correspond to the channel numbers specified on the front plate of the X-HART 32 01 module (channel count starting with 1).
If a connected HART field device is addressed, the following applies:
Channel address (channel number upon command 77) = channel number - 1.

Example:
Channel number = 15
The field device is addressed with channel address = 14 (0x0e).

7 General

This chapter describes parameters that are relevant for all communication protocols.

7.1 Maximum Communication Time Slice

The maximum communication time slice is the time period in milliseconds (ms) per CPU cycle assigned to the processor module for processing the communication tasks. Even if the protocol processing could not be completed within one communication time slice, the CPU still executes the safety-relevant monitoring for all protocols within one CPU cycle.

i If not all upcoming communication tasks can be processed within one CPU cycle, the whole communication data is transferred over multiple CPU cycles. The number of communication time slices is then greater than 1.

For calculating the maximum response time, the number of communication time slices must be equal to 1.

7.1.1 Determining the Maximum Duration of the Communication Time Slice

For a first estimate of the maximum duration of the communication time slice, the sum of the following times must be entered in the M*ax. Com. Time Slice [ms]* system parameter located in the properties of the resource.

- For each COM module: 3 ms.
- For each redundant safe**ethernet** connection: 1 ms.
- For non-redundant safe**ethernet** connection: 0.5 ms.
- For each kilobyte user data of non-safety-related protocols, e.g., Modbus: 1 ms.

HIMA recommends comparing the value estimated for *Max. Com. Time Slice [ms]* with the value displayed in the Control Panel and, if necessary, correcting it in the properties of the resource. This can be done during an FAT (factory acceptance test) or SAT (site acceptance test).

To determine the actual duration of the maximum communication time slice

1. Operate the HIMA system under full load (FAT, SAT):
   All communication protocols are in operation (safeethernet and standard protocols).
2. Open the **Control Panel** and select the **Com. Time Slice** structure tree folder.
3. Read the vale displayed for *Maximum Com. Time Slice Duration per Cycle [ms]*.
4. Read the value displayed for *Maximum Number of Required Com. Time Slice Cycles*.

The duration of the communication time slice must be set so that, when using the communication time slice, the CPU cycle cannot exceed the watchdog time specified by the process.

7.2 Load Limitation

A computing time budget expressed in % (*µP budget*) can be defined for each communication protocol. It allows the available computing time to be distributed among the configured protocols. The sum of the computing time budgets configured for all communication protocols on a CPU or COM module may not exceed 100%.

The defined computing time budgets of the individual communication protocols are monitored. If a communication protocol has already achieved or exceeded its budget and no reserve computing time is available, the communication protocol cannot be processed.

If sufficient additional computing time is available, it is used to process the communication protocol that has already achieved or exceeded its budget. It can therefore happen that a communication protocol uses more computing time budget than has been allocated to it.

It is possible that more than 100% computing time budget is displayed online. This is not a fault; the computing time budget exceeding 100% indicates the additional computing time used.

i The additional computing time budget is not a guarantee for a certain communication protocol and can be revoked from the system at any time.

7.3 Configuring the Function Blocks

The fieldbus protocols and the corresponding function blocks operate on the COM module of the HIMA system. In the SILworX structure tree, these function blocks must therefore be created as child element of **Configuration**, **Resource**, **Protocols**.

To control the function blocks on the COM module, function blocks can be created in the SILworX user program (see Chapter 7.3.1). These can be used as standard function blocks.

Shared variables are used to connect the function blocks in the SILworX user program to the corresponding function blocks in the SILworX structure tree. These must have been previously created in the Global Variable Editor.

7.3.1 Purchasing Function Block Libraries

The function block libraries for PROFIBUS DP and TCP Send/Receive must be added to the project using the *Restore* function (context menu of the project).

The function block library is available from HIMA support upon request.

7.3.2 Configuring the Function Blocks in the User Program

Drag the required function blocks onto the user program. The inputs and outputs must be configured as described for the individual function block.

Upper part of the function block

The upper part of the function block corresponds to the user interface used by the user program to control the function block.

The variables used in the user program are connected at this level. The prefix A means Application.

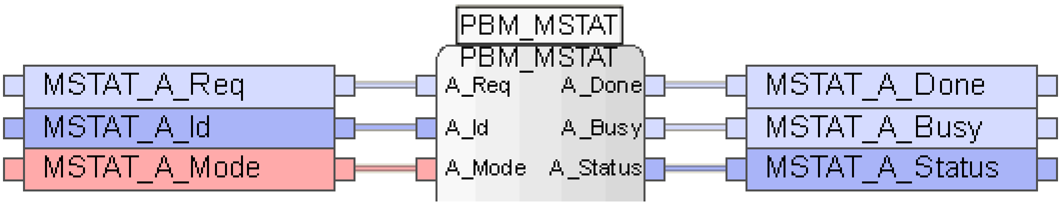


Figure 32: PNM_MSTST Function Block (Upper Part)

Lower part of the function block

The lower part of the function block represents the connection to the function block (in the SILworX structure tree).

The variables that must be connected to the function block located in SILworX structure tree are connected here. The prefix F means Field.

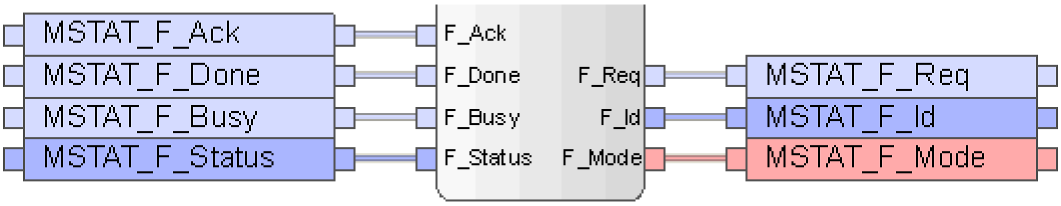


Figure 33: PNM_MSTST Function Block (Lower Part)

7.3.3 Configuring the Function Blocks in the SILworX Structure Tree

To create the function block in the SILworX structure tree

1. In the structure tree, open **Configuration**, **Resource**, **Protocols**, e.g., **PROFIBUS Master**.
2. Right-click **Function Blocks** and select **New**.
3. In the SILworX structure tree, select the suitable function block.

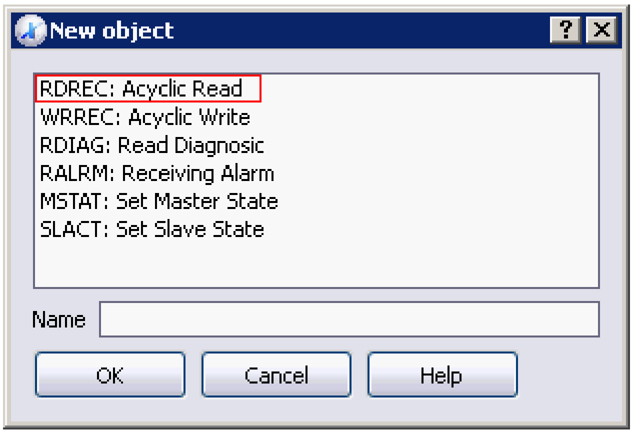
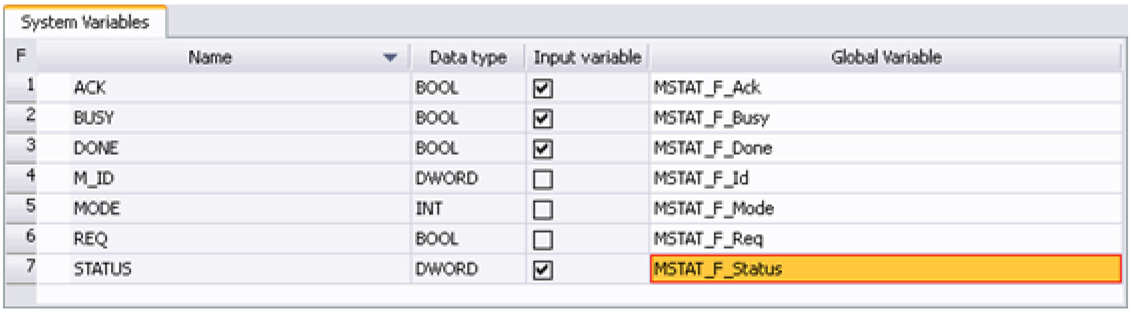


Figure 34: Selecting Function Blocks

The inputs of the function block (checkmark in the Input Variables column) must be connected to the same variables that are connected in the user program to the *F_Outputs* of the function block.

The outputs of the function block (no checkmark in the Input Variables column) must be connected to the same variables that are connected in the user program to the *F_Inputs* of the function block.



F	Name	Data type	Input variable	Global Variable
1	ACK	BOOL	☑	MSTAT_F_Ack
2	BUSY	BOOL	☑	MSTAT_F_Busy
3	DONE	BOOL	☑	MSTAT_F_Done
4	M_ID	DWORD	□	MSTAT_F_Id
5	MODE	INT	□	MSTAT_F_Mode
6	REQ	BOOL	□	MSTAT_F_Req
7	STATUS	DWORD	☑	MSTAT_F_Status

Figure 35: System Variables of the MSTAT Function Block

Appendix

Glossary

Term	Description
ARP	Address resolution protocol, network protocol for assigning the network addresses to hardware addresses.
Bit variable	Variable that is addressed bit by bit.
CENELEC	Comité Européen de Normalisation Électrotechnique (European Commitee for Electrotechnical Standardization).
COM	Communication module.
Connector board	Connector board for the HIMax module.
CPU	Processor module.
CRC	Cyclic redundancy check.
Data view	The global variables for output and output data are assigned to a data view to allow access to Modbus sources.
EN	European standard.
Export area	The export area is the process data volume that is written to by the system (a user program, hardware input or another protocol) and is read by the Modbus master.
FB	Fieldbus.
FBD	Function block diagrams.
ICMP	Internet control message protocol, network protocol for status or error messages.
IEC	International electrotechnical commission.
Import area	Process data volume that is written to by the Modbus master and can be used as input data for the system (in a user program, hardware output or another protocol).
Interference-free	Supposing that two input circuits are connected to the same source (e.g., a transmitter). An input circuit is termed "interference-free" if it does not distort the signals of the other input circuit.
KE	Communication end point.
MAC address	Media access control address, hardware address of one network connection.
NSIP	Not safety-related protocol.
PADT	Programming and debugging tool (acc. to IEC 61131-3), PC with SILworX.
PE	Protective ground.
PELV	Protective extra low voltage.
PES	Programmable electronic system.
R	Read.
R/W	Read/Write.
Rack ID	Base rack identification (number).
Register variable	Variable that is addressed word by word.
SB	System bus.
SFF	Safe failure fraction, i.e., portion of faults that can be safely controlled.
SIF	Safety-instrumented function.
SIL	Safety integrity level (in accordance with IEC 61508).
SILworX	Programming tool for HIMax, HIQuad X und HIMatrix.
SIP	Safety-instrumented protocol.
SNTP	Simple network time protocol (RFC 1769).
SRS	System.Rack.Slot.
SW	Software.
TMO	Timeout.
W	Write.
WD	Watchdog.
WDT	Watchdog time.

Index of Figures

Index of Tables

Index

MANUAL
Communication

HI 801 101 E

For further information, please contact:

HIMA Paul Hildebrandt GmbH
Albert-Bassermann-Str. 28
68782 Brühl, Germany

Phone	+49 6202 709-0
Fax	+49 6202 709-107
E-mail	info@hima.com

Learn more about HIMA solutions online:

🌐 www.hima.com/en/

HIMA SMART SAFETY.

www.hima.com