Manual

HIMatrix® F

Safety Manual
for Railway Applications

Contact

HIMA contact details:

HIMA Paul Hildebrandt GmbH

Postfach 1261

68777 Brühl

Phone: +49 6202 709-0

Fax: +49 6202 709-107

E-mail: info@hima.com

Document designation	Description
HI 800 436 D, Rev. 3.03 (1747)	German original document
HI 800 437 E, Rev. 3.03.00 (1806)	English translation of the German original document

Table of Contents

1 Introduction

This manual contains information on how to operate the HIMatrix safety-related automation devices in the intended manner.

The following requirements must be met to safely install and start up the HIMatrix automation devices, 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 devices.
- De-activation or bypassing of safety functions.
- Failure to comply with the instructions detailed in this manual.

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

- They are only used for the intended applications.
- They are only operated under the specified environmental conditions.
- They are only operated in connection with the approved external devices.

To provide a clearer exposition, this manual does not specify all details of all versions of the HIMatrix automation devices. Refer to the corresponding manuals for further details.

This safety manual represents the "Original instructions" as of Machinery Directive (Directive 2006/42/EC).

The "Original documentation" for the HIMA system is written in German language. The statements made in the German documentation shall apply.

1.1 Structure and Use of the Document

This safety manual examines the following topics:

- Intended Use
- Safety concept
- Central functions
- Inputs
- Outputs
- Software
- Safety-related aspects of the user program
- Configuring communication
- Appendix:
  - Glossary
  - Indexes

i This manual usually refers to compact controllers and remote I/Os as *devices*, and to the plug-in cards of a modular controller as *modules*.
Modules is also the term used in SILworX.

1.2 Target Audience

This document is aimed at the planners, design engineers and programmers of automation systems as well as the persons authorized to start up, operate and maintain the devices and systems concerned. 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 particularly marked. When the cursor hovers over a hyperlink, it changes its shape. Click the hyperlink to jump to the corresponding position.

Safety notices and operating tips are particularly 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 situations which, if not avoided, could result in minor or modest 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.

2 Usage Notes

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

2.1 Intended Use

This chapter describes the requirements for using HIMatrix systems.

2.1.1 Scope

The safety-related HIMatrix controllers can be used up to safety integrity level SIL 4 in accordance with EN 50126, EN 50128 and EN 50129.

The HIMatrix systems are certified for use in process controllers, protective systems, burner controllers, and machine controllers.

2.1.1.1 De-Energize to Trip Principle

The automation devices are designed in accordance with the de-energize to trip principle.

A system that operates in accordance with the de-energize to trip principle does not require power to perform its safety function.

Thus, if faults occur, the de-energized state is adopted as the safe state for inputs and outputs.

2.1.1.2 Energize to Trip Principle

The HIMatrix controllers can also be used in applications that operate in accordance with the energize to trip principle.

A system operating in accordance with the energize to trip principle requires power (such as electrical or pneumatic power) to perform its safety function.

When designing the controller system, the requirements specified in the application standards must be taken into account. For instance, line diagnosis for inputs and outputs or a message reporting a triggered safety function may be required.

2.1.2 Non-Intended Use

The transfer of safety-relevant data through public networks like the Internet is permitted if additional security measures such as VPN tunnel or firewall have been implemented to increase security.

No safety-related communication can be ensured with fieldbus interfaces.

2.2 Tasks of Operators and Machine and System Manufacturers

Operators as well as machine and system manufacturers are responsible for ensuring that HIMatrix systems are safely operated in automated systems and plants.

Machine and system manufacturers must sufficiently validate that the HIMatrix systems were properly programmed.

2.2.1 Connection of Communication Partners

Only devices with electrically protective separation may be connected to the communication interfaces.

2.2.2 Use of Safety-Related Communication

When implementing safety-related communications between various devices, ensure that the overall response time does not exceed the process safety time. All calculations must be performed in accordance with the rules given in Chapter 9.

2.3 ESD Protective Measures

Only personnel with knowledge of ESD protective measures may modify or extend the system or replace a module.

NOTICE

Electrostatic discharge can damage the electronic components within the controllers!

- **When performing the work, make sure that the workspace is free of static, and wear a grounding strap.**
- **If not used, ensure that the modules are protected from electrostatic discharge, e.g., by storing them in their packaging.**

Modifications or extensions to the system wiring must be performed by personnel with knowledge of ESD protective measures.

2.4 Additional System Documentation

In addition to this manual, the following documents for configuring HIMatrix systems are also available:

Manual	Content	Document no.
HIMatrix safety manual	Safety functions of the HIMatrix system.	HI 800 023 E
HIMatrix system manual	Description of the HIMatrix systems	HI 800 141 E
Certificates	Test results	-
Revision list	Operating system versions certified by the TÜV	-
Manuals for the components	Description of the individual components	
Communication manual	Description of the communication protocols, ComUserTask and their configuration in SILworX.	HI 801 101 E
SILworX online help	Instructions on how to use SILworX.	-
SILworX first steps manual	Introduction to the use of SILworX for engineering, start-up, testing and operation.	HI 801 103 E

Table 1 HIMatrix System Documentation

The current documents can be obtained upon request by sending an e-mail to: documentation@hima.com. The revision index in the footer can be used to compare the manuals in use with the available edition and determine if they are up-to-date.

3 Safety Concept for Using the PES

This chapter contains important general information on the functional safety of HIMatrix systems.

- Safety and availability
- Time parameters important for safety
- Safety requirements
- Certification
- Test conditions
- Additional test conditions for railway applications

3.1 Safety and Availability

The HIMatrix systems are certified for use in process controllers, protective systems, burner controllers, and machine controllers.

No imminent risk results from the HIMatrix systems.

⚠ WARNING

Possible physical injury caused by safety-related automation systems improperly connected or programmed.

Check all connections and test the entire system before start-up!

3.1.1 Calculating the HR Values

The HR values for the HIMatrix systems have been calculated in accordance with IEC 61508.

The HR values are provided by HIMA upon request.

The safety functions, consisting of a safety-related loop (input, processing unit, output and safety communication among HIMA systems), meet the requirements described above in all combinations. The controllers, remote I/Os and F60 modules meet these requirements.

3.1.2 Self-Test and Fault Diagnosis

The operating system of the controllers executes comprehensive self-tests at start-up and during operation. The following components are tested:

- Processors
- Memory areas (RAM, NVRAM)
- Watchdog
- Individual I/O channels

If faults are detected during the tests, the operating system switches off the defective device, module or faulty I/O channel.

In non-redundant systems, this means that sub-functions or even the entire PES may be shut down.

All HIMatrix devices and modules are equipped with LEDs to indicate that faults have been detected. This allows the user to quickly diagnose faults detected in a device or the external wiring.

Additionally, the user program can evaluate various system variables displaying the device and module status.

Extensive diagnostics of the system performance and detected faults are stored in the diagnostic memory of the controllers. The diagnostics can also be read after a system fault using the PADT.

For more information on how to evaluate diagnostic messages, refer to the HIMatrix system manual (HI 800 141 E).

For a very few number of component failures that do not affect safety, the HIMatrix system does not provide any diagnostic information.

3.1.3 PADT

Using the PADT, the user creates the program and configures the controller. The safety concept of the PADT supports the user in the proper implementation of the control task. The PADT implements numerous measures to check the entered information.

The PADT is a personal computer installed with the programming tool.

3.1.4 Structuring Safety Systems in Accordance with the Energize to Trip Principle

Safety systems operating in accordance with the energize to trip principle have the following functions:

1. The safe state of a device is the de-energized state. This state is adopted, for instance, if a fault has occurred in the device.
2. The controller can trigger the safety function on demand by switching on an actuator.

3.1.4.1 Detection of Failed System Components

Thanks to the automatic diagnostic function, the safety system is able to detect that components have failed.

3.1.4.2 Safety Function in Accordance with the Energize to Trip Principle

The safety function is performed when the safety system energizes one or several actuators, thus ensuring that the safe state is adopted.

The user must plan the following actions:

- Line monitoring (short-circuits and open-circuits) within input and output devices. These functions must be configured accordingly.
- The operation of the actuators can be monitored through a position feedback.

3.2 Time Parameters Important for Safety

Time parameters important for safety are:

- Process safety time
- Safety time of the controller
- User program safety time
- Worst case response time
- Resource watchdog time
- Watchdog time of the user program

3.2.1 Process Safety Time

The process safety time is a property of the process and describes the time interval during which the process allows faulty signals to exist without a safety-critical state occurring.

3.2.2 Safety Time of the Controller

The safety time is the time in which the controller in the RUN state has to react to the occurrence of an internal fault.

From the process view point, the safety time is the maximum time within which the safety system must provide a response at the output after a change of the input signals (response time if faults occur). The safety time can be configured within 20...22 500 ms.

3.2.3 User Program Safety Time

The safety time for the user program cannot be set directly. To calculate the safety time for a user program, HIMatrix uses the parameters Max. Safety Time of the resource and Maximum Number of Cycles. Refer to Chapter 8.2.10 for more details.

3.2.4 Worst Case Response Time

The worst case response time applies to an undisturbed system. It is the maximum time that a HIMatrix system may require to respond to a change of an input signal with an output signal. In HIMatrix controllers running in cycles, the worst case response time is twice the maximum cycle time. The requisites are:

- The user program logic is designed so that delays, e.g., due to unfavorable execution order, cannot occur.
- The entire user program must be processed within one CPU cycle.
- Response-essential data are not exchanged between various user programs.

The cycle time of the controller includes processing of the following tasks:

- Reading the inputs
- Processing the user program/s
- Writing to the outputs
- Process data communication
- Performing test routines

3.2.5 Resource Watchdog Time

The watchdog time is preset in SILworX in the dialog box for configuring the resource properties. This time is the maximum permissible duration of a RUN cycle (cycle time). If the cycle time exceeds the preset watchdog time, the system is shut down. Afterwards, if Autostart was configured, the system restarts. If Autostart is not configured, the system enters the STOP/VALID CONFIGURATION state.

The processor system watchdog time may be set to ≤ ½ * PES safety time.

Range of values for the watchdog time	Default value for the controllers	Default value for the remote I/Os
4…5000 ms	200 ms	100 ms

Table 2: Range of Values for the Watchdog Time

i Determine the safety time and the watchdog time for the system to be controlled.

3.2.5.1 Calculating a Suitable Watchdog Time

The following must apply for the watchdog time: **watchdog time ≤ ½ * safety time**

The following calculations are based on the assumption of typical safety-related automation networks. These networks are almost only composed of HIMA devices of the HIMatrix and HIMax families.

In addition to HIMA devices, extended networks include numerous other participants. Within extended networks, the activities of other participants can strongly increase the cycle time of the HIMatrix PES. For instance, a broadcast to all, possibly to more than 1000 participants may increase the cycle time by 20…40 ms.

Such factors must extra be taken into account when determining the watchdog time!

Extended networks are the exception.

3.2.5.2 Conservative Estimate of the Watchdog Time through Testing

The following estimate applies under the following condition:

The HIMatrix system is located in a network that is not disturbed and is subject to a normal communication load. Exceptional loads, e.g., due to frequent broadcast telegrams or defective devices are precluded.

This condition is met under the assumption of a homogenous network that only includes safety-related PES.

In any case, the estimated watchdog time must be verified during the factory acceptance test (FAT) and the site acceptance test (SAT) using the real cycle times by evaluating the cycle time statistics.

If the resulting values are not expected or the mentioned condition cannot be reliably met, HIMA recommends contacting HIMA customer support.

To determine a suitable value, HIMA recommends testing a system that is as complete as possible:

- The HIMatrix hardware is completely mounted, e.g., the F60 rack includes all designated modules.
- Communication partners, including remote I/Os, are available.
- The user program logic is as complete as possible.

To determine the minimum value for the watchdog time

1. Operate the system under full load. Communication should also run under full load.
2. Specify input data to allow that the longest program paths are preferably passed through. To this end, input value sequences may be necessary.
3. Reset the cycle time statistics in the Control Panel.
4. Perform the reload multiple times, if the resource is capable of reload.
5. In the Control Panel, observe the maximum cycle time values.
   ☑ T_{Cycle} is identified.
6. Operate the system for a long time and note down the greatest cycle time values of the user programs at basic loads and load peaks.
   ☑ TPeak is identified.
7. Calculate the minimum watchdog time T_{WD} using:
   $T_{WD} = T_{Cycle +} T_{Res} + T_{Com} + T_{Config} + \Delta T_{Peak}$, where

   T_{Cycle} Observed maximum cycle time (basic load, already includes portions of T_{Com} and T_{Config})

   T_{Marg} Safety margin 6 ms

   T_{Com} The configured system parameter: *Max. Com.Time Slice ASYNC [ms]*

   In this case, do not use the preset value, but set a value more suitable for the project in accordance with the expected communication load (e.g., safe**ethernet**, Modbus)!

   T_{Config} The configured system parameter *Max. Duration of Configuration Connections [ms].*

   In this case, do not use the preset value, but set a value calculated for the project, see Chapter 7.4.1.2!

   TPeak Observed load peaks less observed basic load, see Steps 6.

► The value set for the watchdog time should be: determined minimum value + margin for future changes or extensions.

The maximum cycle time values during the reload depend on the configured watchdog time. If the PES 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.

3.2.6 Watchdog Time of the User Program

Each user program has its own watchdog and watchdog time.

The watchdog time for the user program cannot be set directly. To calculate the watchdog time for a user program, the HIMatrix system uses the parameters Watchdog Time [ms] of the resource and Maximum Number of Cycles.

Make sure that the calculated watchdog time is not greater than the response time required for the process portion processed by the user program.

3.3 Safety Requirements

The following safety requirements must be met when using the safety-related components of the HIMatrix system:

3.3.1 Hardware Configuration

Personnel configuring the HIMatrix hardware must observe the following safety requirements.

Product-Independent Requirements

- To ensure safety-related operation, only approved fail-safe hardware and software may be used. The approved hardware and software are listed in the *Revision List of Devices and Firmware of HIMatrix Systems of HIMA Paul Hildebrandt GmbH*. The latest versions can be found in the version list maintained together with the test authority. The latest version list can be downloaded from the HIMA website.
- The operating requirements specified in this safety manual (see Chapter 3.5) about EMC, mechanical, chemical, climatic influences must be observed.
- Non-fail-safe, interference-free hardware and software may be used for processing non-safety-relevant signals, but not for handling safety-related tasks.

Product-Dependent Requirements

- Only devices that are safely separated from the power supply may be connected to the system.
- The safe electrical power supply separation must be guaranteed within the 24 V system supply. Only power supply units ensuring that the controllers and remote I/O modules are supplied with 24 V low voltage may be used.
- To comply with the protective provisions for electrical safety and grounding, the manufacturer of the specific application must ensure that proper measures are implemented for separating the indoor and outdoor equipment in accordance with EN 50122. This shall protect the HIMatrix systems against influences from the outdoor equipment in the overhead contact line zone or the pantograph zone, as well as against traction return currents. Power supply devices allowed for railway applications must be used.

3.3.2 Programming

Personnel developing user programs must observe the safety requirements specified below.

Product-Independent Requirements

- In safety-relevant applications, ensure that the safety-relevant system parameters are properly configured. The safety manual describes the possible configurations, see Chapter 7.4.
- In particular, this applies to the system configuration, maximum cycle time and safety time, see Chapter 3.2.

3.3.3 Requirements for Using the Programming Tool

- **SILworX** must be used for programming.
- Once the application has been created, compile the program twice and compare the two resulting configuration CRCs to ensure that the program was compiled properly.
- The proper implementation of the application specification must be validated and verified. A comprehensive test of the logic must be performed by trial.
- The system response to faults in the fail-safe input and output modules must be defined in the user program in accordance with the system-specific safety-related conditions.
- The SILworX programming tool is provided with a feature that, after the user program or system configuration has changed, only displays the performed changes. The analysis of the changes (change impact analysis IA) must define the required test scope. This impact analysis must take the expected changes based on the performed modifications, the result of the SILworX comparison feature and the required regression tests into account.

3.3.4 Communication

- When implementing safety-related communications between various devices, ensure that the overall response time does not exceed the permitted response time. All calculations must be performed in accordance with the rules given in Chapter 9.2.
- Data must be transmitted over closed transmission systems (Category 1) in accordance with EN 50159.
- Open transmission systems (Category 2 and Category 3) in accordance with EN 50159 may be used, if additional measures are taken to guarantee that the transmission channel is secure (e.g., firewalls or encryption).
- At this stage, the serial interfaces may only be used for non-safety-related purposes.
- Only devices with electrically protective separation may be connected to the communication interface.

3.3.5 Requirements for Railway Applications

- The relevant standards must be used for railway applications.
- The digital outputs are equipped with line short-circuit monitoring. Responses to detected short-circuits must be programmed in the user program.
- The temperature state (operating temperature) of the HIMatrix systems must be evaluated in the user program. Also safety-related measures must be triggered in the user program. For more information, refer to HIMatrix system manual (HI 800 141 E).
- Error messages must be evaluated in the user program. Errors are signaled by state bits and are available to the user program. Additionally, errors are stored in the diagnostic memory of the controller and can be evaluated using the programming tool. For more information, refer to HIMatrix system manual (HI 800 141 E).
- Detection of ground faults must be configured externally.

3.3.6 Cyber Security for HIMatrix Systems

Industrial controllers must be protected against IT-specific problem sources. Those problem sources are:

- Attackers inside and outside of the customer's plant
- Operating failures
- Software failures

A HIMatrix installation consists of the following parts to be protected:

- HIMatrix PES
- PADT
- OPC Server: X-OPC DA, X-OPC AE (optional)
- Communication connections to external systems (optional)

HIMatrix with basic settings is already a system fulfilling the requirements for cyber security (IT security).

Protective mechanisms for preventing unintentional or unapproved modifications to the safety system are integrated into the PES and the programming tool:

- Each change to the user program or configuration results in a new configuration CRC.
- The operating options depend on the rights of the user logged into the PES.
- The programming tool prompts the user to enter a user name and password in order to log in to the PES.
- PES variables can only be accessed if the PADT is operating with the current version of the user project (archive maintenance!).
- Connection between the PADT and PES is not required in RUN and can be interrupted.
  The PADT can be shortly connected for maintenance work or diagnostic tasks.

All requirements for protection against manipulation specified in the safety and application standards must be met. The operator is responsible for authorizing employees and implementing the required protective actions.

⚠ WARNING

Physical injury possible due to unauthorized manipulation of the controller!

The controller must be protected against unauthorized access!

Examples:

- **Changing the default settings for login and password!**
- **Controlling the physical access to the controller and PADT!**

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

- 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.

The user is responsible for implementing the necessary measures in a way suitable for the plant!

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

3.4 Test Conditions

Refer to the HIMatrix safety manual (HI 800 023 E) for the standards used to test and certify the HIMatrix system for industrial use.

3.5 Additional test conditions for railway applications

The following table shows the HIMatrix variants that are approved for railway applications:

Compact controllers
F30 034
F35 034
Remote I/Os
F1 DI 16 014
F2 DO 4 01
F2 DO 8 014
F2 DO 16 014
F2 DO 16 02
F3 AIO 8/4 014
F3 DIO 8/8 014
F3 DIO 16/8 014
F3 DIO 20/8 024
Modular system F60
PS 014
CPU 034
AI 8 014
CIO 2/4 014
DI 24 014
DI 32 014
DIO 24/16 014
DO 8 01
MI 24 014
GEH 014

Table 3: HIMatrix Variants Available for Railway Applications

The HIMatrix variants for railway applications have been additionally developed to meet the following EMC, climatic and environmental requirements:

3.5.1 Altitude Range

The following classes in the specified altitude range apply to the HIMatrix variants:

- For use in signaling applications in accordance with EN 50125-3: AX up to 2000 m
- For use on rolling stock in accordance with EN 50125-1: AX up to 2000 m

3.5.2 Climatic Requirements

All the standard variants of the HIMatrix system family are designed and tested for temperature range of 0…60 °C and a relative air humidity of 10…95 % (non-condensing). The following temperature classes result for railway applications in accordance with EN 50125-3:

HIMatrix	In external ambient	In control cabinet	In container		In building	
			N.T.C	T.C	N.C.C.	C.C
Standard	-	-	-	T1, T2, TX	T1	T1, T2, TX

Table 4: Standard HIMatrix Temperature Classes in Accordance with EN 50125-3

NOTICE

The standard variants of the HIMatrix system family are not approved for use on rolling stock in accordance with EN 50155 since they are not equipped with protective lacquer.

3.5.2.1 Use in Signaling Applications

The HIMatrix variants for railway applications are designed for a temperature range of -25…+70 °C. All the HIMatrix variants for railway applications were tested in accordance with EN 50125-3 and can be used in the following temperature classes:

HIMatrix	In external ambient	In control cabinet	In container		In building	
			N.T.C	T.C	N.C.C.	C.C
F30 034	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F35 034	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F1 DI 16 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F2 DO 8 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F2 DO 16 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F3 AIO 8/4 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F3 DIO 8/8 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F3 DIO 16/8 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
F3 DIO 20/8 024	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
PS 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
CPU 034	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
AI 8 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
CIO 2/4 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
DI 24 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
DI 32 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
DIO 24/16 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX
MI 24 014	T1	T1	T1, T2	T1, T2, TX	T1, T2, TX	T1, T2, TX

Table 5: Temperature Classes in Accordance with EN 50125-3

3.5.2.2 Use on Rolling Stock

All the HIMatrix variants for railway applications were tested in accordance with EN 50155 and can be used in the following temperature classes:

HIMatrix	Temperature classes
F30 034	T1
F35 034	T1
F1 DI 16 014	T1
F2 DO 8 014	T1
F2 DO 16 014	T1
F3 AIO 8/4 014	T1
F3 DIO 8/8 014	T1
F3 DIO 16/8 014	T1
F3 DIO 20/8 024	T1
PS 014	T1
CPU 034	T1
AI 8 014	T1
CIO 2/4 014	T1
DI 24 014	T1
DI 32 014	T1
DIO 24/16 014	T1
MI 24 014	T1

Table 6: Temperature Classes in Accordance with EN 50155

3.5.2.3 Derating of Digital Outputs

With an operating temperature higher than 60 °C the load of the digital outputs must be derated. In this case, each output can be loaded with a maximum of 0.5 A, see the device-specific manuals.

3.5.3 Mechanical Requirements

The HIMatrix components were tested in accordance with EN 50125-3 and EN 50155.

3.5.3.1 Use in Signaling Applications

The devices and modules listed in Table 3 have been mechanically tested in accordance with EN 50125-3. The following table lists the most important tests and limits for mechanical requirements:

EN 50125-3	Mechanical Tests
	Vibration immunity test: 2.3 m/s^2 between 5...2000 Hz, HIMatrix in operation
	Shock immunity test: 20 $m/s^{2,}$ 11 ms, HIMatrix in operation

Table 7: Mechanical Requirements for Use in Signaling Applications

3.5.3.2 Use on Rolling Stock

The devices and modules listed in Table 3 have been mechanically tested in accordance with EN 50155. Testing was performed in accordance with EN 61373, Category 1, Class B.

3.5.4 EMC Requirements

The following table lists the most important tests and limits for EMC requirements:

Test standard	Type of assessment	Interference immunity tests
EN 61000-4-2	ESD test	6 kV contact discharge, 8 kV air discharge
EN 61000-4-3	EM field	80…1000 MHz: 10 V/m 800…1000 MHz: 20 V/m 1400…2000 MHz: 10 V/m 2000…2700 MHz: 5 V/m 5100…6000 MHz: 3 V/m
EN 61000-4-4	Burst test	Supply voltage: 2 kV I/O lines: 2 kV Ground: 1 kV
EN 61000-4-5	Surge [1)]	Supply voltage: 2 kV CM 1 kV DM I/O lines: 2 kV CM 1 kV DM Shielded lines: 2 kV CM
EN 61000-4-6	Conducted disturbances	Supply voltage: 10 V I/O lines: 10 V Ground: 10 V
EN 61000-4-8	Power frequency magnetic field	16 2/3 Hz, 50 Hz, 60 Hz: 100 A/m DC 300 A/m
[1)] The H 7013 external filter is absolutely required if HIMatrix compact systems are used. Surge absorbers from other manufacturers may be used, if the specifications provided in the data sheets are equivalent or better.		

Table 8: EMC Requirements for Use in Signaling Applications in Accordance with EN 50121-4

i External surge filters are also required for all unshielded input and output lines, if they are connected to lines within the 3-meter or, within the 10-meter range, if they are connected to lines that are longer than 30 m.

External filters for relay outputs are not required in the F60 DO 8 01, remote I/Os F2 DO 8 014 and F2 DO 16 024.

Test standard	Type of assessment	Interference immunity tests
EN 61000-4-2	ESD test	6 kV contact discharge, 8 kV air discharge
EN 61000-4-3	EM field	80…1000 MHz: 20 V/m 1400…2000 MHz: 10 V/m 2000…2700 MHz: 5 V/m 5100…6000 MHz: 3 V/m
EN 61000-4-4	Burst test	Supply voltage: 2 kV I/O lines: 2 kV
EN 61000-4-5	Surge [1)]	Supply voltage: 2 kV CM 1 kV DM
EN 61000-4-6	Conducted disturbances	Supply voltage: 10 V I/O lines: 10 V
[1)] The H 7013 external filter is absolutely required if HIMatrix compact systems are used. Surge absorbers from other manufacturers may be used, if the specifications provided in the data sheets are equivalent or better.		

Table 9: EMC Requirements for Use on Rolling Stock in Accordance with EN 50121-3-2

The devices and modules specified in Table 3 were successfully tested and met the EMC requirements in accordance with EN 50121-4 and EN 50121-3-2.

3.5.5 Severe Requirements

The HIMatrix system must be installed in enclosures with suitable degree of protection (e.g., IP54) to ensure protection against the environmental influences as of classes 4C3, 4B1 and 4S2.

3.5.6 Supply Voltage

The following table lists the most important tests and limits for the supply voltage of the HIMatrix systems:

IEC/EN 61131-2	Verification of the DC supply characteristics
	Voltage range test: 24 VDC, -15...+20 %, $r_p \leq 5$ %
	Momentary external current interruption immunity test: DC, PS 2: 10 ms
	Reversal of DC power supply polarity test: Tested for 10 s

Table 10: Supply Voltage Failures Immunity Test

3.5.6.1 Supply Voltage Requirements for Use on Rolling Stock

The supply of the HIMatrix systems occurs from an accumulator battery with 24 V nominal voltage.

The following values apply to the HIMatrix supply voltage: 24 VDC, -15…+20 %, 5 % ripple.

This results in the following tolerance values:

- Minimum voltage: 19.2 V (0.8 U_N)
- Nominal voltage: 24 V (U_N)
- Rated voltage: 27.6 V (1.15 U_N)
- Maximum voltage: 30 V (1.25 U_N)

The HIMatrix variants specified in Table 3 were tested in accordance with EN 50155, Chapter 5.1.

Taking external measures, the user must ensure that a minimum voltage of 0.8 U_N is maintained, since otherwise individual devices or the entire system will reboot.

Taking external measures, the user must be able to intercept voltage fluctuations above 1.25 U_N in accordance with EN 50155, Chapter 5.1.1.1.

HIMatrix systems are designed for voltage interruptions of up to 20 ms. As such, the HIMatrix meets the requirement of Class S2 in accordance with EN 50155, Chapter 5.1.1.2.

The HIMatrix system meets the requirements for DC voltage ripple factor in accordance with EN 50155, Chapter 5.1.1.4.

If a power or rotary converter is used to supply the system, the HIMatrix meets the requirements for normal operation in accordance with EN 50155, Chapter 5.1.2. For permissible voltage fluctuations in accordance with the standard, external measures must be implemented by the user.

The requirements in accordance with EN 50155, Chapter 5.1.3, for switching two supply voltages are not met. External measures must be implemented by the user.

4 Central Functions

The devices of type F1..., F2..., F3... are compact systems that cannot be modified.

The controllers of type F60 are modular systems that, when combined with a power supply module and a processor module, may be used with up to six I/O modules.

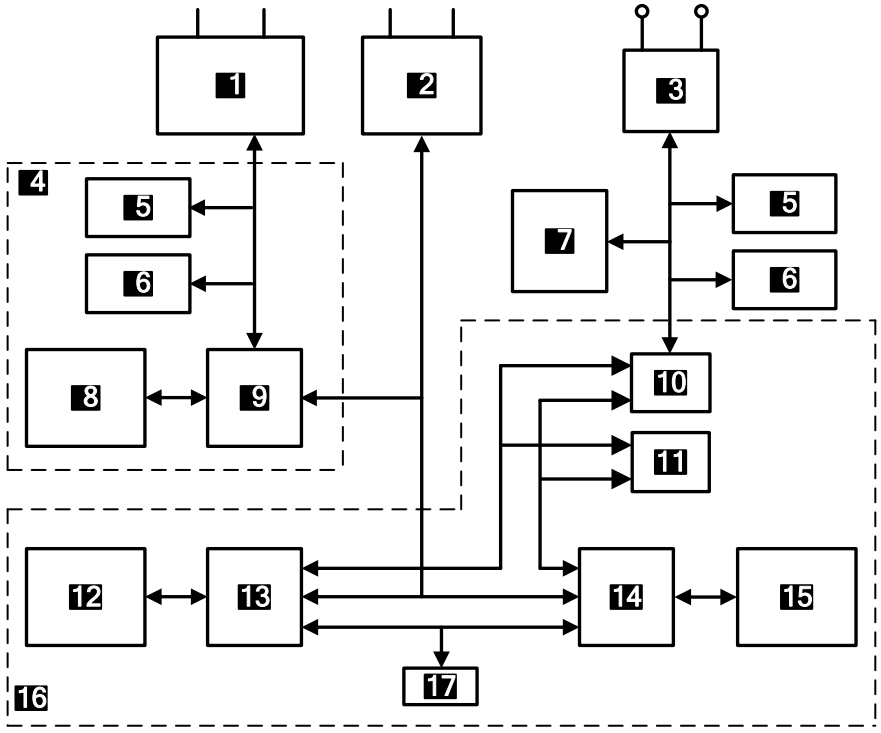
4.1 Power Supply Units

The HIMatrix systems must be supplied by power supply units ensuring a 24 V low voltage to the controllers and remote I/Os.

Observing the permitted voltage limits guarantees the controller's proper operation.

4.2 Functional Description of the Processor System

The processor system is the central component of the controller. It is composed of the following function blocks:



1 Fieldbus interfaces
2 Ethernet interfaces
3 I/O bus module
4 Communication system
5 NVSRAM
6 Flash
7 V_{CC} and temperature monitoring
8 Communication system SDRAM
9 Communication system processor
10 Comparator
11 Watchdog
12 SDRAM 1 of the processor system
13 Processor 1 of the processor system
14 Processor 2 of the processor system
15 SDRAM 2 of the processor system
16 Safety-related processor system
17 Real time clock

Figure 1: Function Blocks of the F60 CPU 034

Characteristics of the Processor System:

- Two synchronous microprocessors (processor 1 and processor 2).
- Each microprocessor has its own SDRAM memory.
- Testable hardware comparators for all external accesses of both microprocessors.
- In the event of an error, the watchdog is set to the safe state.
- Flash EPROM, the program memory for operating systems and user programs, suitable for at least 100 000 memory cycles.
- Data memory in NVSRAM.
- Gold capacitor for buffering date/time.
- Communication processor for fieldbus and Ethernet connections.
- Interface for data exchange among devices, F60 controllers and the PADT, based on Ethernet.
- Optional interface(s) for data exchange via fieldbus.
- LEDs for indicating the system states.
- I/O bus logic for connection to I/O modules.
- Safe watchdog (WD).
- Monitoring of power supply units, testable (1.8 VDC / 3.3 VDC).
- Temperature monitoring.

4.3 Self-Tests

The self-test devices detect single faults that may lead to a safety-critical operating state and trigger, within the safety time of the controller, predefined fault responses which bring the faulty components into the safe state.

The following section specifies the most important self-test routines of safety-related processor systems.

4.3.1 Microprocessor Test

The following is tested:

- All instructions and addressing modes used.
- The writability of the flags and the commands affected by the flags.
- The writability and crosstalk of the registers.

4.3.2 Memory Areas Test

The operating system, user program, constants and parameters as well as the variable data are stored in memory areas of both processors and are tested by a hardware comparator.

4.3.3 Protected Memory Areas

The operating system, user program and parameter area are each stored in a memory. They are secured by write protection and a CRC test.

4.3.4 RAM Test

A write and read test is performed to check the modifiable RAM areas, in particular for stuck-at issues and crosstalk.

4.3.5 Watchdog Test

The watchdog signal is switched off if it is not triggered by both CPUs within a defined time window and if the hardware comparator test fails. An additional test determines the switch-off ability of the watchdog signal.

4.3.6 Test of the I/O Bus Inside the Controller

The connection between the CPU and the associated inputs and outputs (I/O modules) is tested.

4.4 Responses to Faults in the Processor System

A hardware comparator within the processor system constantly checks if the data from microprocessor 1 are identical to the data from microprocessor 2. If they are different, or if the test routines detect a fault, the watchdog signal is switched off. This means that the input signals are no longer processed by the controller and the outputs switch to the de-energized, switched-off state.

If such a fault occurs for the first time, the controller is restarted (reboot). If a further internal fault occurs within the first minute after start-up, the controller enters the STOP/INVALID CONFIGURATION state and will remain in this state.

4.5 Fault Diagnosis

Each F60 module has an own LED for reporting module malfunctions or faults in the external wiring. This allows the user to quickly diagnose faults detected in a module.

In the compact systems F1..., F2..., F3..., these error messages are grouped in one common error message.

Additionally, the user program can evaluate various system variables associated with the inputs, outputs or the controller.

Faults are only signaled if they do not hinder communication with the processor system, i.e., the processor system must be still able to evaluate the faults.

The user program logic can evaluate the error codes of the system variables and of all the input and output signals.

Extensive diagnostics of the system performance and detected faults are stored in the diagnostic memory of the processor and the communication system. The diagnostics can also be read after a system fault using the PADT.

For more information on how to evaluate diagnostic messages, refer to the HIMatrix system manual (HI 800 141 E).

5 Inputs

Overview of the HIMatrix system inputs:

Device / module	Type	Number	Safety-related	Interference-free	Electrically separated
Compact Systems					
F30 034	Digital	20	•	•	– [1]
F35 034	Digital	24	•	•	– [1]
	24-bit counter	2	•	•	– [1]
	Analog	8	•	•	– [1]
F1 DI 16 014	Digital	16	•	•	– [1]
F3 DIO 8/8 014	Digital	8	•	•	– [1]
F3 DIO 16/8 014	Digital	16	•	•	– [1]
F3 AIO 8/4 014	Analog	8	•	•	– [1]
F3 DIO 20/8 024	Digital	20	•	•	– [1]
Modular System F60					
DIO 24/16 014	Digital	24	•	•	•
DI 32 014 (configurable for line control)	Digital	32	•	•	•
DI 24 014 (110 V)	Digital	24	•	•	•
CIO 2/4 014	24-bit counter	2	•	•	•
AI 8 014	Analog	8	•	•	•
MI 24 014	Analog or digital	24	•	•	•
[1] Ground L-					

Table 11: Overview of the Inputs

5.1 General

Safety-related inputs can be used for both safety-related and non-safety-related signals.

The controllers provide status and fault information as follows:

- Through the diagnostic LEDs on the devices and modules.
- Using system variables that the user program is able to evaluate.
- Storing messages in the diagnostic memory which can be read by the PADT.

Safety-related input modules automatically perform high-quality, cyclic self-tests during operation. These test routines are TÜV-tested and monitor the safe functioning of the corresponding module.

For a few number of component failures that do not affect safety, no diagnostic information is provided.

5.2 Safety of Sensors, Encoders and Transmitters

In safety-related applications, the controller and connected sensors, encoders and transmitters must all meet the safety requirements and achieve the specified SIL. For information on how to achieve the required SIL for sensors, see IEC 61511-1, Section 11.4.

5.3 Response in the Event of a Fault

If the test routines detect an error, the following responses are triggered:

- With respect to inputs, the user program processes the initial value of the global variables.
- The error code and other system variables can be used to program application-specific fault responses. Refer to the module-specific manual for more details.

If an error occurs, a compact system activates the *ERROR* LED, a F60 module the *ERR* LED.

5.4 Safety-Related Digital Inputs

The described properties apply to both digital input channels of F60 modules and digital input channels of all compact systems (unless stated otherwise).

5.4.1 General

The digital inputs are read once per cycle and saved internally; cyclic tests are performed to ensure their safe functioning.

Under certain circumstances, input signals that are present for shorter than the time between two samplings, i.e., shorter than a cycle time, are not detected.

5.4.2 Test Routines

The test routines check whether the input channels are able to forward both signal levels (low and high), irrespective of the signals actually present on the input. This functional test is performed whenever the input signals are read.

5.4.3 Surges on Digital Inputs

Due to the short cycle time of the HIMatrix systems, a surge pulse as described in EN 61000-4-5 can be read in to the digital inputs as a short-term high level.

The following measures ensure proper operation in environments where surges may occur:

1. Install shielded input wires.
2. Program noise blanking in the user program. A signal must be present for at least two cycles before it is evaluated. This measure increases the maximum response time!

i The measures specified above are not necessary if the plant design precludes surges within the system.

In particular, the design must include protective measures with respect to overvoltage, lightning, earth grounding and plant wiring in accordance with the relevant standards and the instructions specified in the HIMatrix system manual (HI 800 141 E).

5.4.4 Configurable Digital Inputs

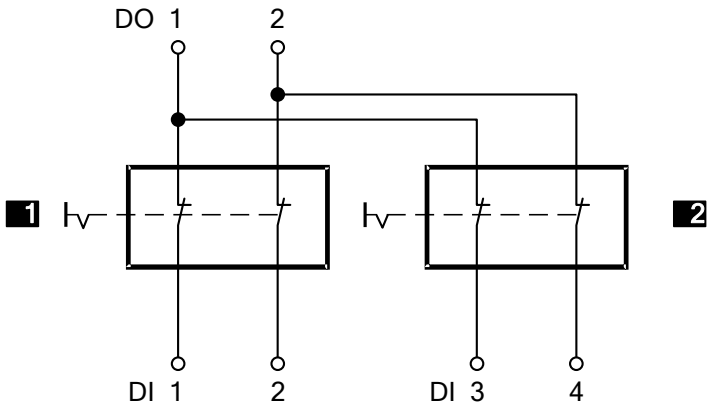
The digital inputs of the F35 034 controller and the MI 24 014 module operate as analog inputs, but return digital values due to the configuration of switching thresholds.

For configurable digital inputs, the test routines and safety-related functions mentioned for analog inputs apply as specified in Chapter 5.5.1.

5.4.5 Line Control

Line control is used to detect short-circuits or open-circuits e.g., on EMERGENCY STOP devices and can be configured for the HIMatrix systems with digital inputs (not for the F35 034 controller and MI 24 014 module).

To this end, connect the digital outputs of the system to the digital inputs of the same system as follows (example):

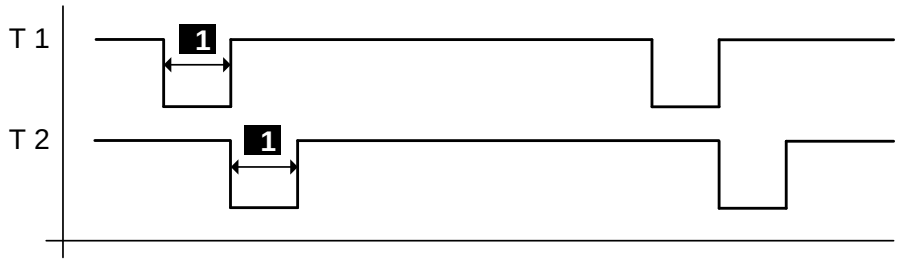


1 EMERGENCY STOP 1
2 EMERGENCY STOP 2

EMERGENCY STOP switches in accordance with EN 60947-5-1 and EN 60947-5-5

Figure 2: Line Control

The controller pulses the digital outputs to detect short-circuits and open-circuits on the lines connected to the digital inputs. To do so, configure the *Value [BOOL]* -> system variable in SILworX. The pulsed outputs can be assigned to any digital inputs.



1 Configurable 5...2000 µs

Figure 3: Pulsed Signal T1, T2

An (evaluable) error code is created, if the following errors occur:

- Cross-circuit between two parallel wires.
- Invalid connections of two lines (e.g., DO 2 to DI 3).
- Ground fault on one wire (with earthed ground only).
- Open-circuit or open contacts.

Refer to the HIMatrix system manual (HI 800 141 E) for a description of and further details on line control.

5.5 Safety-Related Analog Inputs (F35 034, F3 AIO 8/4 014 and F60)

The analog input channels convert the measured input currents into an INTEGER value. The values are available to the user program as variables that are assigned to the system variable -> *Value [INT]*:

The range of values for the inputs depends on the device or module:

- F35 034 controller:

Input channels	Measurement procedure	Current, voltage	Range of values in the application	
			FS1000 [1)]	FS2000 [1)]
8	Unipolar	0…+10 V	0…1000	0…2000
8	Unipolar	0…20 mA	0…500 [2)] 0…1000 [3)]	0…1000 [2)] 0…2000 [3)]

[1)] Configurable by selecting the type in the PADT
[2)] With external 250 Ω shunt adapter
[3)] With external 500 Ω shunt adapter

Table 12: Analog Inputs of the F35 034 Controller

- F3 AIO 8/4 014 remote I/O:

Input channels	Measurement procedure	Current, voltage	Range of values in the application
8	Unipolar	0…+10 V	0…2000
8	Unipolar	0/4…20 mA	0…1000 [1)] 0…2000 [2)]

[1)] With external 250 Ω shunt adapter
[2)] With external 500 Ω shunt adapter

Table 13: Analog Inputs of the F3 AIO 8/4 014 Remote I/O

- F60 controller

Input channels	Measurement procedure	Current, voltage	Range of values in the application	
			FS1000 [1)]	FS2000 [1)]
AI 8 014				
8	Unipolar	-10…+10 V	-1000…1000	-2000…2000
8	Unipolar	0…20 mA	0…1000 [3)]	0…2000 [3)]
8	Unipolar	0…20 mA	0…500 [2)]	0…1000 [2)]
4	Bipolar	-10…+10 V	-1000…1000	-2000…2000
MI 24 014				
	Unipolar	0…20 mA	0…2000 [4)]	

[1)] It can be configured by selecting the type in the PADT (F60)
[2)] with external 250 Ω shunt
[3)] with external 500 Ω shunt (accuracy 0.05 %, 1 W) No longer available at HIMA.
[4)] internal shunts

Table 14: Analog Inputs of the F60 Controller

The AI 8 014 module of the HIMatrix F60 can be configured in the user program for eight unipolar or four bipolar functions. However, it is not allowed to combine functions on a module.

The analog inputs of the F35 034 controller, the F3 AIO 8/4 014 remote I/O and the AI 8 014 module operate with voltage measurement. With the analog inputs of the F35 034 and F3 AIO 8/4 014, digital outputs of the own system (F35 034) or other HIMatrix controllers can be monitored to detect open-circuits. Refer to the manual of the corresponding HIMatrix controllers for more details.

If no line monitoring function is performed by the system, random input values are processed on the high-resistance inputs. The value resulting from this floating input voltage is not reliable; for voltage inputs, the channels must be terminated with a 10 kΩ resistor. The internal resistance of the source must be taken into account.

To measure currents, the shunt is connected in parallel to an input; in doing so the 10 kΩ resistor is not required.

The inputs of the MI 24 014 module are only current inputs, because of the internal shunts, and cannot be used as voltage inputs.

If input channels are not used, the measurement input must be connected to the ground. If an open-circuit occurs, negative influences (fluctuating input voltages) on other channels can thus be avoided. It is sufficient not to assign unused inputs global variables.

5.5.1 Test Routines

The analog values are processed in parallel via two multiplexers and two analog/digital converters with 12-bit resolution and the results are compared. Additionally, test values are used by the existing D/A converters, converted back to digital values, and then compared with the default value.

5.6 Safety-Related Counters (F35 034 and F60)

Unless otherwise noted, the points previously mentioned apply for the counter module of the F60 system as well as for the counters of the F35 034.

5.6.1 General

A counter channel can be configured for operation as a high-speed up or down counter with 24-bit resolution or as a decoder in Gray code.

If used as high-speed up or down counters, the pulse input and count direction input signals are required in the application. The counter is only reset in the user program.

The counter encoders have the following resolutions:

- F60 CIO 2/4 014: 4-bit or 8-bit resolution.
- F35 034: 3-bit or 6-bit resolution.

A reset is possible.

Two independent 4-bit inputs may only be connected to an 8-bit input (example of F60) using the user program. No switching option is planned for this purpose.

The encoder function monitors the change of the bit pattern on the input channels. The bit patterns on the inputs are directly transferred to the user program. They are represented in the PADT as decimal numbers corresponding to the bit pattern (*Counter[0x].Value*).

Depending on the application, this number (which corresponds to the Gray code bit pattern) can be converted into, for example, the corresponding decimal value.

5.7 Checklist for Safety-Related Inputs

HIMA recommends using the available checklist for engineering, programming and starting up safety-related inputs. The checklist can be used as a planning document and also serves as proof of careful planning.

When engineering or starting up the system, a checklist must be filled out for each of the safety-related input channels used in the system to verify the requirements to be met. This is the only way to ensure that all requirements were considered and clearly recorded. The checklist also provides documentation about the connection between the external wiring and the user program.

The checklist *HIMatrix_Checklist_Inputs.doc* is available as Microsoft© Word© document. The ZIP file *HIMatrix_Checklists.zip* contains all the checklists and can be obtained upon request by sending an e-mail to documentation@hima.com.

6 Outputs

Overview of the HIMatrix system outputs:

Device	Type	Number	Safety-related	Galvanically separated
Compact Systems				
F30 034 (configurable for line control)	Digital	8	•	–[1)]
F35 034	Digital	8	•	–[1)]
F1 DI 16 014	Pulse	4	-	–[1)]
F2 DO 4 01	Digital	4	•	–[1)]
F2 DO 8 014	Relay	8	•	
F2 DO 16 014	Digital	16	•	–[1)]
F2 DO 16 02	Relay	16	•	
F3 DIO 8/8 014	Digital 1-pole	8	•	–[1)]
	Digital 2-pole	2		
F3 DIO 16/8 014	Digital 1-pole	16	•	–[1)]
	Digital 2-pole	8		
F3 AIO 8/4 014	Analog	4	-	–[1)]
F3 DIO 20/8 024 (configurable for line control)	Digital	8	•	–[1)]
Modular System F60				
DIO 24/16 014 (configurable for line control)	Digital	16	•	
DO 8 01 (250 V)	Relay	8	•	•
CIO 2/4 014	Digital	4	•	
[1)] Ground L-				

Table 15: Overview of the Outputs

6.1 General

The controller writes to the safety-related outputs once per cycle, reads back the output signals and compares them with the specified output data.

The safe state of the outputs is the *0* value or an open relay contact.

Three testable switches are integrated in series in the safety-related output channels. Thus, a second independent shutdown option, which is a safety requirement, is integrated into the output module. If a fault occurs, this integrated safety switch-off option safely de-energizes the individual channels of the defective output module (de-energized state).

Additionally, the watchdog signal of the CPU is the second safety switch-off function: If the watchdog signal is lost, the CPU immediately enters the safe state.

This function is only effective for all the digital outputs and relay outputs of the controller.

The corresponding error code provides additional options for programming fault responses in the user program.

6.2 Safety of Actuators

In safety-related applications, the controller and connected actuators must all meet the safety requirements and achieve the specified SIL. For information on how to achieve the required SIL for actuators, see IEC 61511-1, Section 11.4.

6.3 Response in the Event of a Fault

If the test routines detect a faulty output, the controller switches off the affected output, i.e., it enters the safe state.

The error code and other system variables can be used to program application-specific fault responses. Refer to the module-specific manual for more details.

If an error occurs, a compact system activates the *ERROR* LED, a F60 module the *ERR* LED.

6.4 Safety-Related Digital Outputs

The points listed below apply to both digital output channels of F60 modules and digital output channels of the compact devices. Unless specified otherwise, the relay modules are an exception in either case.

6.4.1 Test Routines for Digital Outputs

The modules are tested automatically during operation. The main test functions are:

- Reading the output signals back from the switching amplifier. The switching threshold for a read-back low level is 2 V. The diodes used prevent the signals from being fed back.
- Checking the integrated redundant safety switch-off function.
- Cyclical shutdown test of the outputs performed as background test for max. 200 µs. The minimum time between two tests is ≥ 20 s.

The operating voltage of the entire system is monitored. All the outputs are de-energized at an undervoltage of < 13 V.

6.4.2 Behavior in the Event of External Short-Circuit or Overload

If the output is short-circuited to L- or overloaded, the device or module is still testable. A safety shutdown is not required.

The controller monitors the device's or module's total current consumption and sets all output channels to the safe state if the threshold is exceeded.

In this state, the outputs are checked every few seconds to determine whether the overload is still present. In a normal state, the outputs are switched on again.

6.4.3 Line Control

The controller can pulse safety-related digital outputs or special pulsed outputs and use them with the safety-related digital inputs of the same system (but not the digital inputs of the F35 034 or F60 MI 24 014) to detect open-circuits and short-circuits (see Chapter 5.4.5).

NOTICE

Malfunctions of the connected actuators are possible!

Pulsed outputs must not be used as safety-related outputs (e.g., for activating safety-related actuators)!

Relay outputs cannot be used as pulsed outputs.

6.5 Safety-Related 2-Pole Digital Outputs

The following points apply to 2-pole digital outputs of the remote I/Os F3 DIO 8/8 014 and F3 DIO 16/8 014.

The devices are tested automatically during operation. The main test functions are:

- Reading the output signals back from the switching amplifier. The diodes used prevent the signals from being fed back.
- Checking the integrated (redundant) safety shutdown.
- Cyclical shutdown test of the outputs performed as background test for max. 200 µs. The minimum time between two tests is ≥ 20 s.
- Line diagnosis with 2-pole connection
  F3 DIO 16/8 014:
  - Short-circuit to L+, L-.
  - Short-circuit between the 2-pole connections.
  - Open-circuit in one of the 2-pole connections.

  F3 DIO 8/8 014:
  - Short-circuit to L+, L-.

The system monitors its operating voltage and de-energizes all outputs at an undervoltage of less than 13 V.

With a 2-pole connection, observe the following notes:

i A relay or actuator connected to the output may accidentally be switched on!

A requirement for applications in machine safety is that the outputs DO+, DO- are switched off if an open-circuit is detected.

i If the requirements previously described cannot be met, observe the following case:

If a short-circuit occurs between DO- and L-, a relay may be energized or some other actuator may be set to a different switching state.

Reason: During the monitoring time specified for line diagnosis, a 24 V level (DO+ output) is present on the load (relay, switching actuator) allowing it to receive enough electrical power to potentially switch to another state.

The monitoring time must be configured such that an actuator cannot be activated by the line diagnosis test pulse.

i Detection of open-circuits may be disturbed!

In a 2-pole connection, no DI input may be connected to a DO output. This would inhibit the detection of open-circuits.

6.5.1 Behavior in the Event of External Short-Circuit or Overload

If the output is short-circuited to L-, L+ or overloaded, the device is still testable. A safety shutdown is not required.

The total current consumption of the device is monitored. If the threshold is exceeded, the device sets all channels to the safe state.

In this state, the device checks the outputs every few seconds to determine whether the overload is still present. In a normal state, the device switches the outputs on again.

6.6 Relay Outputs

The relay outputs correspond to functional digital outputs, but offer galvanic separation and higher electrical strength.

6.6.1 Test Routines for Relay Outputs

The device or the module automatically tests its outputs during operation. The main test functions are:

- Reading the output signals back from the switching amplifiers located before the relays.
- Testing the switching of the relays with forcibly guided contacts.
- Checking the integrated redundant safety switch-off function.

The system monitors its operating voltage and de-energizes all outputs at an undervoltage of less than 13 V.

The outputs of the DO 8 01 module and those of the remote I/Os F2 DO 8 014 and F2 DO 16 02 are equipped with three safety relays:

- Two relays with forcibly guided contacts.
- One standard relay.

This enables the outputs to be used for safety switch-off functions.

6.7 Analog Outputs with Safety-Related Shutdown (F3 AIO 8/4 014)

The remote I/O writes to the analog outputs once per cycle and saves the values internally.

The outputs are not safety-related, but they can be safely switched off together.

To achieve SIL 4, the output values must be read back via safety-related analog inputs and evaluated in the user program. Responses to faulty output values must be programmed in the user program as well.

6.7.1 Test Routines

The remote I/O automatically tests the two safety switches used to shut down all four module outputs during operation.

6.8 Checklist for Safety-Related Outputs

HIMA recommends using the available checklist for engineering, programming and starting up safety-related inputs. The checklist can be used as a planning document and also serves as proof of careful planning.

When engineering or starting up the system, a checklist must be filled out for each of the safety-related output channels used in the system to verify the requirements to be met. This is the only way to ensure that all requirements were considered and clearly recorded. The checklist also provides documentation about the connection between the external wiring and the user program.

The checklist *HIMatrix_Checklist_Outputs.doc* is available as Microsoft© Word© document. The ZIP file *HIMatrix_Checklists.zip* contains all the checklists and can be obtained upon request by sending an e-mail to documentation@hima.com.

7 Software for HIMatrix Systems

The software for the safety-related automation devices of the HIMatrix systems can be divided into the following blocks:

- Operating system.
- User program.
- SILworX programming tool in accordance with IEC 61131-3

The *operating system* is loaded into the central part (CPU) of the controller. HIMA recommends using the latest version valid for the safety-related applications.

The *user program* is created using the SILworX programming tool and contains the application-specific functions to be performed by the automation device. Parameters are also set using SILworX.

The user program is compiled with the code generator and transferred to the non-volatile memory automation device through an Ethernet interface.

7.1 Safety-Related Aspects of the Operating System

Each approved operating system is identified by a unique name. The version number and the CRC signature are given to help distinguish the systems from one another. The valid versions of the operating system and corresponding signatures (CRCs) - approved by the TÜV for use in safety-related automation devices - are subject to a revision control and are documented in a list maintained by HIMA in co-operation with the TÜV.

The current version of the operating system can be read using the SILworX programming tool. A control check performed by the user is required (see Chapter 7.6).

7.2 Operation and Functions of the Operating System

The operating system executes the user program cyclically. In a simplified form, the following functions are performed:

- Reading of the input data.
- Processing of the logic functions, programmed in accordance with IEC 61131-3.
- Writing of the output data.

The following basic functions are also executed:

- Comprehensive self-tests.
- Test of I/O modules during operation.
- Data transmission.
- Diagnostics.

7.3 Safety-Related Aspects of Programming

7.3.1 Safety Concept for the Programming Tool

Safety concept for the SILworX programming tool:

- When the programming tool is installed, a CRC checksum ensures the program package's integrity on the way from the manufacturer to the user.
- The programming tool performs validity checks to reduce the likelihood of faults while entering data.
- Compiling the program twice and comparing the two CRC checksums ensures that data corruption resulting from random faults in the PC in use is detected.
- The programming tool and the measures defined in this safety manual make it sufficiently improbable that a code generated properly from a semantic and syntactic view point can still contain undetected systematic faults resulting from the code generation process.

When starting up a safety-related controller for the first time, a comprehensive functional test must be performed to verify the safety of the entire system.

- Verify that the tasks to be performed by the controller were properly implemented using the data and signal flows.
- Perform a comprehensive functional test of the logic by trial (see Chapter 7.3.2).

If a user program is modified, only the program components affected by the change must be tested. To this end, the safe version comparison in SILworX can be used to determine and display which changes were performed compared to the previous version.

Whenever the safety-related controller is started up, the verification and validation requirements specified in the application standards must be observed!

7.3.2 Verifying the Configuration and the User Program

To verify that the user program created performs the required safety function, the user must create suitable test cases for the required system specification.

An independent test of each loop (consisting of input, the key interconnections in the application and output) is usually sufficient.

Suitable test cases must also be created for the numerical evaluation of formulas. Reasonable are equivalence class tests, which are tests within defined ranges of values, at the limits of or within invalid ranges of values. The test cases must be selected such that the calculations can be proven to be correct. The required number of test cases depends on the formula used and must include critical value pairs.

HIMA recommend performing an active simulation with sources. This will prove that the sensors and actuators in the system, also those connected via remote I/Os, are properly wired. This is also the only way to verify the system configuration.

SILworX can be used as testing aid for:

- Checking inputs.
- Forcing outputs.

This procedure must be followed both when initially creating the user program and when modifying it.

7.3.3 Archiving a Project

HIMA recommends archiving the project whenever the program is loaded into the controller.

SILworX creates a project in a project file. This must be suitably stored, e.g., on a storage medium.

i If the project file is deleted or corrupted, access to the PES program and variables is no longer possible!

For this reason, HIMA strongly recommends archiving and storing the files on additional data media!

7.3.4 Options for Identifying the Program and the Configuration

The user programs are unambiguously identified with the configuration CRC of the project. This configuration CRC can be compared to the configuration CRC of the loaded projects.

To ensure that the saved project file remained unchanged, compile the corresponding resource and compare the configuration CRC with the CRC of the loaded configuration. This CRC can be displayed with SILworX.

i Perform a comprehensive functional test when starting up a safety-related controller for the first time or after modifying the user program.

Create a project archive.

7.4 Resource Parameters

⚠ WARNING

Physical injury possible due to defective configuration!

Neither the programming tool nor the controller can verify certain project-specific parameters. For this reason, enter these parameters correctly in the programming tool and verify the whole entry upon completion of the PES load from within the PES itself.

These parameters are:

- **Rack ID, see HIMatrix system manual (HI 800 141 E).**
- **The parameters marked as safety parameters in Table 16.**

The following parameters are defined in the programming tool for actions permitted during the automation device's safety-related operation and are referred to as safety-related parameters.

Settings that may be defined for safety-related operation are not firmly bound to any specific requirement classes. Instead, each of these must be agreed upon together with the competent test authority for each separate implementation of the controller.

7.4.1 System Parameters of the Resource

The system parameters of the resource determine how the controller will behave during operation and can be set in SILworX, in the *Properties* dialog box of the resource. All the parameters can also be changed online, except for *Minimum Configuration Version*.

Parameter	S [1)]	Description	Setting for safe operation
Name	-	Name of the resource.	User-defined
System ID [SRS]	Y	System ID of the resource: 1...65 535 Default value: 60 000 The value assigned to the system ID must differ from the default value, otherwise the project is not able to run!	Unique value within the controller network. This network includes all controllers that can potentially be interconnected.
Safety Time [ms]	Y	Safety time in milliseconds: 20...22 500 ms Default value: 600 ms for controllers, 400 ms for remote I/Os	Application-specific
Watchdog Time [ms]	Y	Watchdog time in milliseconds: 4...5000 ms Default value: 200 ms for controllers, 100 ms for remote I/Os.	Application-specific
Target Cycle Time [ms]	N	Targeted or maximum cycle time, see *Target Cycle Time Mode*: 0...5000 ms. The maximum target cycle time value may not exceed the value resulting from: configured *Watchdog Time [ms]* - minimum value that can be set for *Watchdog Time [ms]*. Otherwise, it is rejected by the PES. Default value: 0 ms If the default value 0 ms is set, the target cycle time is not taken into account, see Chapter 7.4.1.1	Application-specific
Target Cycle Time Mode	N	Use of *Target Cycle Time [ms],* see Chapter 7.4.1.1 Default value: *Fixed-tolerant*	Application-specific
Multitasking Mode	N	Mode 1 The duration of a CPU cycle is based on the required execution time of all user programs. Mode 2 The processor provides the execution time portion not needed by lower priority user programs to higher priority user programs. Operation mode for high availability. Mode 3 The processor waits until the execution time not needed by the user programs has expired, thus increasing the cycle.	Application-specific
Max.Com. Time Slice ASYNC [ms]	N	Highest value in ms for the time slice used for communication during a resource cycle, see the communication manual (HI 801 101 E): 2...5000 ms Default value: 60 ms	Application-specific
Max. Duration of Configuration Connections [ms]	N	It defines how much time within a CPU cycle is available for processing the configuration connections: 2...3500 ms, see Chapter 7.4.1.2 Default value: 12 ms	Application-specific
Maximum System Bus Latency [µs]	N	Not applicable for HIMatrix controllers! Default value: 0 µs	-

Parameter	S [1)]	Description	Setting for safe operation
Allow Online Settings	Y	Indicates whether the following parameters and online functions can be changed online using the PADT: ▪ *System ID* ▪ *Autostart* ▪ *Global Forcing Allowed* ▪ *Global Force Timeout Reaction* ▪ *Load Allowed* ▪ *Reload Allowed* ▪ *Start Allowed* ▪ *Multitasking Mode* ▪ *Fast Start-Up* ▪ Set Mono/Redundancy Operation ▪ Module: Set Date/Time ▪ All parameters of the user program Refer to Chapter 7.4.1.3 for a description of the impact on additional parameters.	OFF is recommended
		ON: The parameters and online functions can be changed online.	
		OFF: The parameters and online functions cannot be changed online.	
		i *Allow Online Settings* can only be set to ON via reload or if the PES is stopped.	
		Default value: ON.	
Autostart	Y	The user program starts automatically in the following cases: ▪ After connection of the supply voltage. ▪ After PES restart due to errors.	Application-specific
		ON: The user program starts automatically.	
		OFF: The user program does not start automatically.	
		Default value: OFF	
Start Allowed	Y	ON: Cold start or warm start permitted with the PADT in RUN or STOP.	Application-specific
		OFF: Start not allowed.	
		Default value: ON	
Load Allowed	Y	ON: Configuration download is allowed.	Application-specific
		OFF: Configuration download is not allowed.	
		Default value: ON	
Reload Allowed	Y	Indicates if a configuration reload is possible. Refer to Chapter 7.4.1.3 for a description of the impact on additional parameters.	Application-specific
		ON: Configuration reload is allowed.	
		OFF: Configuration reload is not allowed. A running reload process is not aborted when switching to OFF.	
Global Forcing Allowed	Y	ON: Global forcing is permitted for this resource.	Application-specific
		OFF: Global forcing is not permitted for this resource.	
		Default value: ON	
Global Force Timeout Reaction	N	Specifies how the resource should behave when the global force timeout has expired. The following settings are possible: ▪ *Stop Forcing Only* ▪ *Stop Resource* Default value: *Stop Forcing*	Application-specific

Parameter	S [1]	Description	Setting for safe operation
Minimum Configuration Version	N	With this setting, code that is compatible with previous or newer CPU operating system versions in accordance with the project requirements, may be generated. The generated code corresponds to the code that was generated by the specified SILworX version, see Chapter 7.4.1.4. Default value: SILworX V8 for new projects	Application-specific
		SILworX V2, SILworX V3: Not applicable for HIMatrix controllers!	
		SILworX V4, SILworX V5, SILworX V6: The code is generated like in SILworX V4, V5 or V6.48. The generated code is compatible with the CPU operating system V8.	
		SILworX V6b: The code is generated like in SILworX V6,114. The generated code is compatible with the CPU operating system V10.	
		SILworX V7: The code is generated like in SILworX V7. The generated code is compatible with the CPU operating system V11.	
		SILworX V8: The code is generated like in SILworX V8. The generated code is compatible with the CPU operating system V12.	
Fast Start-Up	Y	After connecting the supply voltage, the resource starts up faster, < 10 s, see Chapter 7.4.1.5 Default value: OFF	Application-specific
[1] Safety-related system parameter yes/no (Y/N)			

Table 16: Resource System Parameters

7.4.1.1 Use of the Parameters *Target Cycle Time* and *Target Cycle Time Mode*

These parameters can be used to maintain the cycle time constant to a value that is as close as possible to *Target Cycle Time [ms]*. To do this, this parameter must be set to a value > 0. HIMatrix then limits the reload task to ensure that the target cycle time is maintained.

The following table describes the effect of the target cycle time mode.

Target Cycle Time Mode	Effect on user programs	Effect on reload of processor modules
Fixed	The PES maintains the target cycle time and extends the cycle, if required. If the processing time of the user programs exceeds the target cycle time, the cycle duration is increased.	Reload is not processed if the target cycle time is not sufficient.
Fixed-tolerant		A maximum of every fifth cycle is extended to allow reload.
Dynamic-tolerant	HIMatrix executes the cycle as quickly as possible.	A maximum of every fifth cycle is extended to allow reload.
Dynamic		Reload is not processed if the target cycle time is not sufficient.

Table 17: Effect of Target Cycle Time Mode

7.4.1.2 Calculating the *Maximum Duration of Configuration Connections [ms]*

If communication for the application software is not completely processed within a CPU cycle, it is resumed in the next following CPU cycle at the interruption point.

This slows down communication to remote I/Os and PADTs, but it also ensures that all connections to external partners are processed equally and completely.

Suitable value: Select the value such that the cyclic processor tasks can be executed within the time resulting from *Watchdog Time - Max. Duration of Configuration Connections*.

The volume of the configuration data to be communicated depends on the number of configured remote I/Os, the existing connections to PADTs and the system modules with an Ethernet interface.

A first setting can be calculated as follows:

$T_{Config} = (n_{Com} + n_{RIO} + n_{PADT}) * 0.25\ ms + 2\ ms$, where

T_{Config}	System parameter *Max. Duration of Configuration Connections [ms]*
n_{Com}	Number of modules with Ethernet interfaces {CPU, COM}
n_{RIO}	Number of configured remote I/Os
n_{PADT}	Maximum number of PADT connections = 5

The calculated time can either be modified in the properties of the resource or directly online based on the figure gathered in the online statistics.

When generating the code or converting the project, a warning message is displayed in the PADT if the value defined for *Max. Duration of Configuration Connections* is less than the value resulting from the previous equation.

i If *Max. Duration of Configuration Connections* is set too low, communication between PADT and PES runs very slow and may even fail!

7.4.1.3 Notes on *Allow Online Settings* and *Reload Allowed*

The following parameters cannot be changed online if *Allow Online Settings* **and** *Reload Allowed* are set to OFF:

- *Watchdog Time [ms]*
- *Safety Time [ms]*
- *Target Cycle Time [ms]*
- *Target Cycle Time Mode*

If *Allow Online Settings* or *Reload Allowed* is set to ON, the parameters can be changed online.

7.4.1.4 Notes on *Minimum Configuration Version*

- In a new project, the latest *Minimum Configuration Version* is selected. Verify that this setting is in accordance with the hardware and operating system version in use.
- In a project converted from a previous SILworX version, the value for *Minimum Configuration Version* remains the value set in the previous version. This ensures that the configuration CRC resulting from the code generation is the same as in the previous version and the generated configuration is still compatible with the operating systems in the hardware.

  For this reason, the value of *Minimum Configuration Version* should only be changed in connection with other changes performed to the affected resource.
- If features only available in higher configuration versions are used in the project, SILworX automatically generates a configuration version higher than the preset *Minimum Configuration Version*. This is indicated by SILworX at the end of the code generation. The resource rejects to load a higher configuration version that does not match its operating system.

  To remove such incompatibilities, it can be helpful to compare the information provided by the version comparison with the overview of the module data.

7.4.1.5 Notes on *Fast Start-Up*

This parameter exists for SILworX V7 and higher, and requires a resource with CPU operating system V11 or higher and a COM operating system V16 or higher. Additionally, the resource must be equipped with a CPU bootloader V11.2 or higher and a COM bootloader V16.8 or higher. The bootloader is not the same as the OS loader (emergency loader) and cannot be replaced by the user.

Fast start-up is only effective when the PES supply voltage is connected. Operation at SIL 3 level is still ensured.

Fast start-up is achieved through:

- Reduced self-test.
- No detection of duplicate IP addresses.

  If detection of duplicate IP addresses is deactivated and the network configuration is faulty, duplicate IP addresses might be in use in the network!

  The parameter settings must ensure that no duplicate IP addresses exist in the network!

If an LED test is required during start-up, the parameter *Fast Start-Up* must be set to OFF!

7.4.2 Hardware System Variables

These variables are used to change the behavior of the controller while it is operating in specific states. These variables can be set in the hardware detail view of the SILworX Hardware Editor.

Variable	S [1]	Function	Default setting	Setting for safe operation
Force Deactivation	Y	To prevent or immediately stop forcing.	FALSE	Application-specific
Spare 0...Spare 16	N	No function	-	-
Emergency Stop 1 ... Emergency Stop 4	Y	To shut down the controller if faults are detected by the user program.	FALSE	Application-specific
Relay Contact 1... Relay Contact 4	N	Only applicable to F60! OR-linked system variables that control the relay of the FAULT contact on the F60 PS 01. The relay is a change-over contact with common contact 2, break contact 3 and make contact 1. ▪ If the F60 module is in the RUN state and the system variables *Relay Contact 1...4* are FALSE, contact 1-2 is closed (contact 2-3 is open). ▪ If the F60 module is in the RUN state and no global variables are connected to the system variables *Relay Contact 1…4*, contact 1-2 is closed (contact 2-3 is open). ▪ If the F60 module is in the RUN state and at least one of the system variables *Relay Contact 1…4* is TRUE, contact 1-2 is open (contact 2-3 is closed). ▪ If the F60 module is not in the RUN state, contact 1-2 is open (contact 2-3 is closed). ▪ If the F60 module is de-energized, contact 1-2 is open (contact 2-3 is closed).	-	Application-specific
Read-only in RUN	Y	After starting the controller, the access permissions are downgraded to *Read-Only*. Exceptions are forcing and reload.	FALSE	Application-specific
Reload Deactivation	Y	To preclude that the controller is loaded through a reload.	FALSE	Application-specific
User LED 1, User LED 2	N	Applicable only for special controllers! Controls the corresponding LED, if existing.	0	-
[1] Safety-related system parameter yes/no (Y/N)				

Table 18: Hardware System Variables

Global variables can be connected to these system variables; the value of the global variables is modified using a physical input or the user program logic.

Example: A key switch is connected to a digital input. The digital input is assigned to a global variable associated with the system variable *Read-Only in Run*. The key owner can thus activate or deactivate the operating actions *Stop*, *Start* and *Download*.

7.5 Protection Against Manipulation

Together with the responsible test authority, the user must define which measures should be implemented to protect the system against manipulation.

Protective mechanisms for preventing unintentional or unapproved modifications to the safety system are integrated into the PES and the SILworX programming tool:

- Each change to the user program or configuration results in a new CRC. These changes can only be transferred to the PES via download or reload.
- The operating options depend on the rights of the user logged into the PES.
- The SILworX programming tool prompts the user to enter a password in order to log in to the PES.
- No connection between PADT and PES is required in RUN.

All requirements for protection against manipulation specified in the safety and application standards must be met. The operator is responsible for authorizing employees and implementing the required protective actions.

⚠ WARNING

Danger: Physical injury due to unauthorized manipulation of the controller!

The controller must be protected against unauthorized access!

For instance:

- **Changing the default settings for login and password!**
- **Controlling the physical access to the controller and PADT!**

PES data can only be accessed if the PADT in use is operating with the current version of the SILworX programming tool and the user project is available in the currently running version (archive maintenance!).

The user only need to connect the PADT to the PES when loading the user program or performing diagnostics. The PADT is not required during normal operation. Disconnecting the PADT and PES during normal operation protects against unauthorized access.

7.6 Checklist for Creating a User Program

To comply with all safety-related aspects during the programming phase, HIMA recommends using the following checklist prior to and after loading a new or modified program. The checklist can be used as a planning document and also serves as proof of careful planning.

The checklist *HIMatrix_Checklist_Program.doc* is available as Microsoft© Word© document. The ZIP file *HIMatrix_Checklists.zip* contains all the checklists and can be obtained upon request by sending an e-mail to documentation@hima.com.

8 Safety-related aspects of the user program

General sequence for programming HIMatrix automation devices for safety-related applications:

- Specify the controller functionality.
- Write the user program.
- Compile the user program using the C-code generator.
- Compile the user program a second time and compare the resulting CRCs.
- The program is error-free and able to run.
- Verify and validate the user program.

The PES can start safety-related operation.

8.1 Scope for Safety-Related Use

(Refer to Chapter 3.3 for more details about specifications, rules and explications to safety requirements).

The user program must be written using the SILworX programming tool. For further details on the operating system released for personal computers, refer to the release documentation for the SILworX version to be used.

The programming tool includes the following functions:

- Input (FBD Editor), monitoring and documentation.
- Variables with symbolic names and data types (BOOL, UINT, etc.).
- Assignment of the HIMatrix controllers (Hardware Editor).
- Code generator (for translating the user program into a machine code).
- Configuration of communication.

8.1.1 Programming Basics

The tasks to be performed by the controller must be defined in a specification or a requirements specification. This documentation serves as the basis for checking its proper implementation in the user program. The specification format depends on the tasks to be performed. These include:

- Combinational logic
  - Cause/effect diagram.
  - Logic of the connection with functions and function blocks.
  - Function blocks with specified characteristics.
- Sequential controllers (sequence control system)
  - Written description of the steps and their enabling conditions and of the external components to be controlled.
  - Flow charts.
  - Matrix or table form of the step enabling conditions and the external components to be controlled.
  - Definition of constraints, e.g., operating modes, emergency stop, etc.

The I/O concept of the system must include the analysis of the field circuits, i.e., the type of external components:

- External components (field devices)
  - Input signals during normal operation ("de-energize to trip" principle with digital field devices)
  - Input signals in the event of a fault
  - Definition of safety-related redundancies required for safety (1oo2, 2oo3).
  - Monitoring of discrepancy and response.

- Positioning and activation during normal operation.
- Safe response/positioning at shutdown or after power loss.

Programming goals for user program:

- Easy to understand.
- Easy to trace and follow.
- Easy to modify.
- Easy to test.

8.1.2 Functions of the User Program

Programming is not subject to hardware restrictions. The user program functions can be freely programmed.

When programming, account for the de-energize to trip principle for the physical inputs and outputs. Only elements complying with IEC 61131-3 together with their functional requirements are used within the logic.

- The physical inputs and outputs usually operate in accordance with the de-energize to trip principle, i.e., their safe state is *0*.
- The user program includes meaningful logic and/or arithmetic functions irrespective of the de-energize to trip principle of the physical inputs and outputs.
- The program logic should be clear and easy to understand and well documented to assist in debugging. This includes the use of functional diagrams.
- To simplify the logic, the inputs and outputs of all function blocks and variables can be inverted in any given order.
- The programmer must evaluate the fault signals from the inputs/outputs or from logic blocks.

HIMA recommends encapsulating functions to user-specific function blocks and functions based on standard functions. This ensures that a user program can be clearly structured in modules (functions, function blocks). Each module can be viewed and tested on an individual basis. By grouping smaller modules into larger ones and then all together into a single user program, the user is effectively creating a comprehensive, complex function.

8.1.3 Variable Declaration

A variable is a placeholder for a value within the program logic. The variable name is used to symbolically address the storage space containing the stored value. A variable is created in the variable declaration for the program or function block.

Two essential advantages result from using symbolic names instead of physical addresses:

- The plant denominations of inputs and outputs can be used in the user program.
- The modification of how the signals are assigned to the input and output channels does not affect the user program.

Variables with invalid source, e.g., due to a hardware fault in a physical input, adopt the configured initial value. If no configured initial value exists, the variables have the initial standard initial value 0 or FALSE.

⚠ CAUTION

Reciprocal influence of user program parts is possible!

The use of the same global variables in several user program parts such as function blocks or functions can lead to a variety of unintentional consequences caused by the reciprocal influence among the user program parts.

- **Carefully plan the use of the same global variables in several user program parts.**
- **Use the cross-references in SILworX to check the use of global data. Global data may only be assigned values by one entity, either within a user program component, through communication or by the hardware!**

8.1.4 Factory Acceptance Test and Test Authority

HIMA recommends involving the test authority as soon as possible when designing a system that is subject to approval.

8.2 Procedures

This chapter describes the procedures typically used for developing the user programs for safety-related HIMatrix controllers.

8.2.1 Assigning Variables to Inputs or Outputs

The required test routines for safety-related I/O devices, I/O modules or I/O channels are automatically executed by the operating system.

To assign a variable to an I/O channel

1. Define a global variable of a suitable type.
2. Enter an appropriate initial value, when defining the global variable.
3. Assign the global variable the channel value of the I/O channel.
4. In the user program, evaluate the error code *-> Error Code [Byte]* and program a safety-related response.

► The global variable is associated with an input/output channel.

8.2.2 Locking and Unlocking the Controller

Locking the controller locks all functions and prevents users from accessing them during operation. This also protects against manipulations to the user program. The locking extent should be considered in connection with the safety requirements for the PES application, and can also be agreed upon with the test authority responsible for the factory acceptance test (FAT).

Unlocking the controller deactivates any locks previously set (e.g., to perform work on the controller).

i The locking and unlocking functions are only available with controllers and the F3 DIO 20/8 014 remote I/O, but not with the remaining remote I/Os!

Three system variables serve for locking:

Variable	Function
Read-only in RUN	TRUE: Starting, stopping, and downloading the controller are locked. FALSE: Starting, stopping, and downloading the controller are possible.
Reload Deactivation	TRUE: Reload is locked. FALSE: Reload is possible.
Force Deactivation	TRUE: Forcing is deactivated. FALSE: Forcing is possible.

Table 19: System Variables for Locking and Unlocking the PES

If all three system variables are TRUE: no write access to the controller is possible.

Example of Locking and Unlocking the PES

To make a controller lockable

1. Define a global variable of type BOOL and set its initial value to FALSE.
2. Assign global variables to the three system variables *Read only in Run*, *Reload Deactivation* and *Force Deactivation*.
3. Assign the global variable to the channel value of a digital input.
4. Connect a key switch to the digital input.
5. Compile the program, load it into the controller, and start it.

► The owner of a corresponding key is able to lock and unlock the controller. If the corresponding digital input device or input module fails, the controller is unlocked.

This simple example can be modified using multiple global variables, digital inputs and key switches so that the permissions for forcing, reload, stop, start and download can be distributed on different keys and persons.

8.2.3 Code Generation

The code is generated after entering the complete user program and the I/O assignments of the controller. During these steps, the configuration CRC, i.e., the checksum for the configuration files, is created.

This is a signature for the entire configuration that is issued as a 32-bit, hexadecimal code. It includes all of the configurable or modifiable elements such as the logic, variables or switch parameter settings.

i Before loading a user program for safety-related operation, the user program must be first compiled twice. The two generated versions must have the same checksum.

By default, SILworX automatically compiles the resource configuration twice and compares the checksums.

The result of the CRC comparison is displayed in the Logbook.

By compiling the user program twice and comparing the checksums of the generated code, the user can detect potential corruptions of the user program resulting from random faults in the hardware or operating system of the PC in use.

8.2.4 Safe Version Comparison

The safe SILworX version comparison compares the following resource configuration types with one another:

- Resource configuration loaded into the controller.
- Resource configuration existing in the PADT.

- Exported (archived) resource configuration.

The comparison result achieves SIL 3, since it is derived from loadable files and includes the CRCs.

To verify the program changes, the safe version comparison must be started **before** the program is loaded to the controller.

It exactly determines the changed parts of the resource configuration. This, in turn, facilitates testing the changes and identifying the test data, and may be submitted to the inspection authority as proof of the change.

Structured programming, and the use of significant names from the first configuration version on, facilitate understanding of the comparison result.

For details on the safe version comparison, refer to the corresponding manual (HI 801 286 E).

8.2.5 Loading and Starting the User Program

The configuration can only be loaded into the PES of the HIMax system by performing a download if it has been set to the STOP state beforehand.

A load process includes all user programs of the project configuration. The system monitors that the user program is loaded completely. Afterwards, the user program can be started, i.e., the routine begins to be processed in cycles.

i HIMA recommends performing a project data backup, e.g., on an external removable medium, after loading a user program into the controller.

This is done to ensure that the project data corresponding to the configuration loaded into the controller remains available even if the PADT fails.

HIMA recommends performing a data backup on a regular basis also independently from loading the program.

8.2.6 Reload

The use of reload for changing the resource configuration must be agreed upon with the competent test authority!

If user programs were modified, the changes can be transferred to the PES during operation. After being tested by the operating system, the modified user program is activated and assumes the control task.

i **Observe the following points when reloading sequence chains:**

The reload information for sequence chains does not take the current sequence status into account. The sequence chain can therefore be changed and set to an undefined state by performing a reload. The user is responsible for this action.

Examples:

- Deleting the active step. As a result, no chain step has the *active* state.
- Renaming the initial step while another step is active.
  As a result, a sequence has two active steps!

Observe the following points when reloading actions:

During the reload, actions are loaded with their complete data. All potential consequences must be carefully analyzed prior to performing a reload.

Examples:

- If a timer action qualifier is deleted due to the reload, the timer expires immediately. Depending on the remaining settings, the *Q* output can therefore be set to TRUE.
- If the status action qualifier (e.g., the *S* action qualifier) is deleted for a set element, the element remains set.
- Removing a *P0* action qualifier set to TRUE actuates the trigger function.

Prior to performing a reload, the operating system checks if the required additional tasks would increase the cycle time of the current user programs to such an extent that the defined watchdog time is exceeded. In this case, the reload process is aborted with an error message and the controller continues operation with the previous project configuration.

The controller can abort a running reload process.

A successful reload is ensured by planning a sufficient reserve for the reload when determining the watchdog time or temporarily increasing the controller watchdog time by a reserve.

Any temporary increases in the watchdog time must be agreed upon with the competent test authority.

Exceeding the target cycle time can also lead to an abort of the reload.

The reload can only be performed if the *Reload Allowed* system parameter is set to ON and the *Reload Deactivation* system variable is set to OFF.

The user is responsible for ensuring that the watchdog time includes a sufficient reserve time. This should allow the user to manage the following situations:

- Variations in the user program's cycle time.
- Sudden, strong cycle loads, e.g., due to communication.
- Expiration of time limits during communication.

The use of reload requires a license. Refer to the HIMatrix system manual (HI 800 141 E) for further details on the reload process.

8.2.7 Forcing

Forcing is the procedure by which a variable's current value is replaced with a force value. The variable receives its current value from a physical input, communication or a logic operation. If the variable is forced, its value no longer depends on the process, but is defined by the user.

⚠ WARNING

Failure of safety-related operation possible due to forced values!

- **Forced value may lead to incorrect output values.**
- **Forcing prolongs the cycle time. This can cause the watchdog time to be exceeded.**

Forcing is only permitted after receiving consent from the test authority responsible for the acceptance test.

When forcing values, the person in charge must take further technical and organizational measures to ensure that the process is sufficiently monitored in terms of safety. HIMA recommends setting a time limit for the forcing procedure.

For more details on forcing, refer to the HIMatrix system manual (HI 800 141 E).

8.2.7.1 Forcing of Data Sources

Changing the assignment of a forced global variable to one of the following data sources can lead to unexpected results:

- Physical inputs
- Communication protocols
- System variables

The following sequence of actions causes a variable to be unintentionally forced:

1. A forced global variable A is assigned to one of the specified data sources. This indeed causes the data source to be forced!
2. The assignment is removed. The data source maintains the property *Forced*.
3. Another global variable B is assigned to the data source.
4. A reload is performed to load the project change into the PES.

As a result, the **newly assigned** variable B is forced even though this was not intended!

Workaround: First stop forcing variable A.

The channel view of the Force Editor shows which channels have been forced.

Global variables having the user program as data source retain the *forced* setting even when the assignment is changed.

8.2.8 Changing the System Parameters during Operation

Some system parameters may be changed during operation (online). A typical application case is the temporary increase of the watchdog time to be able to perform a reload.

Prior to using an online command to set parameters, make sure that this change will not result in a safety-critical state. If required, organizational and/or technical measures must be taken to prevent any damage. The application standards must be observed!

The safety time and watchdog time values must be checked and compared to the safety time required by the application and to the actual cycle time. These values cannot be verified by the PES!

The controller ensures that the watchdog time is not set to a value less than the watchdog time value of the configuration loaded in the PES.

The following parameter settings can be changed online.

- System ID
- Resource Watchdog Time
- Safety Time
- Target Cycle Time
- Target Cycle Time Mode
- Allow Online Settings
- Autostart
- Start Allowed
- Load Allowed
- Reload Allowed
- Global Forcing Allowed
- Global Force Timeout Response

Allow Online Settings allows enables the user to change the remaining parameters. *Allow Online Settings* can be set to TRUE in the STOP state.

System parameters may also be changed during operation by performing a reload.

8.2.9 Project Documentation for Safety-Related Applications

The SILworX programming tool allows the user to automatically print the documentation for a project. The most important document types include:

- Interface declaration
- Signal list
- Logic
- Description of data types
- Configurations for system, modules and system parameters
- Network configuration
- List of signal cross-references
- Code generator details

This documentation is required for the factory acceptance test (FAT) of a system subject to approval by a test authority (e.g., TÜV). The factory acceptance test (FAT) only applies to the user functionality, but not to the safety-related modules and automation devices of the HIMatrix system that have already been approved.

8.2.10 Multitasking

Multitasking refers to the capability of the HIMatrix systems to process up to 32 user programs within the processor system.

The individual user programs can be started and stopped independently from one another.

A user program cycle can take multiple processor system cycles. This can be controlled with the resource and user program parameters. SILworX uses these parameters to calculate the user program watchdog time:

$$\text{Watchdog Time}_{\text{User program}} = \textit{Watchdog Time}_{\text{Processor module}} * \textit{Maximum Number of Cycles}$$

The individual user programs operate in an interference-free manner and independently from one another. However, reciprocal influence can be caused by:

- Use of the same global variables in several user programs.
- Unpredictably long runtimes can occur in individual user programs if no limit is configured with *Max Duration for Each Cycle*.
- The distribution of user program cycles over processor module cycles strongly affects the user program response time and the response time of the variables written by the user program!
- A user program evaluates global variables written to by another user program up to at least as many processor module cycles as the value defined in the program's system parameter *Program's Maximum Number of CPU Cycles*. In the worst case, the following sequence is possible:
  - Program A writes global variables needed by program B.
  - Program A stops its cycle in the same processor system cycle in which program B starts its cycle.
  - Program B is only able to read the values written to by program A when its next cycle starts.
  - The duration of the cycle just started by program B can be *Program's Maximum Number of CPU Cycles * Cycle Time*. Only at this point, program B adopts the values written to by program A.
  - It may take more than the configured *Program's Maximum Number of CPU Cycles* of the processor system until B reacts to these values!

⚠ CAUTION

Reciprocal influence of user programs is possible!

The use of the same global variables in several user programs can lead to a variety of consequences caused by the reciprocal influence among the user programs.

- **Carefully plan the use of the same global variables in several user programs.**
- **Use the cross-references in SILworX to check the use of global data. Global data may only be assigned values by one entity, either within a user program, from safety-related inputs or through safety-related communication protocols!**

The user is responsible for ensuring that operation is not disturbed by a reciprocal influence of the user programs!

For details on multitasking, refer to the HIMatrix system manual (HI 800 141 E).

8.2.11 Factory Acceptance Test and Test Authority

HIMA recommends involving the test authority as soon as possible when designing a system that is subject to approval.

The factory acceptance test (FAT) only applies to the user functionality, but not to the safety-related modules and devices of the HIMatrix system that have already been approved.

9 Configuring Communication

In addition to physical input and output variables, variables can also be exchanged with another system via a data connection. In this case, the variables are declared with SILworX, in the Protocols area of the corresponding resource.

This data exchange can occur in either read-only or read/write mode.

9.1 Standard Protocols

Many communication protocols only ensure a non-safety-related data transmission. These protocols can be used for the non-safety-related aspects of an automation task.

⚠ WARNING

Physical injury due to usage of unsafe import data!

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

Depending on the controller variant, the following standard protocols are available:

- SNTP
- Send/Receive TCP
- Modbus (master/slave)
- PROFIBUS DP (master/slave)
- PROFINET and PROFIsafe (CPU BS V7 and higher)

All standard protocols are interference-free on the safe processor system.

9.2 Safety-Related Protocol: safe**ethernet**

For safety-related data exchange between components safe**ethernet** must be used.

As a HIMatrix system component, safe**ethernet** is certified up to SIL 4.

Use the safe**ethernet** Editor or P2P Editor to configure how safety-related communication is monitored.

For determining the safe**ethernet** parameters *Receive Timeout* and *Response Time*, 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.

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

NOTICE

The safe state may be entered inadvertently

***Receive Timeout* is a safety-related parameter!**

If all values must be transferred, the value of a signal must either be present for longer than *Receive Timeout* or it must be monitored using a loop back.

9.2.1 Receive Timeout

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

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

Since *Receive Timeout* is a safety-relevant component of the worst case response time T_R (see Chapter 3.2.4 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 A wanted fault tolerance of communication can be achieved by increasing the value of *Receive Timeout*, provided that this is permissible in terms of time for the application process (worst case response time).

NOTICE

The maximum value permitted for *Receive Timeout* depends on the application process and is configured in the safe**ethernet Editor, along with the expected maximum response time and the profile.**

9.2.2 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 the safe**ethernet** protocol, the **Response Time** expected to result from the physical conditions of the transport path must be set and a suitable safe**ethernet** profile must be selected.

The preset *Response Time* affects the configuration of all the safe**ethernet** 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:

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

If this condition is met, the loss of at least one data packet can be intercepted without interrupting the safe**ethernet** /P2P 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 Make sure that the communication system complies with the configured response time!

If this condition cannot always be ensured, a corresponding connection system variable for monitoring the response time is available. If more than on occasion the measured response time exceeds the receive timeout by more than a half, the configured response time must be increased.

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

NOTICE

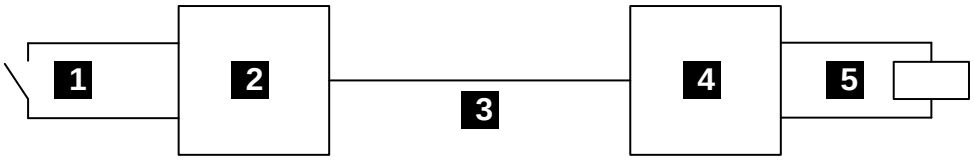
In the following examples, the formulas for calculating the worst case response time only apply for a connection with HIMatrix controllers if the parameter

safety time = 2 * watchdog time

has been set in the systems.

9.2.3 Calculating the Worst Case Response Time

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



1 Input
2 Controller 1
3 Safety-related protocol
4 Controller 2
5 Output

Figure 4: Response Time with Interconnection of two HIMatrix Controllers

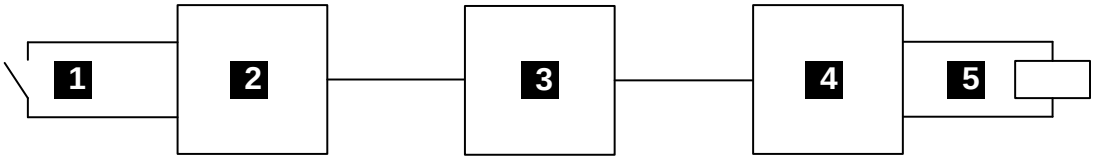
$$T_R = t_1 + t_2 + t_3$$

T_R	Worst case response rime
t_1	2 ∗ watchdog time of controller 1.
t_2	Receive timeout
t_3	2 ∗ watchdog time of controller 2

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

9.2.4 Calculating the Worst Case Response Time with 2 Remote I/Os

The worst case response time T_R is the time between a change in a field component input signal (in) of the first remote I/O module and the response on the corresponding output (out) of the second remote I/O module. It can be calculated as follows:



1 Input
2 Remote I/O 1
3 Controller
4 Remote I/O 2
5 Output

Figure 5: 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 time of remote I/O 1
t_2	Receive timeout$_1$
t_3	2 ∗ watchdog time of the controller
t_4	Receive timeout$_2$
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 controller is used instead of a remote I/O.

9.2.5 Terms

Receive timeout	Monitoring time of controller 1 within which a correct response from controller 2 must be received. Once the time has expired, safety-related communication is terminated.
Receive $timeout_1$	Remote I/O 1 → controller
Receive $timeout_2$	Controller → remote I/O 2
Watchdog time	Maximum permissible duration of a PES RUN cycle (cycle time).
Worst case	The worst case response time is the time between a change in a physical input (in) signal of controller 1 and a response in the corresponding output (out) of controller 2.

The data are transferred using a safety-related protocol.

9.2.6 Assigning safe**ethernet** Addresses

Take the following points into account when assigning network addresses (IP addresses) for safe**ethernet**:

- The addresses must be unique within the network used.
- When connecting safeethernet to another network (company-internal LAN, etc.), make sure that no disturbances may occur. Potential sources of disturbances include:
  - Data traffic.
  - Coupling with other networks (e.g., Internet).

In these cases, implement suitable measures to counteract against such disturbances using Ethernet switches, firewall and similar.

---

i The operator is responsible for ensuring that the Ethernet used for safe**ethernet** communication or peer-to-peer communication 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.

Appendix

Glossary

Term	Description
AI	Analog input
AO	Analog output
ARP	Address resolution protocol, network protocol for assigning the network addresses to hardware addresses
COM	Communication module
CRC	Cyclic redundancy check
DI	Digital input
DO	Digital output
EMC	Electromagnetic compatibility
EN	European standard
ESD	Electrostatic discharge
FB	Fieldbus
FBD	Function block diagrams
HW	Hardware
ICMP	Internet control message protocol, network protocol for status or error messages
IEC	International electrotechnical commission
Interference-free	Inputs are designed for interference-free operation and can be used in circuits with safety functions
MAC	Media access control address, hardware address of one network connection
PADT	Programming and debugging tool (in accordance with IEC 61131-3), PC with SILworX
PE	Protective earth
PELV	Protective extra low voltage
PES	Programmable electronic system
R	Read, the variable is read out
R/W	Read/Write (column title for system variable type)
r_P	Peak value of a total AC component
SC/OC	Short-circuit/open-circuit
SELV	Safety extra low voltage
SFF	Safe failure fraction, portion of faults that can be safely controlled
SIL	Safety integrity level in accordance with IEC 61508
SILworX	Programming tool
SNTP	Simple network time protocol (RFC 1769)
SRS	System.Rack.Slot, addressing of a module
SW	Software
TMO	Timeout
W	Write, the variable receives a value, e.g., from the user program
WD	Watchdog, device for monitoring the system's correct operation Signal for fault-free process
WDT	Watchdog time

Index of Figures

Index of Tables

Index

MANUAL

Safety manual for railway applications

HI 800 437 E

For further information, please contact:

HIMA Rail Segment Team
Phone: +49 6202 709-411

Or contact our Rail Expert Team: rail@hima.com

Learn more about HIMA solutions for our railway applications online:

https://www.hima.com/en/industries-solutions/rail/

www.hima.com