Panasonic

Safety multi application controller

SFL Series (Ver. 2.50)

Instruction Manual

ME-SFL-06

2013.3 | panasonic.net/id/pidsx/global

Thank you very much for purchasing Panasonic Industrial Devices SUNX's Safety Multi Application Controller, **SFL** series.

This device is module type, multi function controller for safety of machinery.

Please read this instruction manual carefully and thoroughly for the correct and optimum use of this product.

This instruction manual should be kept in a convenient place for quick reference.

This instruction manual describes hardware such as CPU modules (ex. **SFL-CPU-MON-V2**), I/O modules (ex. **SFL-S-STP-E**), POWER modules and BASE (**SFL-BASE**), and programming software "**SFL-SOFT**".

This instruction manual has been written for the following personnel who have undergone suitable training and have knowledge of programmable controller, as well as, safety control systems and standards.

- who are responsible for the introduction of the **SFL**
- who design the system using the **SFL**
- who install and connect the **SFL**
- who manage and operate a plant using the **SFL**

Notes

1) All the contents of this instruction manual are the copyright of the publishers, and may not be reproduced (even extracts) in any form by any electronic or mechanical means (including photocopying, recording, or information storage and retrieval) without permission in writing from the publisher.
2) The contents of this instruction manual may be changed without prior notice for further improvement of the product.
3) Though we have carefully drawn up the contents of this instruction manual, if there are any aspects that are not clear, or any error that you may notice, please contact our Panasonic Industrial Devices SUNX office of the nearest distributor.

Preface

Safety Precautions

1. Use the **SFL** as per its specifications. Do not modify the **SFL** since its functions and capabilities may not be maintained and it may malfunction.
2. This device has been developed / produced for industrial use only.
3. This unit is suitable for indoor use only.
4. Use of the **SFL** under the following conditions or environment is not presupposed. Please consult us if there is no other choice but to use the **SFL** in such an environment.
   1) Operating the **SFL** under conditions and environment not described in this manual.
   2) Using this devise in the following fields: nuclear power control, railroad, aircraft, automobiles, combustion facilities, medical systems, aerospace development, etc.
5. When the **SFL** is to be used for enforcing protection of a person from any danger occurring around an operating machine, the user should satisfy the regulations established by national or regional security committees (Occupational Safety and Health Administration: OSHA, the European Standardization Committee, etc.). Contact the relative organization(s) for details.
6. In case of applying the **SFL** to particular equipment, follow the safety regulations in regard to appropriate usage, mounting (installation), operation and maintenance. The users including the installation operator are responsible for the introduction of the **SFL**.
7. Use the **SFL** by installing suitable protection equipment as a countermeasure for failure, damage, or malfunction of the **SFL**.
8. Before using the **SFL**, check whether the device performs properly with the functions and capabilities as per the design specifications.
9. In case of disposal, dispose the **SFL** as industrial waste.

1. Machine designer, installer, employer and operator

   The machine designer, installer, employer and operator are solely responsible to ensure that all applicable legal requirements relating to the installation and the use in any application are satisfied and all instructions for installation and maintenance contained in the instruction manual are followed.

   Whether the **SFL** functions as intended to and systems including the **SFL** comply with safety regulations depends on the appropriateness of the application, installation, maintenance and operation. The machine designer, installer, employer and operator are solely responsible for these items.

2. Engineer

   The engineer would be a person who is appropriately educated, has widespread knowledge and experience, and can solve various problems, which may arise during work, such as a machine designer, or a person in charge of installation or operation etc.

3. Operator

   The operator should read this instruction manual thoroughly, understand its contents, and perform operations following the procedures described in this manual for the correct operation of the **SFL**.

   In case the **SFL** does not perform properly, the operator should report this to the person in charge and stop the machine operation immediately. The machine must not be operated until correct performance of this been confirmed.

4. Environment

   Do not use a mobile phone or a radiophone near the **SFL**.

   Do not install the **SFL** in the following environments.
   1) Areas with high humidity where condensation is likely to occur
   2) Areas exposed to corrosive or explosive gases
   3) Areas exposed to vibration or shock of levels higher than specified
   4) Areas exposed to contact with water
   5) Areas exposed to too much steam or dust

5. Installation

   This device goes to RUN state in approx. 10 seconds after power ON.

   Timing of Input(s) and output(s) depend on its structure and user's program of the **SFL**. Check all of control system work correctly.

6. Wiring

   Be sure to carry out the wiring in the power supply OFF condition.

   All electrical wiring should conform to the regional electrical regulations and laws. The wiring should be done by engineer(s) having the special electrical knowledge.

   Do not run the sensor cable together with high-voltage lines or power lines or put them together in the same raceway.

7. Maintenance

   When replacement parts are required, always use only genuine supplied replacement parts. If substitute parts from another manufacturer are used, the device may not come to detect, resulting in death or serious injury.

   The periodical inspection of the **SFL** must be performed by an engineer having the special knowledge.

8. Others

   Never modify the **SFL**. Modification may cause the device not to work correctly, resulting in death or serious injury.

Table of Contents

1 Overview

This chapter describes the structure of the instruction manual, the basic mode of operation and the structure of the **SFL** safety multi application controller.

1.1 Preface

We have prepared this document with due care to the best of our knowledge and belief. Despite careful checking the possibility of it containing one or two errors cannot be ruled out, however.

Before reading this instruction manual thoroughly, please read through chapter 2 carefully. Understanding the information and procedures described in this chapter is essential for the correct use of the **SFL** safety multi application controller described in the following. Please always remember that faulty installation or programming can lead to serious injuries / damage to man and machine.

The terms and abbreviations used in this document are standard terms used in electrical engineering insofar as not described in more detail.

The basic operating functions of a computer with Microsoft Windows operating system (as from Windows 2000) are assumed as generally known.

1.2 Definition of terms

The definition of the terms and the abbreviations used most frequently in this document are described below.

Module	In this document a module is described as an individual physical unit which is a part of the **SFL**.
Active module	A module which is actively involved in the change of state of the **SFL** (corresponding to active components in electronics).
Assembly	An assembly consists of several individual modules.
Channel	A physical input or output of a module.
A Contact	Designation of an N/O contact
B Contact	Designation of an N/C contact
User program	The control program of the **SFL** created by the programmer.
PS Programm	Safety user program for CPU module (see chapter 3.7) and Safety Input/Output modules.
PN Programm	User program for non safe (operational) applications in CPU module (see chapter 3.7) used with operational Input/Output modules.
PSV	The PSV (Program Specific Value) is the checksum of the user program.
FirmwareFB	Safe function block inside the **SFL**
FB library	Library of function blocks.
authorized person	Person from group described in chapter 2.1.4.
PL	Performance Level. See DIN EN ISO 13849-1.
SIL	Safety Integrity Level. See DIN EN 61508.

1.3 Structure of the document

This manual is subdivided into 9 chapters. There is an additional table of contents preceding every chapter in which the individual sections are shown separately.

1.3.1 Page lay-out

All pages have the same lay-out in principle.

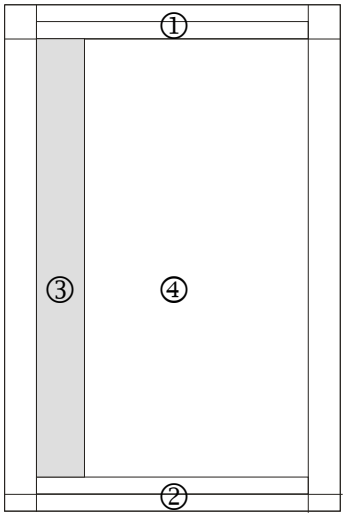


Figure 1-1 Page lay-out of the document

① Headline
The number and the name of the respective chapter are located top right and top left respectively.

② Baseline
The document name, version number and creation date are located bottom left. The page number within the chapter is shown bottom right.

③ Margin
This area contains important information, cross-references to other parts of the text and bibliographic references and the information content of text passages.

④ Texts / Pictures / Tables
Texts, pictures and tables are named within a chapter with the chapter number and an additional index and, if necessary, a brief accompanying text.

1.3.2 Explanation of symbols

Safety warnings

Text passages with a STOP symbol contain important warnings which must be heeded under all circumstances. Failure to heed these warnings may place the **SFL** controller in a state which no longer provides adequate protection for man and/or machine. Please read through such text sections with particular care.

Warnings

A CAUTION sign gives you important information and warnings which guarantee the trouble-free operation of the **SFL** when heeded. Failure to heed these warnings does not restrict the safe operation of the **SFL**.

Information

This symbol indicates useful additional information intended to facilitate the commissioning/servicing of the **SFL** or to provide a deeper insight into the mode of operation of the **SFL** controller.

Cross references

x.x.x / x-xx 

This symbol refers to other text passages within the document which contain additional information. The first digit describes the chapter and the second the page within the chapter.

Additional Information

x.x.x / xx 

This symbol refers to external literature containing additional information on a particular theme. You will find the exact title of the additional literature in the appendix (Chapter 9.5) by means of the number beside the symbol.

1.4 Product description

This device is modular type, safety multi application controller. It serves to monitor and control safety equipment. In connection with safety-orientated sensors/actuators it complies with the safety requirements of category to DIN EN 954-1 or Performance Level (PL) e to DIN EN ISO 13849 –1. It can only be used in systems in which the safe state is synonymous with a de-energized state. It can be used to replace virtually all circuitry so far used in relay technology.

A CPU module can control maximum 15 modules which consists of safety Input/Output modules and non-safety Input/Output modules.(Maximum of 256 channels). But, only safety Input/Output modules can not consist 15 modules (In case of only consists of safety Input/Output module, 14 modules maximum).

The **SFL** is intended for use in control circuits to DIN EN 60204 for a rated voltage of 24V DC. The appropriate shock-hazard protection requirements must be met for the applications.

1.4.1 Overview of the modules

The **SFL** system essentially consists of the following modules:

POWER modul, produces 3.3V DC from 24V DC and provides to CPU and Input/Output modules (**SFL-POWER**, **SFL-BOOSTER**).

CPU module, responsible for processing the user program and controlling the Input/Output modules (**SFL-CPU-OP-MON-V2**, **SFL-CPU-MON-V2**).

Input/Output modules to record/control the states of externally connected sensors/actuators (**SFL-SUB-MON**, **SFL-S-STP-E**, **SFL-S-STP-LC**, **SFL-S-STP-ELC**, **SFL-S-IN-E**, **SFL-S-IN-LC**, **SFL-S-OUT**, **SFL-RELAY**)

Non-Safety Input/Output modules to record/control the states of externally connected sensors/actuators (**SFL-NS-IN**, **SFL-NS-OUT+**)

All the above mentioned modules (with the exception of the back plane) have a redundant structure with two 16 bit microcontrollers monitoring each other.

1.4.2 Brief data

- Can be used in applications up to category 4 to DIN EN 954-1 and PLe to DIN EN ISO 13849-1.
- SFL has 2 operation modes (Mode 1 and Mode 3).
- The operating mode 1 does not require any programming.
- In operating mode 3 programming can be done by ladder diagram (conforming to IEC 61131).
- Programming by means of Windows PC via USB interface
- Safety Inputs/Outputs are available in Mode1 and Mode 3. Operational Inputs/Outputs are available only in Mode 3.
- A maximum of 256 Input/Output channels are available.
- Possibility to store documentation / comments.
- Monitored 24V DC power supply .
- Input/Output modules with self-monitoring.
- Connection facilities:
  Sensor level: EMERGENCY SWITCHING OFF, EMERGENCY STOP, AOPDs, BNS, protective guards, two- hand-operating panels…
  Actuator level: Relays, contactors, signals / indicator lamps...

1.4.3 Area of use

The **SFL** has been developed particularly for decentralized use in medium sized to large plants.

The **SFL** can be used in applications up to category 4 to DIN EN 954-1, Performance Level (PL) e to DIN EN ISO 13849 –1 or the Safety Integrity Level 3 (SIL) to DIN EN 61508 in which the safe state is the deenergized state.

The **SFL** is intended for use in control circuits to DIN EN 60204 for a rated voltage for 24V DC. The appropriate shock-hazard protection requirements are to be met in the applications.

1.5 Mode of operation

1.5.1 Hardware

CPU module and Safety Input/Output modules consists of two systems working independently of each other. Each system is controlled by a microprocessor. The systems monitor each other. In the case of a 2 channel connection, the individual channels are each monitored by a microprocessor. Self-monitoring enables all internal defects to safety-relevant components to be detected within the module. Each module performs a complete self-test at regular intervals.

Non-Safety Input/Output modules consists of one systems. A system is controlled by a microprocessor.

1.5.2 Software

The programmer creates the user program in the form of a ladder diagram (to IEC 61131) using the **SFL-SOFT** programming software. For purposes of verification the ladder diagram is converted to a statement list. After having been checked by the programmer supported by the programming software the user program is translated into a format which can be read by the processor module and can then be transferred to the processor module via a USB interface.

After successful transfer to the processor module the latter sends the user program back to the programming software for checking. After comparing it with the transmitted program the user program is released for operation..

During ongoing operation the processor module compares its data with the states of the Input modules and reacts with the appropriate instructions for the Output modules, depending on the stored user program.

CPU module and Safety Input/Output modules guarantee a safe switching of the desired output (switching off all outputs in the case of errors) within 37.6 ms (Mode 3) or 67.6 ms (Mode 1), each including an assumed relay release time of 15 ms.

In case of error the operational outputs are switched off.

1.6 Safety function

The safety functions realised by the **SFL** are defined as follows.

1. A 2 channel safety input is read. The PS program running in the CPU module controls the state of the outputs depending on the input information. On request, this leads to a 2 channel shut-down. In the case of several input and output pairs, each path is viewed as a separate safety function. The safety function of the output is defined as the switching off of the power in the case of semiconductor outputs or opening contacts in the case of relay outputs.

2. The second safety function is identical to that named under 1. with the one difference that 2 channel safety inputs of a slave station are used here.

The further calculations of the PL/SIL apply to a 2 channel safety function.

In the case of a 1 channel input the requirements of category 2 as per DIN EN 954-1 are only filled if the safety function is marked by the change of the input from ON to OFF. The maximum requirement interval for a safety function of category 2 amounts to 87 minutes. In the event of a safety function of category 2 being realized, a testing of the entire safety function including sensor and actuator in accordance with DIN EN ISO 13849-1.

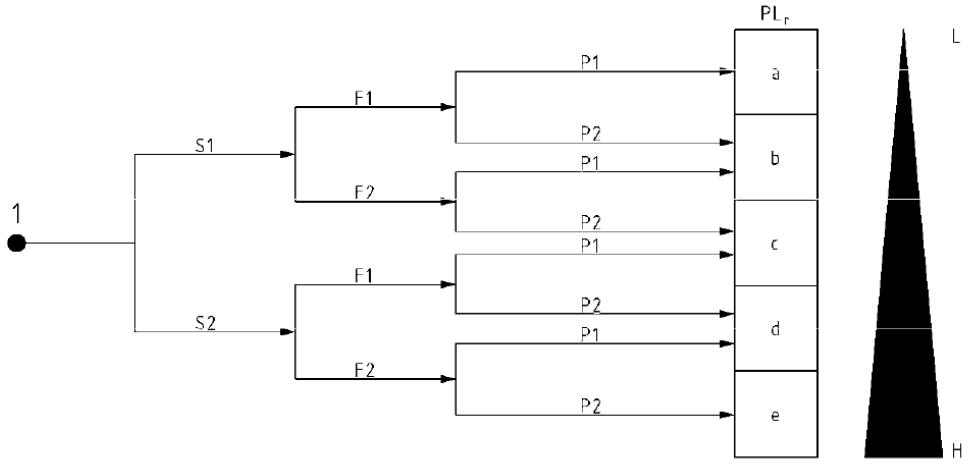
1.7 Performance Level

The Performance Level (in brief PL) to DIN EN ISO 13849-1 describes the ability of safety-orientated assemblies to perform a safety function under foreseeable conditions (which must be included in the assessment) in order to achieve the expected risk reduction. In the same way as the control categories to DIN EN 954-1, the PL is divided into 5 categories. This categorisation is made not with numbers but with the letters a to e.

The simplest way to determine the PL required for an application is to assess the risk based on a risk graph.

1.7.1 Risk graph

Starting from the left, 3 criteria are used in a tree structure to determine the required PL (PL_r for required Performance Level).



Key

1	Starting point for the evaluation of the contribution to risk reduction of a safety function	Risk parameters:	
L	Low contribution to risk reduction	S	Severity of injury
H	High contribution to risk reduction	S1	Slight (normally reversible injury)
PL_r	Required performance level	S2	Serious (normally irreversible injury including death)
		F	Frequency and/or exposure to a hazard
		F1	Seldom to less often and/or the exposure time is short
		F2	Frequent to continuous and/or the exposure time is long
		P	Possibility of avoiding the hazard or limiting the harm
		P1	Possible under specific conditions
		P2	Scarcely possible

Figure 1-2 Risk graph to determine the required PL

Warning:

Use the property F_2 for the criterion F if the intervention is made more than once per shift.

1.7.2 Determining the Performance Level

In order to determine the PL for the entire system, the PL for each individual element of the functional chain (sensor ⇨ **SFL** ⇨ actuator) must be known. The PL of the entire system can then be determined from the individual performance levels using the algorithms described below.

Performance Level of the SFL

The **SFL** safety multi application controller on its own (without the sensory system connected upstream and the actuator system connected downstream) satisfies all requirements of Performance Level e to DIN EN ISO 13849–1 with 2 channelled Input/Output circuitry.

Performance Level of the entire system

When determining the PL of the entire system a series connection of N elements whose PL is already known is assumed.

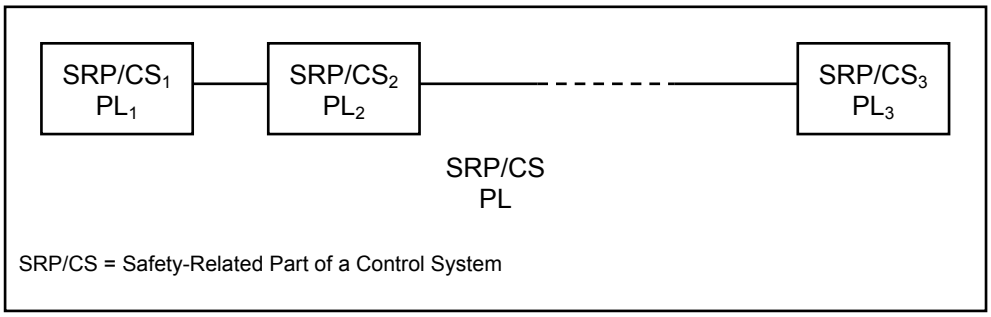


Figure 1-3 Series connection of safety relevant parts of a control system

Firstly, the element with the lowest PL in the entire system is determined. This PL (PL_{low}) is a starting point for the further determination of the overall PL.

Afterwards, the number $N_{low} \leq N$ of the elements is determined using $PL = PL_{low}$.

The PL of the entire system can then be determined using these two factors and Table 1-1.

PL_{low}	N_{low}		PL
a	> 3	⇨	none, not permit
	≤ 3	⇨	a
b	> 2	⇨	a
	≤ 2	⇨	b
c	> 2	⇨	b
	≤ 2	⇨	c
d	> 3	⇨	c
	≤ 3	⇨	d
e	> 3	⇨	d
	≤ 3	⇨	e

Table 1-1 Determination of the PL of an entire system.

The values used for calculation are based on reliable values for the center of each PL.

1.8 Safety Integrity Level

The Safety Integrity Level (SIL) to DIN EN 61508 is one of 4 levels to specify the requirements of safety integrity of the safety functions which are assigned to all elements of the chain. Level 4 is the highest and level 1 the lowest safety integrity level.

The Safety Integrity Level is defined for the following operating modes:

low demand rate : whereby the rate of demand to the safety related system is no more from once per year and is no greater than double the frequency of the repeat check.

with high demand rate/continuous demand : whereby the rate of demand to the safety related system is no more than once per year or is greater than double the frequency of the repeat check.

The average probability of a failure of a safety function with low demand rate is specified by the PFD factor (average probability of failure to perform its design function on demand).

The average probability of a failure with a function with high/continuous demand rate is specified by the PFH factor (average probability of a dangerous failure per hour).

The following is an overview of the **SFL** values.

Safety function : A 2 channel safety input affects a safety output	
Demand rate	**Failure probability**
low	PFD = 2.87 * 10^{-4}
high/continuous	PFH = 1.27 *10^{-8} /h
Safety function : 2 channel safety inputs affect one safety output	
Demand rate	**Failure probability**
low	PFD = 3.17 * 10^{-4}
high/continuous	PFH = 1.64 * 10^{-8} /h

Table 1-2 Overview of the failure probabilities

The **SFL** is suitable for SIL 3 safety functions due to the restrictions of the hardware's safety integrity:

- SFF = 99.6%
- Hardware error tolerance = 1
- Type B partial system

for SIL 3 safety functions suitably.

Either by means of the process (application) or by means of organisational measures, it must be ensured that the safety function is requested at least once per year.

Every element of the chain must satisfy all requirements (e.g. restrictions of the safety integrity of the hardware due to the architecture) of the resultant SIL

In order to determine the SIL, the PFH or PFD factors of the chain (sensor ⇨ **SFL** ⇨ actuator) must be added together depending on the operating mode. The resultant SIL can be determined using the following tables.

Σ PFD$_i$	SIL
$\geq 10^{-4}$ bis $< 10^{-3}$	3
$\geq 10^{-3}$ bis $< 10^{-2}$	2
$\geq 10^{-2}$ bis $< 10^{-1}$	1

Table 1-3 Operating mode with low demand rate

Σ PFH$_i$	SIL
$\geq 10^{-8}$ bis $< 10^{-7}$	3
$\geq 10^{-7}$ bis $< 10^{-6}$	2
$\geq 10^{-6}$ bis $< 10^{-5}$	1

Table 1-4 Operating mode with high demand rate / continuous demand

1.9 Calculation example

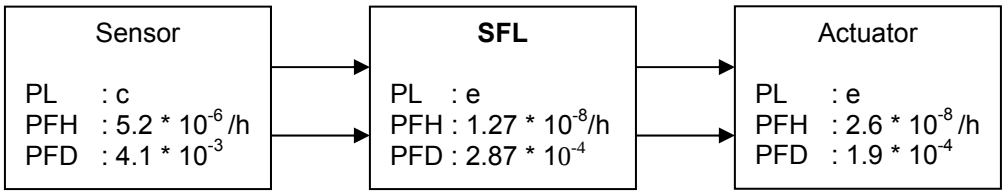


Figure 1-4 Calculation example / Performance Level / Safety Integrity Level

1.9.1 Performance Level

The PL of the entire system is calculated as follows:

PL_{Low} = c

N_{Low} = 1

Result according to Table 1-1 : PL = c

1.9.2 SIL with high demand rate

The SIL of the entire system is calculated as follows:

$PFH = PFH_{sensor} + PFH_{SFL} + PFH_{actuator}$

$= 5.2 * 10^{-6}$ /h + 1.27 *10^{-8} /h + 2.6 * 10^{-8} /h

$= 5.2 * 10^{-6}$ /h

Result according to Table 1-4 : SIL = 1

1.9.3 SIL with low demand rate

The SIL of the entire system is calculated as follows:

$PFD = PFD_{sensor} + PFD_{SFL} + PFD_{actuator}$

$= 4.1 * 10^{-3}$ + 2.87 * 10-4 + 1.9 * 10^{-4}

$= 4.6 * 10^{-3}$

Result according to Table 1-3 : SIL = 2

2 Important information

This chapter contains important warnings and information for the safe and correct use of the **SFL**.

2.1 Safety warnings

Graded safety warnings

Safety warnings are marked in this instruction manual by a symbol and the keyword CAUTION or WARNING in the margin of the page. The safety warnings are printed in bold type and in a box.

2.1.1 Definitions

Safety warnings

Text passages with a STOP symbol contain important warnings which must be heeded under all circumstances. Failure to heed these warnings may place the **SFL** controller in a state which no longer provides adequate protection for man and/or machine. Please read through such text sections with particular care.

Warnings

A CAUTION sign gives you important information and warnings which guarantee the trouble-free operation of the **SFL**. Failure to heed these warnings does not restrict the safe operation of the **SFL**.

2.1.2 Danger through misuse

The consequences of incorrect use may be personal injury to the user or third parties as well as damage to the controller, the product or to the environment. Only use the **SFL** safety multi application controller for its intended purpose!

2.1.3 Dangers from modification and retrofitting

The **SFL** safety multi application controller has been designed and constructed by us to operate safely.

It is not therefore permitted to make modifications or to retrofit the equipment.

This may affect the correct operation of the **SFL** system with the consequence of personal injury, property or environmental damage and loss of any liability.

2.1.4 Authorized persons

Only sufficiently qualified and instructed persons may operate the **SFL** safety multi application controller.

The user software may only be handled and modified by authorized and instructed persons (programmers).

An electrical technician must perform commissioning.

Only qualified personnel may perform service, maintenance, troubleshooting and error correction work.

Operators

An operator is a duly instructed person.

The operator switches the system on and off.

The operator is the actual beneficiary of the safety function.

Programmers

The programmer is a specially authorized and instructed person.

The programmer

- creates or
- modifies

the user programs.

Commissioners

The commissioner is an electrical technician.

The commissioner

- performs commissioning under increased safety measures,
- sets the device parameters
- and performs the requisite tests.

Maintenance technicians

The maintenance technician is a qualified skilled worker. He

- services the electrical and mechanical components of the **SFL** controller,
- performs maintenance work and
- looks for errors and eliminates them.

2.1.5 Accessibility of the programming software

It must be ensured that non-authorized personnel have no access to the installation program of the **SFL-SOFT** programming software or may obtain such. **SFL-SOFT**.

2.1.6 Password protection of the programming software

The **SFL-SOFT** programming software has a password protection designed to protect against non-authorized active access (modification, creation of the user program). 4 standard passwords are generated after initial installation. The programmer must overwrite all 4 standard passwords during initial use.

2.1.7 PIN Code

The **SFL** safety multi application controller has a PIN code designed to protect against non-authorized active access (configuration setting, PIN code changing, write of the user program to CPU module).

In the following cases, the PIN code setting is required.

- When changing the module configuration in Mode 1 (see chapter 5).
- When writing the user program to CPU module in Mode 3 (see chapter 6).
- When changing current PIN code to new PIN code (see chapter 5).
- When changing to Mode 1 or Mode 3 after CPU initialization.

2.1.8 Electrical connections

The **SFL** safety multi application controller must be connected to an electrical power supply.

CAUTION: Electrical voltage

An electrician must connect the system to the mains.

The power supply (24V DC) for the **SFL** safety multi application controller and all connected electrically linked to the **SFL** must confirm to DIN EN 61000-6-2 and meet one of the following demands:

- Safety mains transformer to DIN EN 61558/VDE 0570 Part 2-6: “Special requirements on safety transformers for general applications (IEC 61558-2-6:1997)“.
- Switch mode power supply to DIN EN 60950-1 :"Information technology equipment - Safety - Part 1" and to DIN EN 50178 : ""Electronic Equipment for Use in Power Installations". Furthermore the power supply unit must be suitable for supplying its SELV circuits in accordance with DIN EN 60950-1.

The mains must be appropriately fused!

The information contained in chapters 3 and 4 must be taken into consideration for the operation of the **SFL** system.

9.3 / 9-6 

The **SFL** system satisfies the pertinent provisions of the EMC Directive.

2.1.9 Shock-hazard protection

The **SFL** system is intended for use in control circuits to DIN EN 60204 for a rated voltage of 24V DC. The appropriate requirements placed on shock-hazard protection must be satisfied for the applications.

For reasons of shock-hazard protection all connections must have the appropriate mating connectors.

2.1.10 Maintenance

Maintenance work

Incorrect maintenance could lead to death, injury, damage to property or damage to the environment. Only qualified persons may perform maintenance, trouble-shooting and error elimination work. Switch off the power supply to the **SFL** safety multi application controller. Directly after completing maintenance work replace all protective cladding and safety equipment and check that they function correctly.

Spare parts

The use of unsuitable spare parts could lead to death, injury, damage to property or harm to the environment. Spare parts must comply with the technical requirements set by the manufacturer. Only use original Panasonic Industrial Devices SUNX spare parts.

2.1.11 Disposal

Electrical waste (components, screens, etc.) can damage the environment. Dispose of electrical resources correctly or commission a specialised company to do so.

2.1.12 Liability

The contents of the following instruction manual are subject to technical modification, which may arise particularly due to constant further development of the products form Panasonic Industrial Devices SUNX Co., Ltd. Panasonic Industrial Devices SUNX Co., Ltd. assumes no liability for any printing errors or any other inaccuracies which may be contained in this instruction manual unless these are serious faults which Panasonic Industrial Devices SUNX Co., Ltd. demonstrably already knew about. The applicable national and international standards and provisions must be observed under all circumstances in addition to the instructions contained in the instruction manual.

N.B.

Incorrect use - liability exclusion.

Panasonic Industrial Devices SUNX Co., Ltd. shall not be liable for damage caused through incorrect use or application of the product.

The exact knowledge of the contents of this instruction manual is similarly viewed to be correct use. In particular, the information and safety warnings contained in this instruction manual must be heeded.

If products are used in connection with other components such as safety modules, controllers or sensors, the respective user information must be heeded.

2.2 Correct use

2.2.1 Application

The **SFL** safety multi application controller is a safety-oriented programmable control system for evaluating sensors and controlling actuators via a field bus. The **SFL** can be used in applications up to category 4 to DIN EN 954-1 or Performance Level (PL) e to DIN EN ISO 13849-1. Its use is only possible in systems in which the safe state is equal to the de-energized state.

The **SFL** system is particularly suitable for the safety-oriented evaluation and control of EMERGENCY OFF (EMERGENCY STOP) command devices, interlocking devices and other protective devices which protect operators from hazardous movement in the area of action of a machine.

Design, implementation and operating errors may affect the correct operation of the **SFL** system, resulting in injury or damage to property or to the environment. This is why only sufficiently qualified persons may operate the **SFL** system.

The **SFL** system is exclusively intended for use in machines within the scope of DIN EN 60204-1 (Electrical Equipment of Machinery).

Additional requirements arising from other provisions and regulations are not necessarily satisfied by the **SFL** system.

The **SFL** safety multi application controller may not be used in potentially explosive areas.

2.2.2 Categories, design examinations etc.

The safety structure and functionality of the **SFL** system complies with category 4 to DIN EN 954-1 or PLe to DIN EN ISO 13849-1.

In accordance with the above requirements a prototype test was performed for the **SFL** system by the Berufsgenossenschaftliche Institut für Arbeitsschutz BIA, St. Augustin/Germany (Institute for Occupational Safety).

By way of information:

The above prototype test confirms a degree of safety for the **SFL** system comparable with category 4 to DIN EN 954-1, also for emergency actions within the meaning of DIN EN 60204-1 point 9.2.5.4. Refer also to chapter 2.2.5 „Emergency actions“.

The DIN EN ISO 13849-1 is a standard within the meaning of MRL article 5 point 1 paragraph 2. See also chapter 2.2.3 "Use of electronic equipment for safety functions".

This means that when using electronic equipment the so-called presumed effect of harmonised standards is not completely available.

In Germany, there should not be any problems with employer's liability insurance associations, technical inspectors or technical supervisory agencies with regard to the use of electronic equipment with safety functions. The same applies to the majority of other EU member states.

However, it cannot completely be ruled out at the present time that there may be acceptance problems in isolated cases with respect to the use of electronic equipment for safety functions. Please consult us if this is the case. Refer also to chapter 2.2.5 „Emergency actions“.

The internal structure of the **SFL** system corresponds to DIN EN 954-1 category 4 (see above). However, the category actually achieved in the entire safety circuit (refer to figure below) and thus the degree of safety attained will similarly depend on the structure of the input and output circuitry.

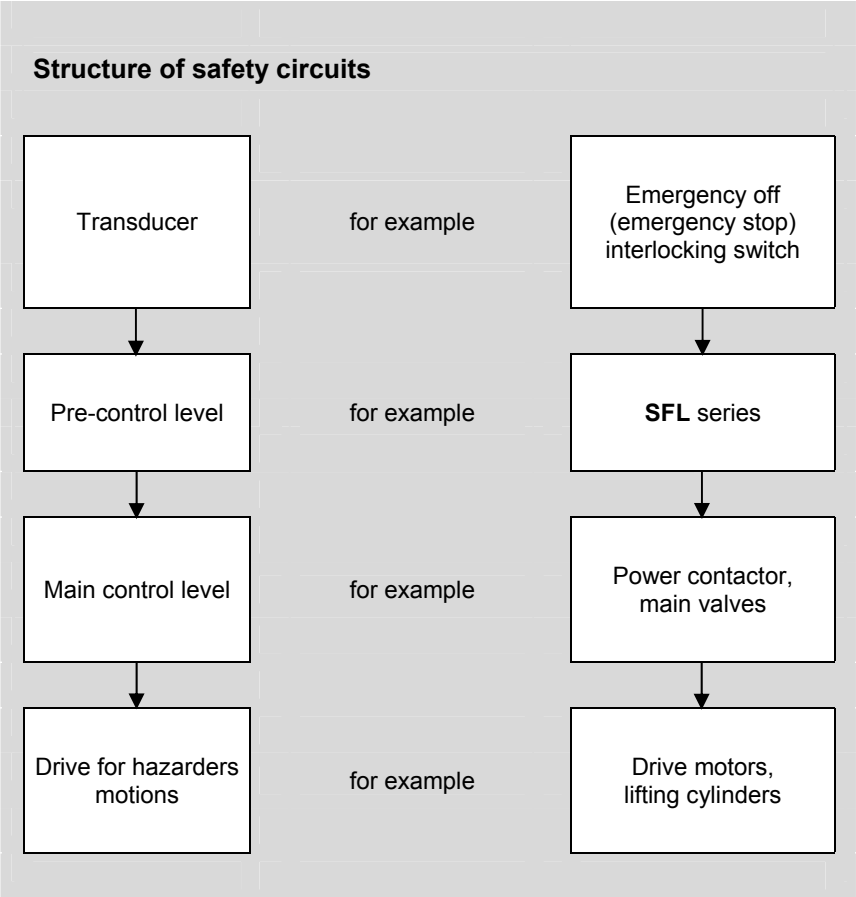


Figure 2-1 Safety chain in plants / machines

The **SFL** system is therefore only a part or a member of a chain of safety-related parts of controllers in addition to the transducers (protective devices), the main control level and the drive part of the machine. The degree of safety actually achieved will therefore depend on the overall structure of this chain.

It is the responsibility of the user to decide which safety-related measures are to be realised in the above mentioned parts of a control system.

The provisions laid down in the EC Machine Directive will be applicable here.

Any more detailed recommendations on how the safety-related parts of the control system are to be structured are to be found in the so-called C standards (machine standards) which interpret the EC Machine Directive or, if non-existent or not applicable, can be determined using the A and B standards (basic safety standards or safety group standards). Special regulations apply to products specified in Annex IV of the EC Machine Directive.

Special provisions or deviating regulations also apply to “old” or used machines which the user should find out about from the competent bodies.

2.2.3 Use of electronic equipment for safety functions.

The new facility to use electronic equipment also includes the facility to transfer safety-related signals serially, i.e. via a bus system. However, additional measures are also required here to guarantee protection in the case of an error.

The following tables together with the explanations provide an overview of the possibilities of using the **SFL** system to DIN EN 60204-1.

Control functions to DIN EN 60204-1	Stop category to point 9.2.2	DIN EN 954-1 category
Stop function	0	Category 4
Stop function	1	Category 4
Stop function	2	Only in connection with the input level of the **SFL** system, additional measures in accordance with EN 1037 (protection from unexpected start-up)
Stop function	0	Category 4 with final electrical isolation due to an electromechanical component
Stop function	1	Category 4 with final electrical isolation due to an electromechanical component
Stop function	2	Not admissible

Table 2-1 Areas of use for electronic equipment

2.2.4 Definition of the stop categories

Stop category 0:

Stopping by immediately switching off the energy supply to the machine drives (i.e. an uncontrolled stop).

Stop category 1:

A controlled stop whereby the power supply to the machine drives is maintained in order to achieve the stop and the power supply is first cut when the stop is achieved.

Stop category 2:

A controlled stop in which the power supply to the machine drives remains intact.

In the case of error the **SFL** performs a stop of category 0 for all the outputs. The planner/programmer must check whether a desired STOP1/STOP2 shut-down can be realised under these aspects without endangering man and machine. Accordingly, further measures are to be taken for applications using stop categories 1 and 2 in order to guarantee the safety of man and machine in the case of a error.

2.2.5 Emergency actions

In the case of emergency actions, a distinction must be made between a stop signal of category 0 or a stop signal of category 1. Stop category 2 is not admissible for emergency actions.

While stop category 0 or 1 comes into question for the implementation of instructions applicable to stopping in an emergency (controlling of hazarders motions) (cf. point 9.5.4.2), instructions aimed at shutting the system down in an emergency (controlling of electrical hazards) may exclusively and also logically belong to stop category 0 (cf. point 9.5.4.3).

By contrast with respect to stop 1 functions, the final shut-down of energy to the machine drive element must be ensured by using electrotechnical equipment. This means that the function may depend on an electronic switching logic (hard or software) and/or on the transmission of instructions via a communication network or a data link if a contact-dependent output level (e.g. a relay level) takes care of electrical isolation.

In accordance with the European and national preface to DIN EN 60204-1, it is admissible to use electronic equipment for the realisation of stop 0 and stop 1 functions if the relevant standards have been heeded. In the case of SFL, DIN EN 954-1 or DIN EN ISO 13849-1 has been observed.

2.3 Information on basic handling

2.3.1 Step-by-step design

With respect to the due care to be applied in designing and implementing the hardware and software of the control parts realised with the **SFL** system there are no differences by comparison with the traditional state of the art, i.e. errors and inadequacies in design and implementation can impair the intended protective functions in the same way.

Step 1:
Risk analysis in accordance with EC Machine Directive or DIN EN ISO 12100 as well as the determination of the (graded) protective measures (protective devices, additional precautionary measures, categories).

Step 2:
Planning or design of the **SFL** stations needed.

Step 3:
Planning or design of the safety relationships between these safety-oriented inputs and outputs under consideration of the desired interdisciplinary and/or partial dependencies and non-dependencies, possibly under additional consideration of different operating modes etc.

Step 4:
Assembly and wiring of the **SFL** stations.

Step 5:
Checking of correct cabling.

Step 6:
Parameter assignment of the **SFL** system. Refer to chapter 7 in this respect.

Step 7:
Reverse analysis of parameter assignment. Refer to chapter 7.13.2 in this respect.

The reverse analysis described in chapter 7.13.2 cannot replace a check of the wiring and in particular of the correct wiring of the outputs.

Step 8:
Initialisation of the **SFL** system. Before initialising the **SFL** system we recommend the temporary connection of the mobile emergency off (emergency stop) control device between mains supply and mains isolation device in order to be able to control any undesirable reactions in the safety circuit resulting from faulty wiring and/or parameter assignments.

Step 9:
Random checking of desired safety-oriented functionalities. It is not necessary to perform a complete acceptance test of the stipulations in step 3 for the **SFL** system due to the reverse analysis already performed in step 7.

Step 10:
Documentation of steps 1 to 9 pursuant to EC Machine Directive.

In project documentation special reference must be made to which outputs can be used for safety functions.

3 System description

This chapter describes the individual components of the **SFL**.

3.1 Overview

3.1.1 Product range

Module identification	Name	Description
CPU Module	**SFL-CPU-MON-V2**	Program memory: 32k Byte for PS; 32k Byte for PN program. 4 inputs for floting sensors, 24V DC 4 inputs for non-floating sensors 24V DC 6 semi-conductor outputs, 24V DC, 0.5A
CPU Module	**SFL-CPU-OP-MON-V2**	Program memory: 32k Byte for PS; 32k Byte for PN program 4 inputs for floting sensors, 24V DC 4 inputs for non-floating sensors 24V DC 6 semi-conductor outputs, 24V DC, 0.5A
Lithium battery		Rechargeable lithium battery for memory buffer of the CPU
Base	**SFL-BASE**	Backplane bus of the modules
SUB-Master-ON	**SFL-SUB-MON**	4 inputs for floting sensors, 24V DC 4 inputs for non-floating sensors 24V DC 6 semi-conductor outputs; 24V DC, 0.5A
STOP Input	**SFL-S-STP-E**	6 inputs for floating sensors; 24V DC 4 semi-conductor outputs; 24V DC, 0.5A
STOP Input	**SFL-S-STP-LC**	6 inputs for non-floating sensors; 24V DC 4 semi-conductor outputs; 24V DC, 0.5A
STOP Output	**SFL-S-STP-ELC**	4 inputs for floating sensors; 24V DC 2 inputs for non-floating sensors, 24V DC 4 semi-conductor outputs; 24V DC, 0.5A
Input	**SFL-S-IN-E**	16 inputs for floating sensors; 24V DC
Input	**SFL-S-IN-LC**	16 inputs for non-floating sensors, 24V DC
Relay Output	**SFL-RELAY**	2 x 2 relay outputs; 24V DC, 4.0 A
Output	**SFL-S-OUT**	16 semi-conductor outputs, 24V DC, 0.3A
Operational input	**SFL-NS-IN**	16 operational inputs, 24V DC
Operational output	**SFL-NS-OUT+**	16 operational outputs, 24V DC
Power	**SFL-POWER**	Voltage supply: 24V DC ±10%, max. 2A
Power	**SFL-BOOSTER**	Voltage supply: 24V DC ±10%, max. 2A

Programming Software	**SFL-SOFT**	Software for **SFL** (CD-ROM version)
USB connection cable		Connection between PC and **SFL** A commercial USB cable (A: B type) is used.
Dedicated cable	**SFL-CC1** **SFL-CC3**	MODBUS connection cable

Table 3-1 Product range

Information on the individual modules can be found in the following chapters.

3.1.2 Basic module structure

All modules have a plastic housing made of PPE. Depending on version their overall width differs (30mm / 45mm).There is a plug on the reverse side for the electrical connection to the back plane BUS and the slots for mechanical fixing. The front area is subdivided into a display area and a connection/operating area. The modules have different colour markings for faster identification.

Module type	Model	Marking
Master module	**SFL-CPU-OP-MON-V2**, **SFL-CPU-MON-V2**	yellow
Sub-Master module	**SFL-SUB-MON**	yellow
E/A module	**SFL-S-STP-E**, **SFL-S-STP-LC**, **SFL-S-STP-ELC**, **SFL-S-IN-E**, **SFL-S-IN-LC**, **SFL-S-OUT**, **SFL-RELAY**	yellow
Operational I/O modules	**SFL-NS-IN**, **SFL-NS-OUT+**	white

Table 3-2 Colour marking of the modules

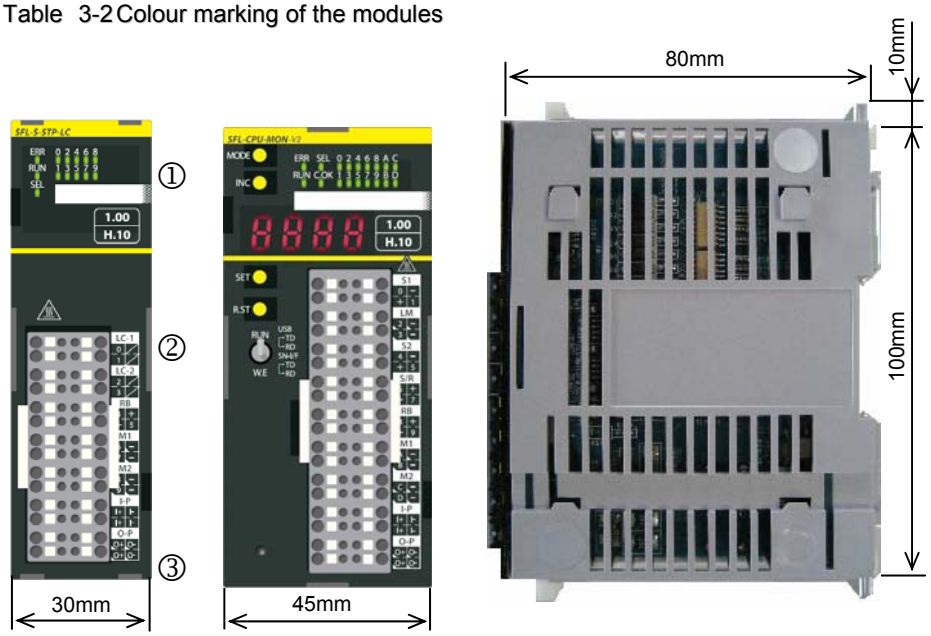


Figure 3-1 Basic module structure

① **Display area for:**
- Status of the inputs/outputs
- Error messages
- Communication

② **Connection area for:**
- Sensors, actuators
- Power supply
- Possible additional operating units

③ **Module width**
- 30mm
- 45mm

3.2 Channel overview

The following table describes the channels of each module

Module	Inputs			Outputs			
		Safe			Safe		
	Opera-tional			Opera-tional	Semi-conductor		Relay
		1 channel category 2	2 channel category 4/3		1 channel category 2	2 channel category 4	Category 4
SFL-CPU-MON-V2 SFL-CPU-OP-MON-V2	-	8	4	-	6	3	-
SFL-SUB-MON	-	8	4	-	6	3	-
SFL-S-STP-E	-	6	3		4	2	
SFL-S-STP-LC	-	6	3		4	2	
SFL-S-STP-ELC	-	6	3		4	2	
SFL-RELAY	-	-	-	-	-	-	2 x 2
SFL-S-IN-E	-	16	8	-	-	-	-
SFL-S-IN-LC	-	16	8	-	-	-	-
SFL-S-OUT	-	-	-	-	16	8	-
SFL-NS-IN	16	-	-		-	-	-
SFL-NS-OUT+	-	-	-	16	-	-	-
SFL-POWER	-	-	-	-	-	-	-
SFL-BOOSTER	-	-	-	-	-	-	-

Table 3-3 Channels used by each module

The specification of 1 channel / 2 channel is to be understood as OR, i.e. the CPU MON module has either 8 x 1 channel OR 4 x 2 channel inputs.

In Mode 1 the assignment of the input is fixed. Details may be found in the description of the individual modules.

3.3 ID code of the modules

Each module type has a clear ID code. The following table describes the ID code for every module.

Module	ID Code			
	Master	**SUB-Master**	**I/O**	**Operational I/O**
SFL-CPU-MON-V2 SFL-CPU-OP-MON-V2	01H	-	-	-
SFL-SUB-MON	-	08H	-	-
SFL-S-STP-E	-	-	10H	-
SFL-S-STP-LC	-	-	20H	-
SFL-S-STP-ELC	-	-	30H	-
SFL-RELAY	-	-	50H	-
SFL-S-IN-E	-	-	80H	-
SFL-S-IN-LC	-	-	81H	-
SFL-S-OUT	-	-	90H	-
SFL-NS-IN	-	-	-	E0H
SFL-NS-OUT+	-	-	-	F0H

Table 3-4 ID codes of the modules

3.4 SFL operating modes

3.4.1 Overview

6.1 / 6–2 

The operating modes Mode 1 and Mode 2 are implemented in the **SFL**.

Mode 3 is the free programmable operating mode and requires the **SFL-SOFT** programming interface (see Chapter 6.1). Mode 1 is the 'hardwired' mode, i.e. programming via software is not necessary. The system function is derived by the order or arrangement of the modules in the system (see Chapter 5.1).

5.1 / 5–2 

On delivery the CPU module is in Mode 0, i.e. the safe program (PS), the operational program (PN) and function blocks (FB) may not be executed.

The following table describes the possible change in operating modes, without implementing a new change to Mode 0.

	Delivery Mode 0	**Mode 1**	**Mode 3**
Mode 1	X	-	-
Mode 3	X	X	-

Table 3-5 Changing the operating modes

Relationship between the operating modes Mode 0, Mode 1 and Mode 3

The following drawing shows the relationship between the operating modes. The PIN Codes ‘1234’ and ‘5678’ are to be understood as examples.

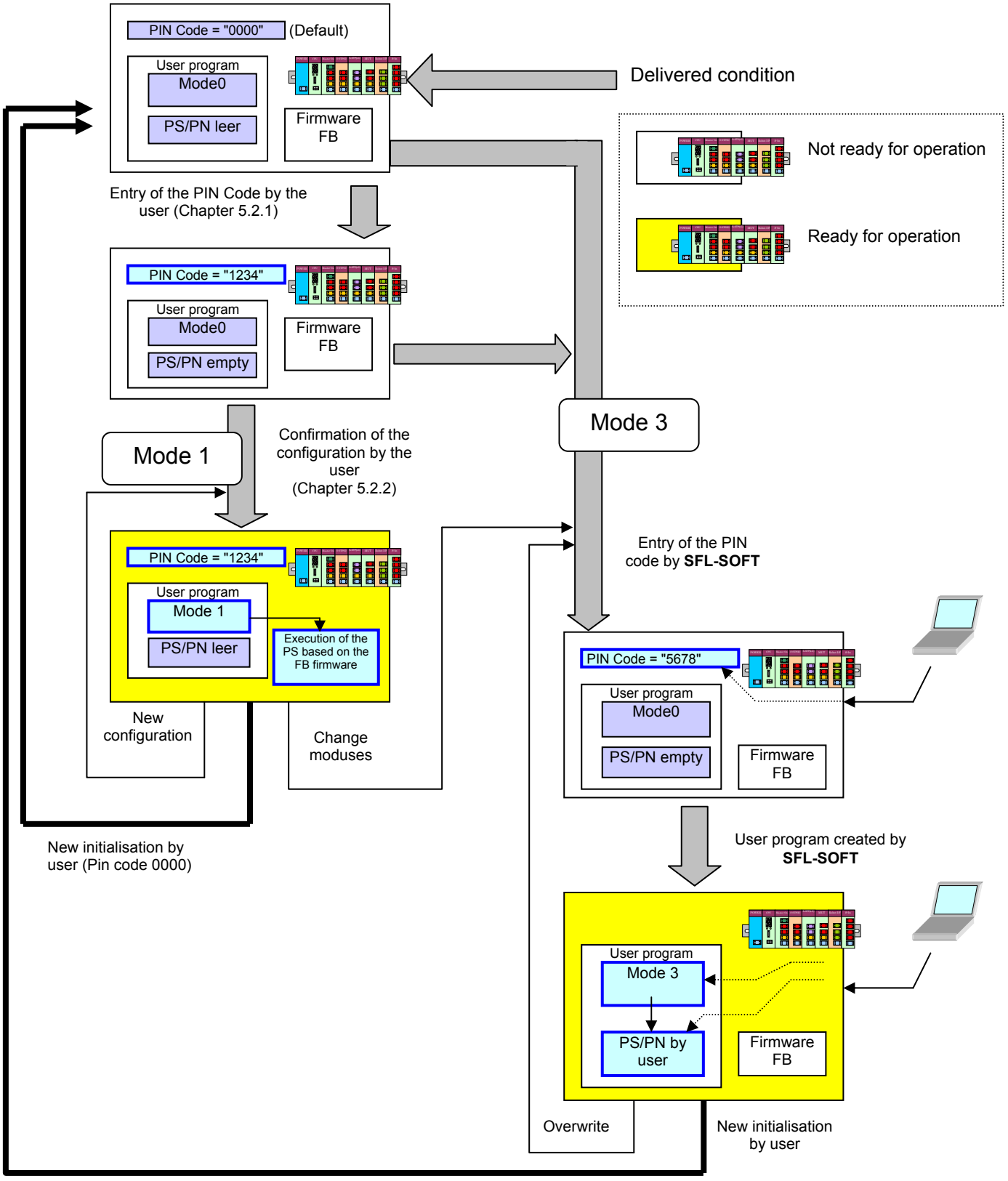


Figure 3-2 Relationship between the operating modes

3.4.2 Operating mode 0

Devices are in the delivered state and are not ready for operation. All outputs are in safe state.

3.4.3 Operating mode 1

Mode 1 is i the ‘hardwired’ mode, i.e. the system functions result from the arrangement or the order of the individual modules.

The version is realised by the firmware FB incorporated in the CPU module.

7.12.2/ 7-63 

The details of the FB firmware see Chapter 7.12.2.

The FB firmware is executed in the order of the connected modules which is why ‘hardwired’ programming is possible.

7.15 / 7-95 

In Mode 1 it is possible to monitor the function by **SFL-SOFT**. For detailed information see Chapter 7.15.

3.4.4 Operating mode 2

This mode is not currently realised.

3.4.5 Operating mode 3

Mode 3 is the mode for free programming.

It is possible to create a safe (PS) and an operational program (PN). These programs can be created in accordance with the user’s specifications.

Programming is based on the ladder diagrams and function blocks in accordance with IEC 61131. Programming with max. flexibility is possible in Mode 3.

In addition the operational functions are possible by using the operational inputs/outputs.

When changing from Mode 1 to Mode 3, the parameter assignment of the inputs/outputs is assumed by Mode 1. This should be considered in further programming.

3.5 Power supply

3.5.1 General description

The supply voltage for the **SFL** is 24V DC. The safe/operational I/O modules can be supplied with power from the power supply unit for the power/booster module or from their own power supply unit.

The power/booster has an internal DC/DC transformer that generates the necessary 3.3V operating voltages for the logic parts from the 24V DC supply. The logic parts of the remaining modules are supplied via the back plane BUS.

All existing and generated power voltage is monitored internally for overvoltage and undervoltage. A detailed description of voltage monitoring is provided in Chapter 3.7.5.

3.7.5 / 3-24

3.5.2 External power supply units

The power supply units used for power supply must comply with DIN EN 61000-6-2 and one of the following requirements:

- Safety mains transformer to DIN EN 61558/VDE 0570 Part 2-6: 'Special Requirements on Safety Transformers for General Applications (IEC 61558-2-6:1997)'
- Switch mode power supply to DIN EN 60950-1: 'Information Technology Equipment – Safety' and to DIN EN 50178: 'Electronic Equipment for Use in Power Installations'. Furthermore, the power supply unit must suitably supply its SELV circuits to DIN EN 60950-1.

3.5.3 SFL-POWER

Technical data

Position	Description
Name	**SFL-POWER**
Supply voltage	24V DC ±10%
Power input	1.0A with max. number of modules
Wattage	Max. 24W
Back plane bus	3.3V DC ±10%, Max. 15W
Fusing	Internal polymer fuse 1.0A
Dimensions / weight	30 × 100 × 80 (W/H/D) / 160g

Table 3-6 Data of the **SFL-POWER** module

Housing description

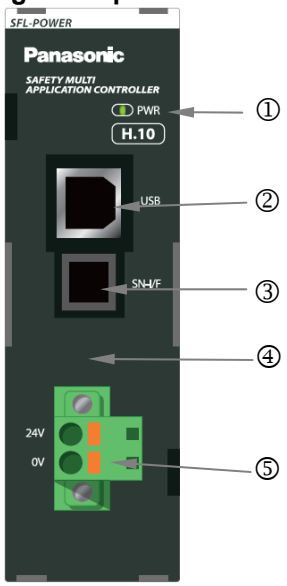


Figure 3-3 **SFL-POWER**

① Power LED's
Green : power OK

② Programming interface
The power module has a USB interface to transfer the user program to the CPU module and to read-out diagnostic information. A standard USB interface must be available at the PC: see Chapter 7.7.

7.7 / 7-11

③ SN-Interface (SN-I/F)
The SN-I/F is used for connection to external MODBUS communication.

④ Polyfuse, 1.0A

⑤ Connection 24V DC

3.5.4 SFL-BOOSTER

3.6.1 / 3-12 The use of the booster module is described in Chapter 3.6.1.

Technical data

Position	Description
Name	**SFL-BOOSTER**
Supply voltage	24V DC ±10%
Power input	1.0A with max. number of modules
Wattage	Max. 24W
Back plane bus	3.3V DC ±10%, max. 15W
Fusing	Internal polymer fuse 1.0A
Dimensions / weight	30 × 100 × 80 (W/H/D) / 155g

Table 3-7 Data of the **SFL-BOOSTER** module

Housing description

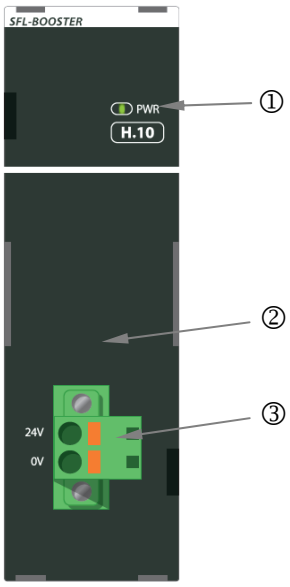


Figure 3-4 **SFL-BOOSTER**

① **Power LED's**
Green: power OK

② **Polyfuse 1.0A**

③ **Connection 24V DC**

3.6 SFL-BASE

Technical data

Position	Description
Name	**SFL-BASE**
Weight	5g

Table 3-8 Data of the **SFL-BASE** module

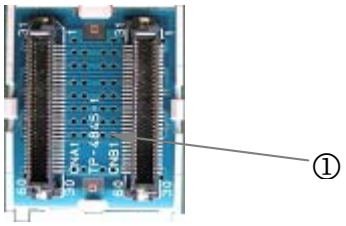


Figure 3-5 **SFL-BASE**

① **Backplane BUS**

3.6.1 General description

The **SFL-BASE** module realises the back plane Bus.

The power module is always the far left module. The CPU is always located to the right of it. I/O modules and operational I/O modules can then be positioned in any order. It must be considered that the operational I/O modules must always be placed to the right of the safe modules.

If 10 or more modules (including CPU) are used, a booster module must be positioned between slot 8 and slot 9.

Slot	-	0	1	2	…	7	8	-	9	10	…	14	15
Module	Power	CPU	Safe/operational modules					Booster	Safe/operational modules				Operational modules

Table 3-9 Arrangement on the back plane Bus

3.7 SFL-CPU-MON-V2 / SFL-CPU-OP-MON-V2

3.7.1 General description

The CPU module is the central control unit of the **SFL** and is responsible for the following:

- Executing the user program/FB firmware
- Evaluating and controlling the safe and operational I/O modules
- Monitoring power supply
- Visualising status/error messages of the **SFL**

The version **SFL-CPU-OP-MON-V2** has an additional battery enabling the operational data (e.g. error protocol, data of the PN program, ….) to be maintained also if the power is switched off (see Chapter 3.7.4 / エラー! ブックマークが定義されていません。).

Operating mode 1

In Mode 1 the **SFL** system can be subdivided into master and SUB-Master groups. In this mode the CPU module is the master for the entire system.

The CPU module has two 2 channel inputs for floating contacts which may be used for applications up to control category 4 (category 4 to DIN EN 954-1) or PLe (to DIN EN ISO 13849-1). If one of the inputs of the master system (CPU) is opened all inputs of the safe I/O modules move to the safe state. If one of the inputs of the sub-master systems is opened, only the outputs of the corresponding sub-master system move to the safe state (see Table 3-10).

In order to activate the outputs the reset input to reset the safety function must be activated on the CPU module to activate the outputs. If the safety function is reset, the ‘Master-on’ output is activated. The ‘Ready’ output has no function in Mode 1. The ‘ErrorReset’ input serves to reset the entire system after an error (e.g. wiring).

If the reset input is used by the SUB-Master or the CPU module the requirements set out in Chapter 7.11.1 Manual resetting (Reset, acknowledgement).

Slot	-	0	1 ... 6	7	8	-	9	10 11 12	13
Modules	Power	CPU	Safe I/O modules	SUB-Master 1 module	Safe I/O module	Booster	SUB-Master 2 module	Safe I/O module	Operational I/O modules

Table 3-10 Example for Master / SUB-Master groups

Master group : slot 0 - 12.

SUB-Master group 1 : slot 7 – slot 8

SUB-Master group 2 : slot 9 – slot 12

Operational I/O : slot 13

Operating mode 3

If the **SFL** is in Mode 3, the CPU module acts like a module with 4 x 2 inputs and 3 x 2 outputs.

Technical data

Position	Description
Name	**SFL-CPU-MON-V2** / **SFL-CPU-OP-MON-V2**
Operating voltage / current	24V DC ±10% / 70mA
Fusing	Internal fuse Input circuit: 3.2A Output circuit: 4.0A
Number of safety inputs	2 x 2 floating and 2 x 2 non-floating
Input resistence	Approx. 4.7kΩ
Input current	5mA
High / low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (Input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Number of safety outputs	3 x 2
Output current	Max. 0.5 A ohmic / output
Maximum switching voltage	24V DC
Connecting plug	9 x 4-pin plug MORIMATSU M800-S 36-pin plug MORIMATSU M820A-09-xx
Dimensions / weight	45 x 100 x 80 mm / (W/H/D) / 230g

Table 3-11 Technical data **SFL-CPU-MON-V2** / **SFL-CPU-OP-MON-V2**

Housing description

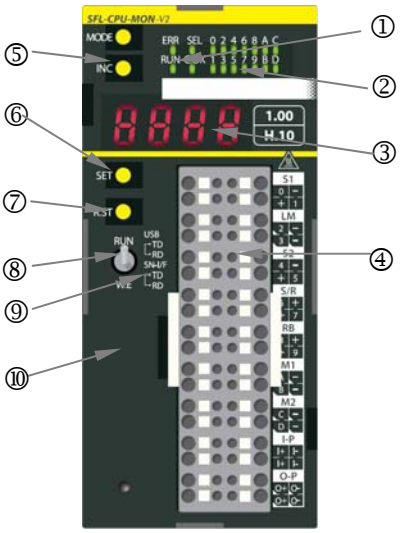


Figure 3-6
SFL-CPU-MON-V2 /
SFL-CPU-OP-MON-V2

① Status LED

ERR = on : Error / alarm
off : Operation
RUN = on : User program active
Flashes : Initialisation phase
off : User program inactive
SEL = Initialisation phase

② Display I/O

00h - 0Fh = Status of I/O

③ 7 SEG LED

④ Terminal

00h - 0Fh = Connection sensor / actuator
I+, O+ = Power supply (24V DC)
I-, O- = Power supply (0V DC)

⑤ MODE/INC button

⑥ SET button

⑦ Restart button

⑧ RUN/W.E. button

⑨ Communication LEDs

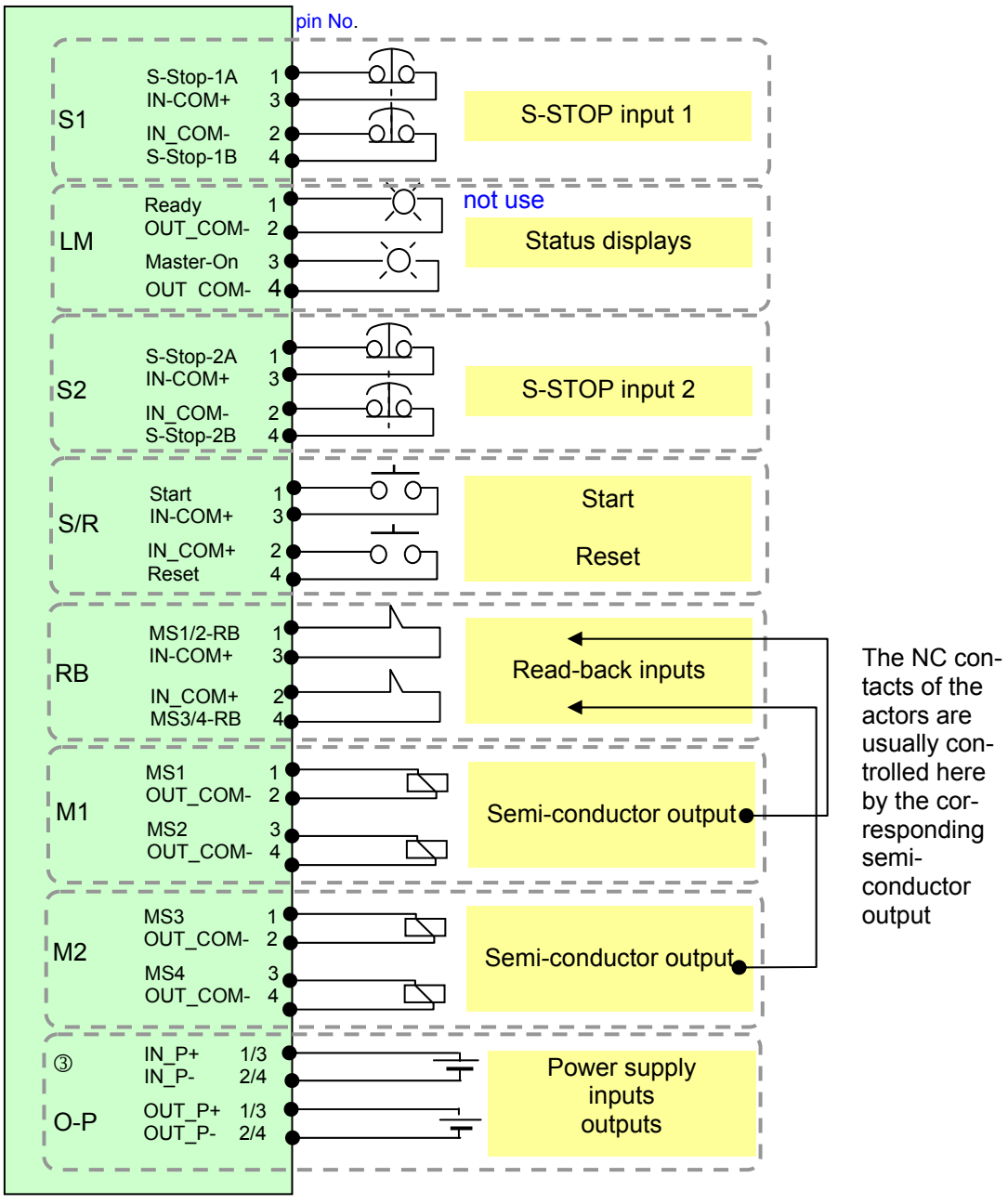
⑩ Battery

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
S1	00	1	0	S-Stop-1A	01	2	-	IN_COM-
	00	3	+	IN_COM+	01	4	1	S-Stop-1B
LM	02	1	2	Ready -	02	2	-	OUT_COM-
	03	3	3	Master-On -	03	4	-	OUT_COM-
S2	04	1	4	S-Stop-2A	05	2	-	IN_COM-
	04	3	+	IN_COM+	05	4	5	S-Stop-2B
S/R	06	1	6	Start	07	2	+	IN_COM+
	06	3	+	IN_COM+	07	4	7	Reset
RB	08	1	8	MS1/2-RB	09	2	+	IN_COM+
	08	3	+	IN_COM+	09	4	9	MS3/4-RB
M1	0A	1	A	MS1 -	0A	2	-	OUT_COM-
	0B	3	B	MS2 -	0B	4	-	OUT_COM-
M2	0C	1	C	MS3 -	0C	2	-	OUT_COM-
	0D	3	D	MS4 -	0D	4	-	OUT_COM-
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-12 Terminal diagram **SFL-CPU-MON-V2/ SFL-CPU-OP-MON-V2**

The pin no. 1 and 3 or 2 and 4 of the I-P or O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-CPU-MON-V2 / SFL-CPU-OP-MON-V2(-V)terminal diagram for Mode 1



Terminal block pin number mapping

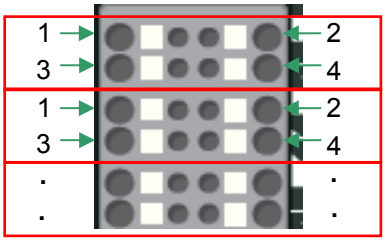


Figure 3-7 **SFL-CPU-MON-V2** / **SFL-CPU-OP-MON-V2** terminal diagram

3.7.2 Displays

Status display

The status display consists of 4 individual 7 segment displays. These display the momentary operating status as well as any fault / error messages from the **SFL**. The individual fault / error messages are described in Chapter 8.3.3.

Figure 3-8 Displays of the CPU module

Status LEDs

The 4 status LEDs display the status of the most important functions of the **SFL**. These are the status of supply voltage as well as operating and error case of the **SFL**.

Name	Colour	Description
RUN	green	Lights up: User program active Flashes: Initialisation phase Off: User program inactive
ERR	red	On: Error / alarm Flashes: Error Off: Operation
SEL	green	On: Initialisation Off: Operation
C.OK	green	On: Initialisation Flashes: Initialisation Off: Operation

Table 3-13 States of status LEDs

SN-I/F communication LEDs

The SN-I/F LEDs show the states of the serial reading/writing line of an optionally connected gateway. The **SFL** interrogates this line cyclically. During operation these LEDs therefore flash constantly.

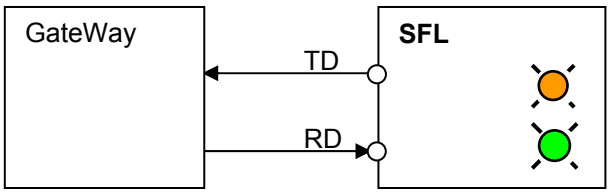
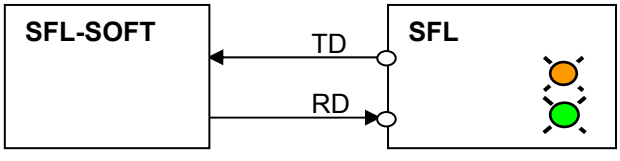


Figure 3-9 Communication LEDs of the **SFL-CPU** module

Programmer I/F communication LEDs

The programmer I/F LEDs show the states of the serial reading/writing line of an optionally connected SFL with **SFL-SOFT**. The **SFL** interrogates this line cyclically. During operation these LEDs therefore flash constantly.



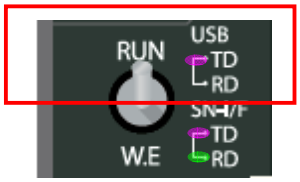


Figure 3-10 Programmer I/F LEDs of the **SFL-CPU** module

3.7.3 Operating elements

The RUN/W.E. switch does not have the function of a start and restart lock. The start and restart lock must be implemented in the user program.

RUN/W.E. switch

The operating mode of the CPU module can be selected using this switch. 2 operating modes are available:

W.E. = the user program can be transferred (Write Enable)

RUN = the user program is in operation.

Figure 3-11

RUN/W.E. switch

After switching on the supply voltage the internally moves to the RUN operating mode (if no errors occur during the self-check) The user program only starts automatically, however, if the RUN/W.E. switch is similarly set to RUN.

R.ST (RESTART) switch

The operating mode can be altered using the RUN/W.E switch and the R.ST switch. The relationship is described in Table 3-14.

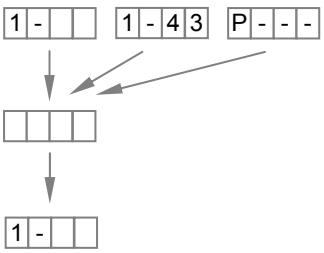
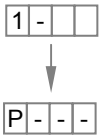
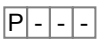
Position of RUN/W.E. switch	Function	Actuation of the R.ST switch
RUN	The user program is executed only in this operating mode.	1. User program is started once initialisation has been completed. 2. Programm is started after the configuration/user program has been transmitted.
	Display during normal operation:  "-" cganges to "8". RUN LED lights up.	The display changes as follows, e.g. Mode 1  "-" changes to "8". RUN LED lights up.
W.E. (without R.ST)	Execution of the user program is stopped.	Ready to transmit the configuration/user program (wait for entry of the PIN code).
	The user program is reset if the RUN/W.E. switch is set W.E. during RUN. The mode is shown (no change from '-' to '8'), e.g. Mode 1. 	The display shows the mode (the examples shows Mode 1).  Changes to the status to transfer the configuration/user program (wait for entry of the PIN code).
W.E. (after R.ST)	Ready to transmit the configuration/the user program (wait for entry of the PIN code).	Switch without function.
	"P - - -" is shown. 	

Table 3-14 Operating modes of **SFL-CPU**

Operating modes

Executing the user program

To start the user program the RUN/W.E. switch must be in the RUN position with Power On.

Interrupting the user program

In order to briefly interrupt the user program (without having to initialise the **SFL**) once again) move the RUN/W.E. switch to the W.E. position. To restart the user program move the RUN/W.E. to the RUN position and actuate the R.ST switch.

New user program

Move the RUN/W.E. switch to the W.E. position and actuate the R.ST switch. ‘P---‘ is shown on the status display. The further settings for Mode 1 are explained in Chapter 5 and the setting for Mode 3 in Chapter 6.

After completion of the settings the RUN/W.E. switch must be moved to the RUN position again. After this a Power OFF / ON is to be executed or the R.ST switch actuated. The ‘1---’ display appears on the status display for Mode 1 and the display ‘3---’ for Mode 3. The program is executed when an ongoing ‘8’ is shown.

MODE/INC buttons

The internal menu structure may be navigated using these buttons. The MODE button selects the individual menu options. The INC button changes to the next subentry.

Simultaneously pressing both buttons (for longer than 2s) causes the system version to be displayed. The status menu may be reached by pressing the MODE button for a long time (longer than 2s). Briefly pressing the MODE button selects the displayed menu entry and briefly pressing the INC button changes to the next menu subentry. Without actuation, the display (if it contains errors or warnings) switches between the mode and the error code in a second rhythm.

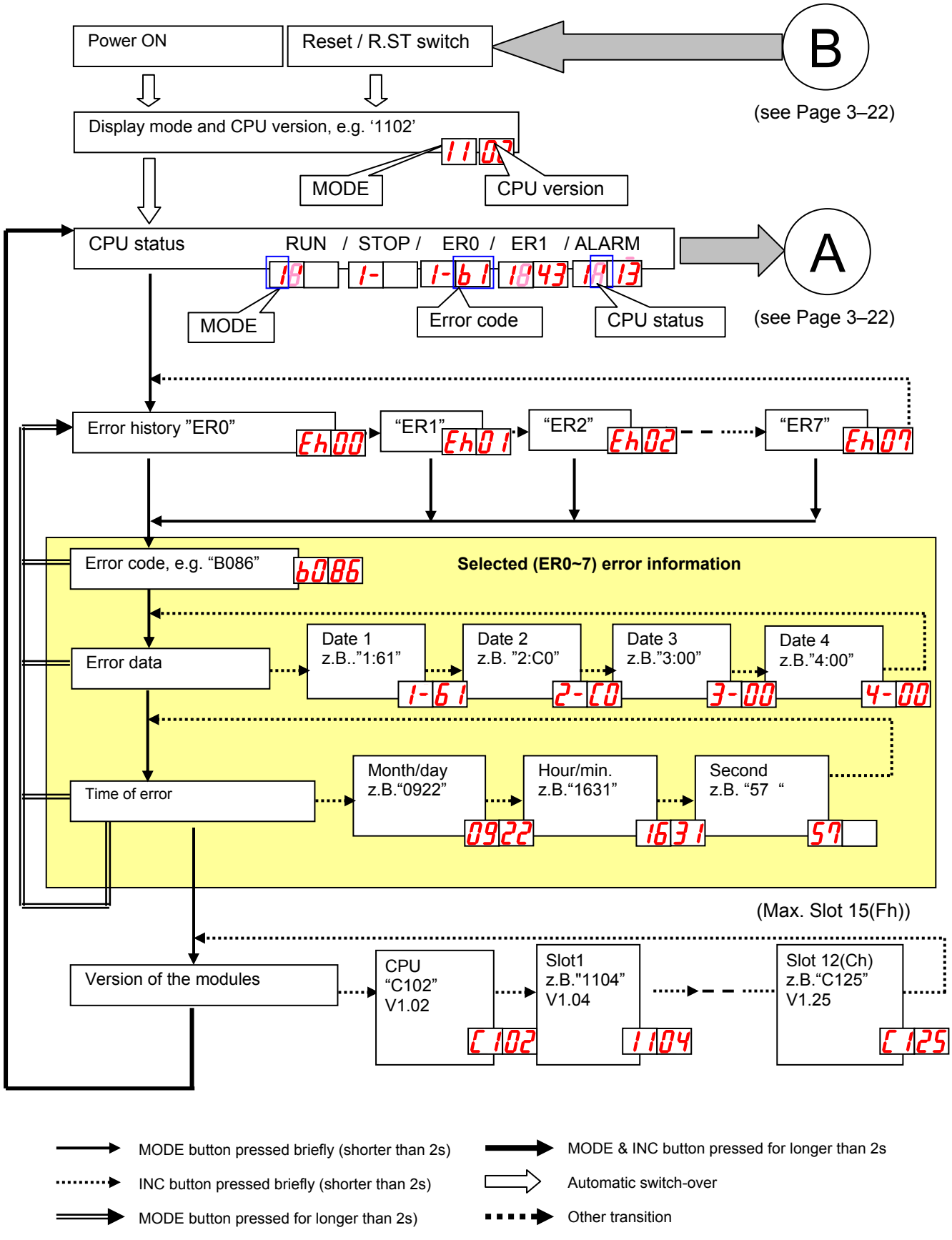


Figure 3-12 CPU module's menu structure

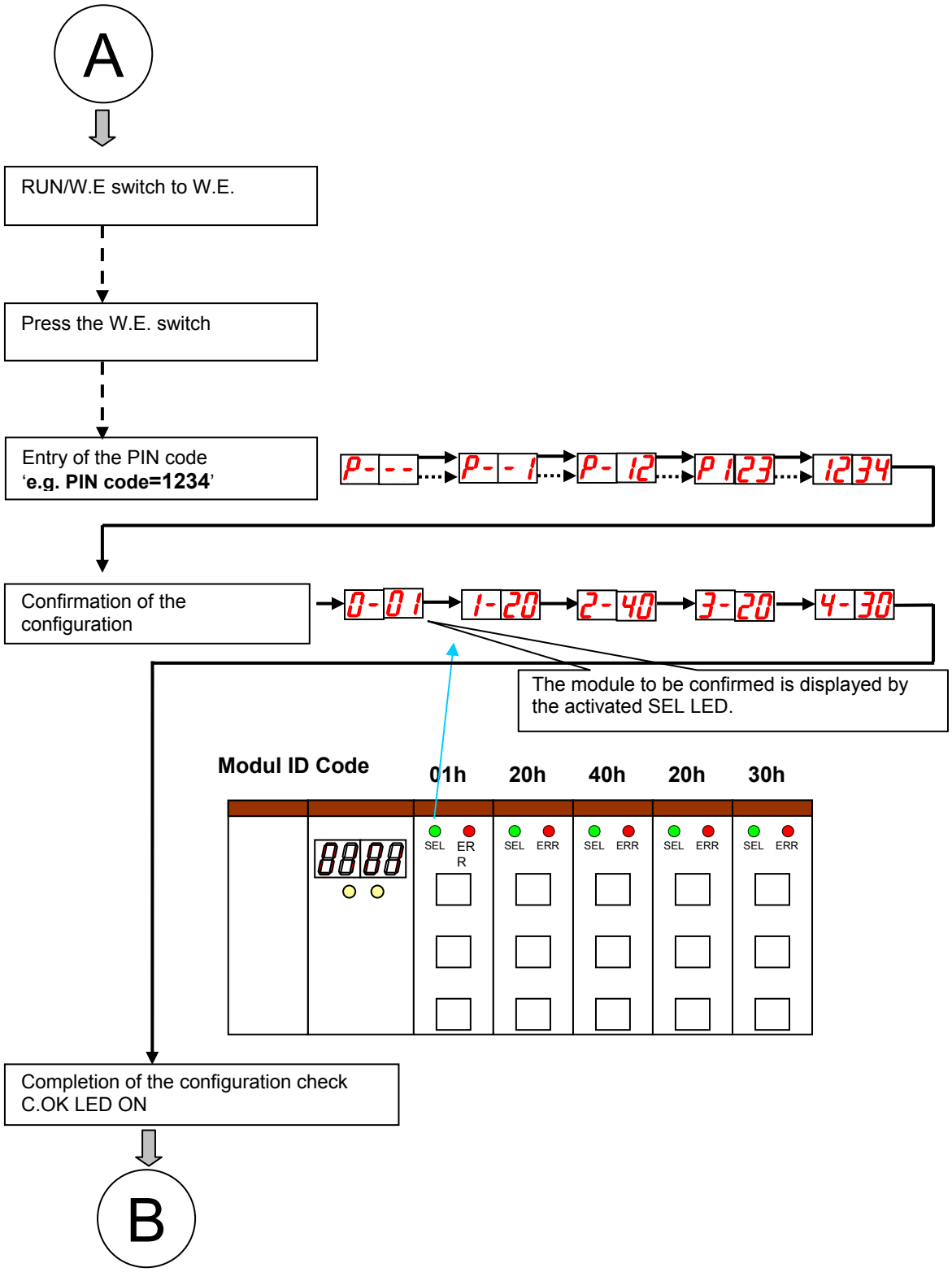


Figure 3-13 Setting of the PIN code and confirmation of the configuration

3.7.4 Back-up battery

In order to maintain the data of the operational data (e.g. error protocol, data of the PN program, …) a rechargeable lithium battery is located in every CPU module of the version **SFL-CPU-OP-MON-V2** (the user program is stored in a non-volatile manner in a flash). The connection is made via a 3 pole plug. It has a service life of approximately 5 years. Approximately 4 hours' operation per day are required to achieve an adequate charged state. Data is then for secure for over 1 year (at 25°C) if the **SFL** is in a de-energised state.

The battery voltage is monitored by the CPU module. If the voltage drops below a minimum level, a battery alarm is given (error code 022). If the battery alarm continues to exist after a charging period of 8 hours, the battery must be replaced.

4.2.3 / 4-9

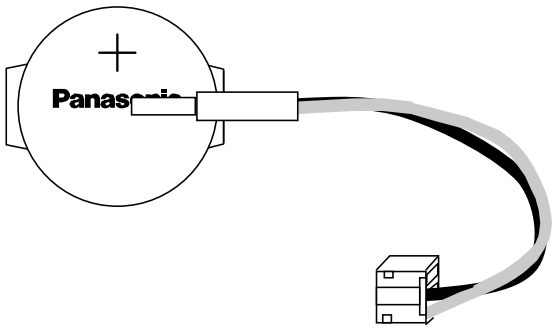


Figure 3-14 Back-up battery with connectin plug

EU Battery Directive

Figure 3-15 EU Battery Directive marking

・This crossed-out wheeled bin mark is valid only in the countries of European Union.

・This crossed-out wheeled bin mark means, when abandoning batteries or storage batteries, they must be processed separately from common garbage .

・There is a separate collection system for used batteries and used storage batteries in European Union. The batteries and the storage batteries shuld be handled properly according to the local rule.

3.7.5 Voltage monitoring

Monitoring under-voltage and over-voltage

The voltage watchdog ensures a controlled power shut-down in the case of error in the power supply. Drops in power of up to 10ms are buffered by a buffer circuit. The state of the internal voltage is displayed by means of the PWR LED.

1. Correct switching-on/switching-off of the supply voltage

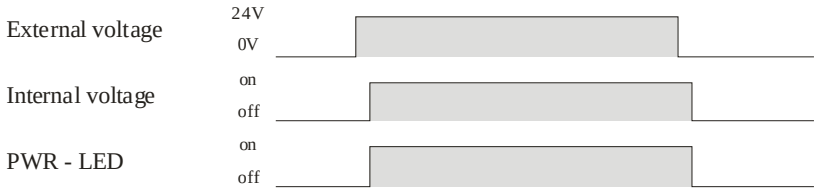


Figure 3-16 Correct switching-on/switching-off of the supply voltage

After applying the supply voltage the internal 3.3V supply is switched on and the PWR LED lights up. If the external supply is switched off, a controlled Power Down is performed and the PWR LED goes out.

2. Brief drop in voltage (≤ 10ms)

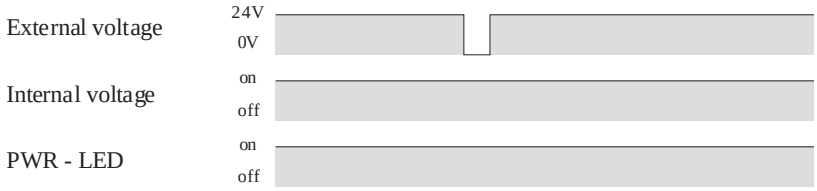


Figure 3-17 Brief drop in voltage (≤ 10ms)

A brief drop in voltage (up to 10ms) is buffered by the internal buffer circuit and has no influence on the operation of the **SFL**.

3. Drop in voltage (> 10ms)

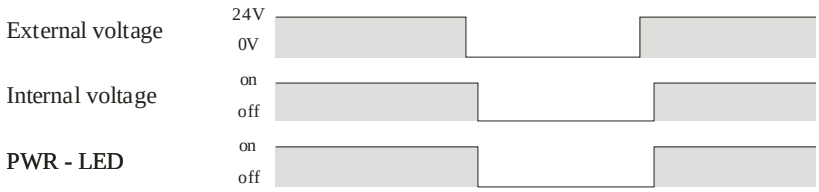


Figure 3-18 Drop in voltage (> 10 ms)

In the case of a drop in voltage > 10ms the internal Power Down measure is initiated. The PWR LED goes out. In addition the special Flag VC1 is set. Drops in voltage > 1s are recognised as shut-down and the special Flag VC1 is not set.

4. Voltage lower than 18V

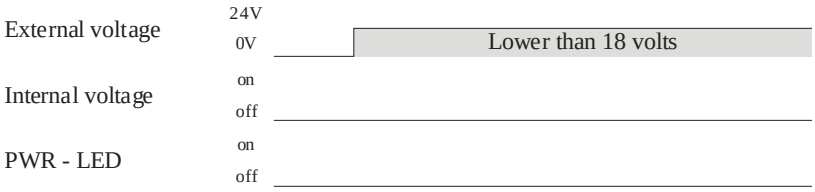


Figure 3-19 Voltage lower than 18V

The internal supply of the **SFL** is not released if the external voltage is below 18V.

5. Voltage greater than 36V

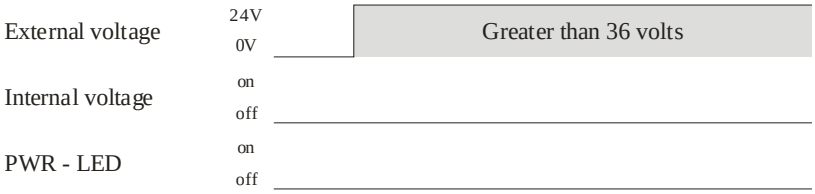


Figure 3-20 Voltage greater than 36V

The internal fuse of the **SFL** is triggered if the external supply voltage is above 36V.

The internal fuse is intended to protect the CPU module from overvoltage. Once the fuse is effective, it is not possible to operate the **SFL** any longer.

Error messages

8.3.3 / 8-5

If errors occur in the above described measures or other internal/external errors, these are read out as error messages. Every error message can be generated by processor A/B. Which processor has generated the message can be recognised by the prefix (A/B). The table below shows an excerpt of the possible error messages of the CPU module. A complete list of all error messages is provided in Chapter 8.3.3.

Test	Error code	Meaning
Internal power supply	A/B 013	Voltage of the internal 24V supply voltage too low.
Back-up battery	A/B 022	Voltage of the back-up battery too low.
Back-up battery	A/B 0AC	Voltage of the back-up battery has an illegal value.
Real-time clock	A/B 0A3	Error in response of the real-time clock.
Real-time clock	A/B 0AF	Real-time clock was not yet set.

Table 3-15 Selective error messages of the CPU module

3.7.6 Memory areas

7.10 / 7-31

The CPU module has different memory areas. Depending on the function/operation, the address is to be prefixed by a corresponding area specifier. Chapter 7.10 describes which memory area is used for which function/operation. The following table provides an overview of the individual areas.

.

specifier	Use	Address range	BIT address		Number of BITs		WORD address		Number of WORDs		Data maintained after switching off
			PS	PN	PS	PN	PS	PN	PS	PN	
I/X	Input	BIT	I/Q000 - I/Q0FF (03FF)	X/Y400 - X/Y4FF (07FF)	256 (1024)	256 (1024)	I/Q00W - I/Q0FW (3FW)	X/Y40W - X/Y4FW (7FW)	16 (64)	16 (64)	
Q/Y	Output										
M	Flag		M000 - M7FF	M000 - M7FF	2048	2048	M00W - M7FW	M00W - M7FW	128	128	
K	Hold flag		K000 - K2FF	K000 - K2FF	768	768	K00W - K2FW	K00W - K2FW	48	48	X
V	Special flag		V000 - V0FF	V000 - V0FF	256	256	V00W - V0FW	V00W - V0FW	16	16	
T/C	Timer		T000 - T0FF	T/C000 - T/C1FF	256	512	T00W - T0FW	T/C00W - T/C1FW	16	32	
L	Communication		L000 - L7FF	L000 - L7FF	2048	2048	L00W - L7FW	L00W - L7FW	128	128	
P	Edge detection		P000 - P1FF	P000 - P1FF	512	512	P00W - P1FW	P00W - P1FW	32	32	
EL	Data exchange with gateways		EL0000 - EL03FF	EL0400 - EL0BFF	1024	2048	EL000W - EL03FW	EL040W - EL0BFW	64	128	
EM	Data exchange PS/PN		EM0000 - EM03FF	EM0400 - EM07FF	1024	1024	EM000W - EM03FW	EM040W - EM07FW	64	64	
D	Data register	WORD	D0000-0 - D00FF-F	D0000 -0 - D01FF-F	512 byte	1024 byte	D0000 - D00FF	D0000 - D01FF	256	512	X
N	Value register		N0000-0 - N00FF-F	N0000-0 - N03FF-F	512 byte	2048 byte	N0000 - N00FF	N0000 - N03FF	256	1024	X
S	Special register		S0000-0 - S03FF-F	S0000 - S03FF	2048 byte	2048 byte	S0000 - S03FF	S0000 - S03FF	1024	1024	X

Table 3-16 Memory areas of the CPU module

Input/Output (I/X, Q/Y)

The address area I/Q000 to I/Q0FF is reserved for physical outputs. The address area from I/Q100 to I/Q3FF, X/Y400 to X/Y7FF can be used for internal outputs (software flag). Since the address assignment for inputs/outputs is provided from the same memory area, assignments of the same addresses to inputs/outputs such as I000/Q000, are not possible.

Flags (M)

These are intended for the interim storage of states for the purpose of later processing. The flags are set to 0 during the initialisation of the program.

Hold flags (K)

Are used in the PN program of the **SFL** for non-volatile storage of states with an installed back-up-battery. This function does not exist in the PS programs of **SFL**. They can be set analogously to the flags. The hold flags are set to 0 when initialising the program.

Special flags (V)

7.10.8 / 7-42 

Provide additional information on the state of the **SFL** (see Chapter 7.10.8).

Timers (T)

Connect a timer module to an address.

Communication (L)

Are not used in the **SFL**.

Edge detection (P)

Connect an edge detection module (high or low) to an address.

PS/PN communication (EM)

7.10.9 / 7-43 

The address area EM0000 to EM07FF is available for communication between the PS and the PN program. EM0000 to EM03FF is reserved for the PS program and EM0400 to EM07FF is reserved for the PN program. See Chapter 7.10.9 for a detailed description.

Data registers (D)

User memories for data and working register for the logic functions (AND, OR, NOT, XOR).

Value registers (N)

Contain the current counting value of a timer. A value register with the same address is allocated to every timer register. Timer statuses can therefore be read out or modified with suitable commands.

Special registers (S)

7.10.9 / 7-43 

8.3.2 / 8-3 

Contain data such as error messages, time/date, program version etc. (see Chapter 7.10.9). Chapter 8.3.2 / 8-3 contains a description of the error messages (memory area S200 to S24F).

Addressing

The different memory areas can be addressed BIT-, BYTE- or WORD-wise.

BIT address area

The WORD addressing is shown by the suffix 'W' . The BYTE addressing is followed by an '**.L**' for the low value part of the WORD and an '**.H**' for the higher value part of the WORD.

WORD address area

The BIT addressing is provided by attaching a hyphen and the bit number. The BYTE addressing is followed by an '**.L**' for the low value part of the WORD and an '**.H**' for the higher value part of the WORD.

	BIT-address	WORD-address	BYTE-address	
BIT-address area	Q000	(LSB)	(LSB)	
	Q001			
	Q002			
	Q003		Q00L	LOW BYTE
	Q004			
	Q005			
	Q006			
	Q007	Q00W	(MSB)	
	Q008		(LSB)	
	Q009			
	Q00A			
	Q00B		Q00H	HIGH BYTE
	Q00C			
	Q00D			
	Q00E			
	Q00F	(MSB)	(MSB)	
WORD-address area	D0000-0	(LSB)	(LSB)	
	D0000-1			
	D0000-2			
	D0000-3		D0000L	LOW BYTE
	D0000-4			
	D0000-5			
	D0000-6			
	D0000-7	D0000	(MSB)	
	D0000-8		(LSB)	
	D0000-9			
	D0000-A			
	D0000-B		D0000H	HIGH BYTE
	D0000-C			
	D0000-D			
	D0000-E			
	D0000-F	(MSB)	(MSB)	

Table 3-17 Adressing types of the memory

3.8 Input modules

3.8.1 General description

All input modules are self-monitoring and comply internally with control category 4 to DIN EN 954-1 (the overall safety categorisation depends on the external circuitry and the user program). The circuitry to category 3/4 must be provided by way of 2 channels to neighbouring terminals (odd and even address) against different voltage potential, whereby a cross fault can be detected. The use of inputs for AOPDs (Active Optoelectronic Protective Devices) requires a cross fault recognition of the AOPDs. The inputs can be parameterised by means of the programming software using one or 2 channels. In the case of 2 channel use, the selection can be made between an NC/NC, NO/NO and an NO/NC (antivalent) combination. The maximum delay time between the channels (in the case of 2 channel circuitry) can be set steplessly between 1.0s and 9.9s similarly using **SFL-SOFT**.

In the case of category 3/4 applications to DIN EN 954-1 or PLd/e to DIN EN ISO 13849-1 the inputs must be parameterised using 2 channels via **SFL-SOFT**.

If, due to the safety function, a 2 channel parameterisation is not possible (e.g. muting) it is absolutely essential when two 1 channel inputs are used to have the inputs on an odd and an even address. A sweeping statement about the category attained or the performance level is not possible in this case.

Basic circuit diagram for input for floating sensors

The chart shows the basic structure of an input for floating sensors. The grey shaded area exists multiply depending on the input module used. Inputs with even terminal number switch to plus by means of the sensor and with odd terminal number to minus.

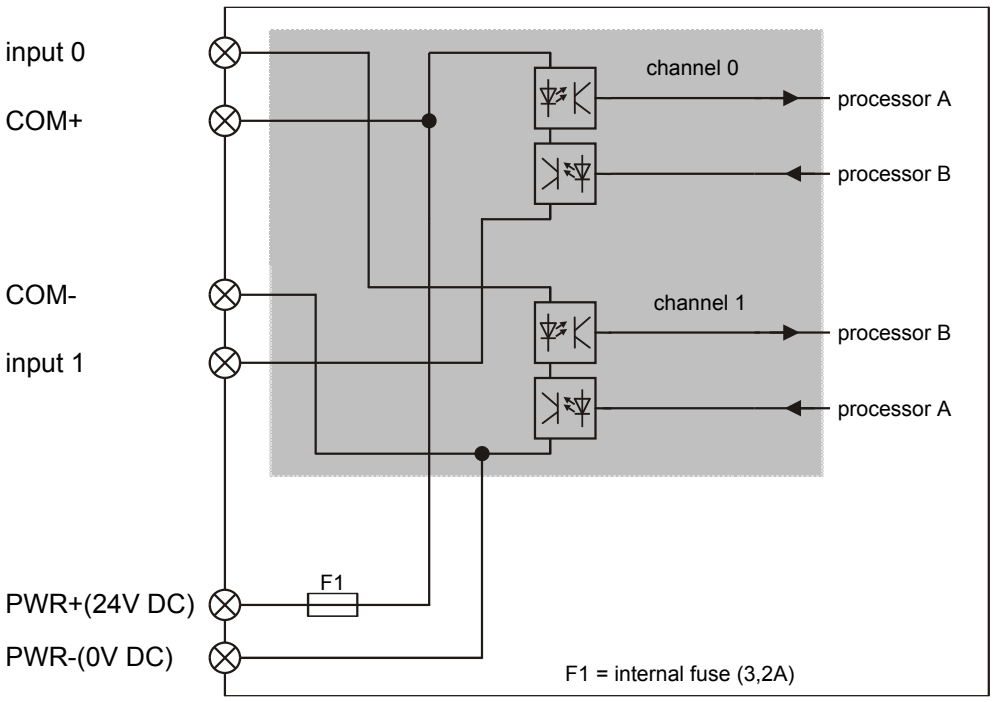


Figure 3-21 Basic circuit diagram for inputs for floating sensors

Basic circuit diagram for input for non-floating (current-sourcing) sensors

The chart shows the basic structure of an input for floating sensors. The grey shaded area exists multiply depending on the input module used.

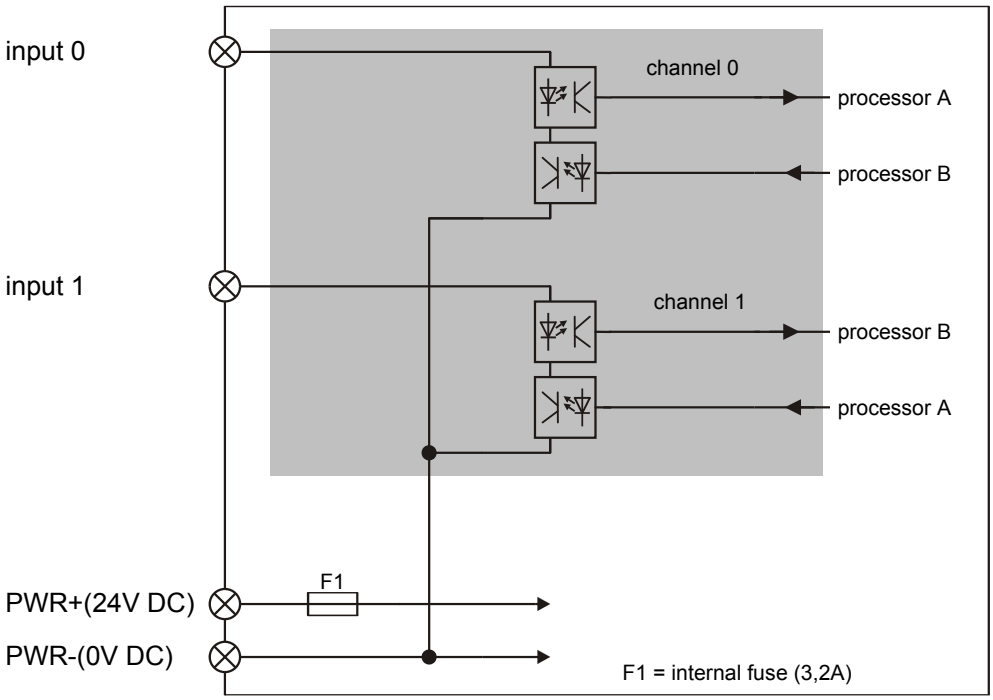


Figure 3-22 Basic circuit diagram inputs for non-floating sensors

3.8.2 Safety functions

Test pulses

The inputs are self-monitored by test pulses. Pulses of 450 to 500µs are generated every 15ms. The interrogation of the input channel is suspended for this brief period.

An external switching process by the connected sensory analysis is taken into consideration during this check.

The self-monotoring of external sensors, such as AOPDs, is not impaired by the output of the test pulses.

Channel monitoring

In the case of 2 channel control the potentials of the individual channels are checked for equality. External switching processes during the test are taken into consideration.

Voltage monitoring

Every input module has an internal monitoring system for the connected 24V DC power supply, generating an error message in the case of an error.

Error messages

If errors occur during the above described measures or other internal/external errors, these are read out by error messages. Every error message can be generated by the A/B processor. The prefix (A/B) shows the processor from which the message comes. The table below provides an excerpt from the possible error messages of the input modules. A complete list of all error messages is provided in Chapter 8.3.3.

8.3.3 / 8-5

Test	Error code	Meaning
Supply sensors	A/B 043-14	Supply voltage for the sensors too low or non-existent
Test pulses	A/B 043-21	Test pulses not detected
Channel monitoring	A/B 043-22	Discrepancy of the input channels (in the case of 2 channel control)

Table 3-18 Selected error messages of the input modules

3.9 Output modules

3.9.1 General description

All output modules are self-monitoring and comply with category 4 to DIN EN 954-1 (the overall category will depend on the external circuitry).

The outputs may be parameterised as 1 or 2 channel using **SFL-SOFT**.

Basic circuit diagram for semi-conductor output

The diagram shows the basic structure of a semi-conductor output. The grey shaded part of the circuit exists multiply depending on output module used.

Suitable protective measures (e.g. free-wheeling diode) are to be taken where inductive loads are switched.

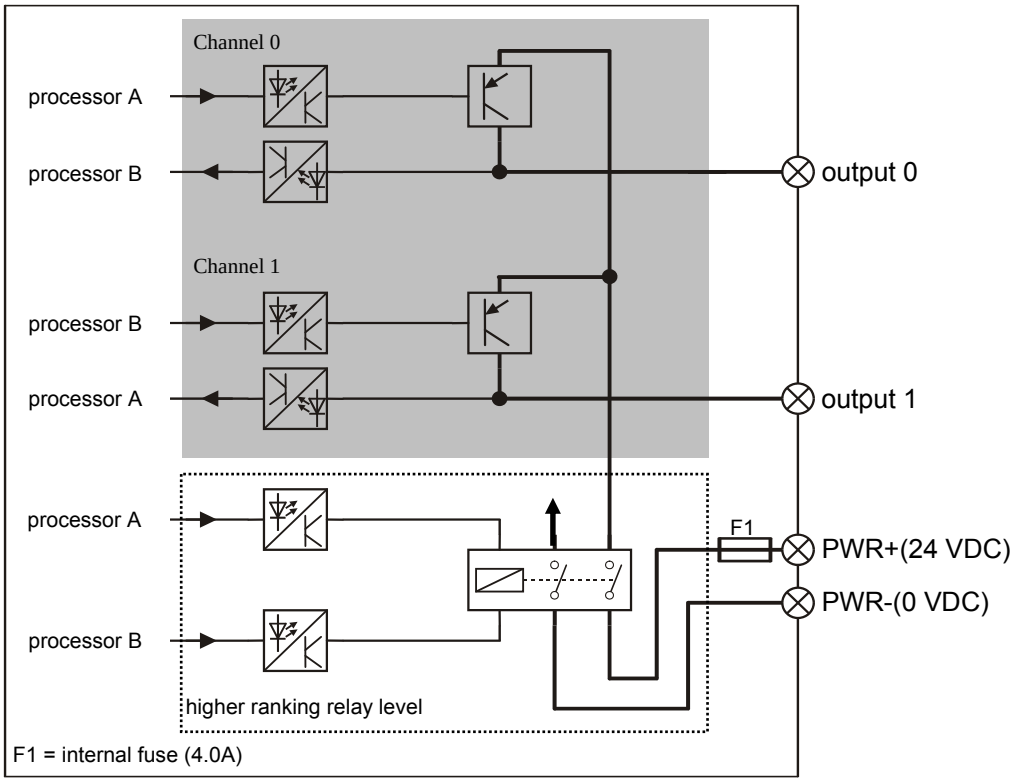


Figure 3-23 Basic circuit diagram for semi-conductor output module

Basic circuit diagram for relay output

The chart shows the basic structure of a relay output. The grey shaded part of the circuit exists twice in the output module **SFL-RELAY** The channel 0/1 can only be controlled in pairs; the same applies to channels 2/3.

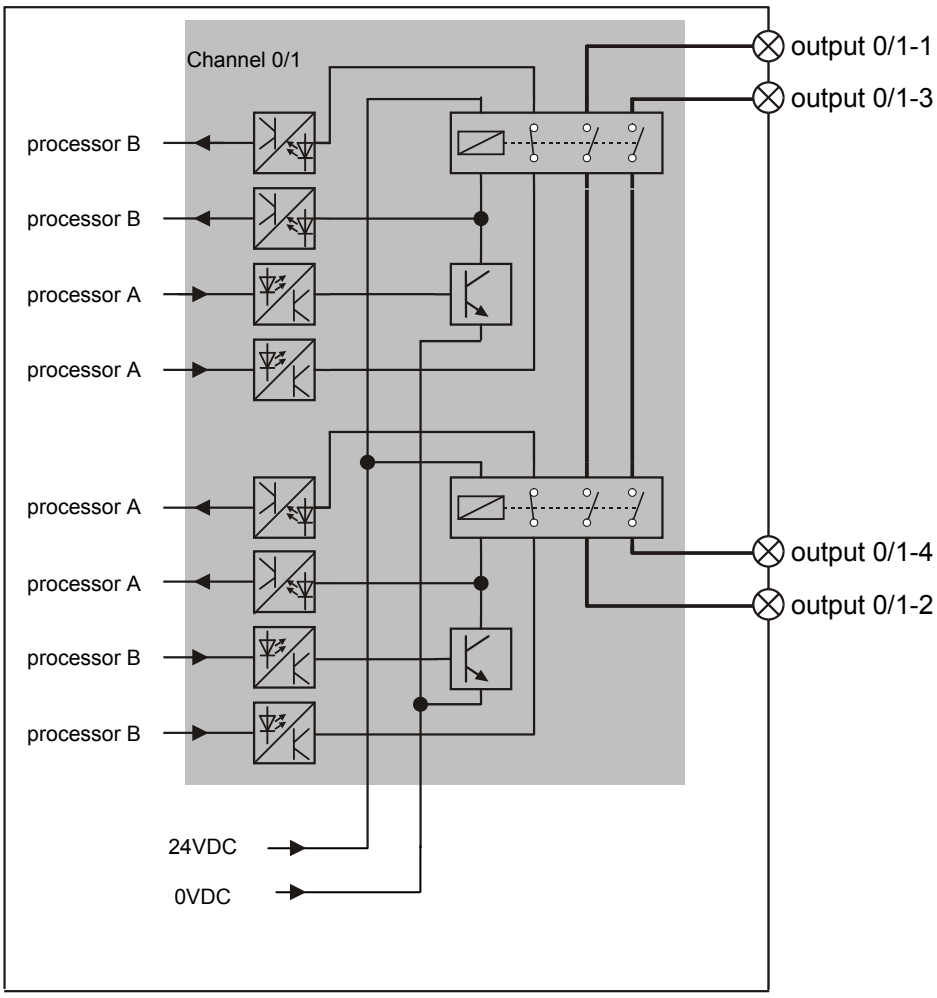


Figure 3-24 Basic circuit diagram for relay outputs

The relay outputs must be opened at least once per year. This can be done manually by actuating the protective device on the input side or automatically in the user program.

The relay outputs must be fused by means of an appropriately dimensioned fuse (max. 4A slow blowing).

3.9.2 Safety functions

Test pulses

The inputs are self-monitored by test pulses. Pulses of 450 to 500µs are generated every 15ms. The output is switched off for this brief period.

It must be insured that the downstream actuators cannot be affected by the test pulse.

Channel monitoring

The correct switching of the output stage is monitored crosswise. This means that processor A/B monitors the switching process of processor B/A. In the case of semi-conductor outputs the power supply to the actuators can be interrupted in the case of error using a shut-down relay. Output modules with relay outputs also check the switching function of the positively driven internal relay.

Voltage monitoring

Every output module has an internal monitoring system of the connected power supply of the actuators and reads out an error message in the case of an error.

Error messages

If errors occur during the above described measures or other internal/external errors, these are read out by error messages. Every error message can be generated by the processor A/B. The prefix (A/B) shows the processor from which the message comes. The table below provides an excerpt from the possible error messages of the input modules. A complete list of all error messages is provided in Chapter 8.3.3.

8.3.3 / 8-5

Test	Error code	Meaning
Supply actuators	A/B 043-14	Supply voltage for the actuators too low or non-existent
Shut-down relays	A/B 043-15	Error in the internal shut-down relay (24V DC)
Test pulse	A/B 043-31	Test pulses not detected
Channel monitoring	A/B 043-32	Discrepancy in reading back the output level
Relay contact	A/B 043-33	Error in output relay

Table 3-19 Selected error messages of the output modules

3.10 SFL-SUB-MON

3.10.1 General description

Mode 1

If the **SFL** is in Mode 1, the **SFL-SUB-MON** module is the master for the group created by it (SUB-Master).
The **SFL-SUB-MON** module has 2 category 4 (to DIN EN 954-1) or PLe (to DIN EN ISO 13849-1) inputs. If one of the inputs is opened, then all outputs of the SUB-Master and the outputs of the group created by it move to safe state (see Table 3-10). Similarly all outputs of this SUB-Master group move to the safe state if an input is opened on the CPU module.

In order to activate the outputs, the reset input on the SUB-Master or on the CPU module must be actuated to reset the safety function. If the safety function is reset, the 'Master-on' output is activated. The 'Ready' output has no function in Mode 1. The 'ErrorReset' input serves to reset the entire system following an error (e.g. wiring)

If the reset input is used by the SUB-Master or CPU module, the requirements set out in Chapter 7.11.1 Manual resetting (Reset, acknowledgement) must be satisfied.

Mode 3

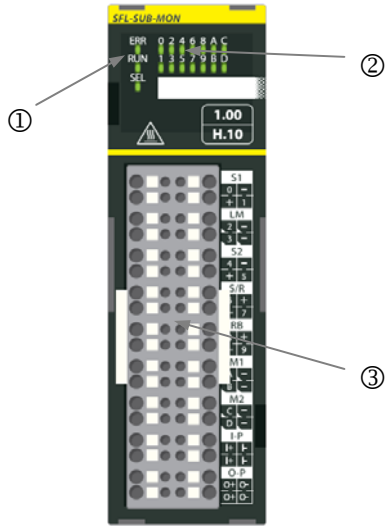
If the **SFL** is in Mode 3, then the SUB-Master module acts like a module with 4 x 2 inputs and 3 x 2 outputs.

Technical data

Position	Description
Name	**SFL-SUB-MON**
Operating voltage/current	24V DC ±10% / 70mA
Fusing	Internal fuse Input circuit: 3.2A Output circuit: 4.0A
Number of safety inputs	2 x 2 floating and 2 x 2 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Number of safety outputs	3 x 2
Output current	Max. 0.5A ohmic / output
Maximum switching voltage	24V DC
Connection plug	9 x 4-pin plug MORIMATSU M800A-01-xx 36-pin plug MORIMATSU M820-09-xx
Dimensions / weight	30 x 100 x 80 mm / (W/H/D) / 195g

Table 3-20 Technical data for **SFL-SUB-MON**

Housing description



① StatusLED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display I/O
00h - 0Fh = status of I/O

③ Terminal
00h - 0Fh = connection sensor / actuator
I+, O+ = power supply (24V DC)
I-, O- = power supply (0V DC)

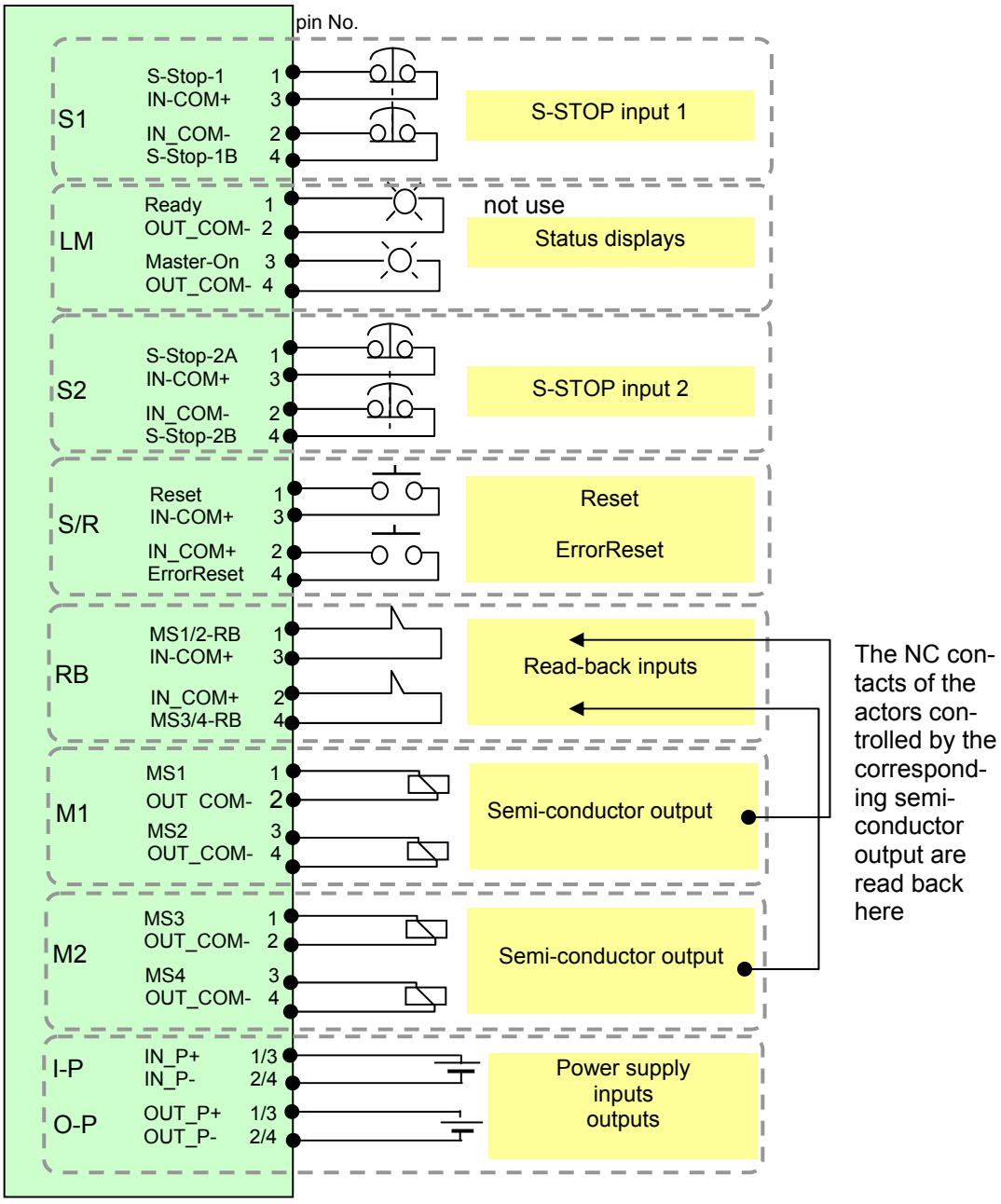
Figure 3-25 **SFL-SUB-MON**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
S1	00	1	0	S-Stop-1A	01	2	-	IN_COM-
	00	3	+	IN_COM+	01	4	1	S-Stop-1B
LM	02	1	2	Ready	02	2	-	OUT_COM-
	03	3	3	Master-On	03	4	-	OUT_COM-
S2	04	1	4	S-Stop-2A	05	2	-	IN_COM-
	04	3	+	IN_COM+	05	4	5	S-Stop-2B
S/R	06	1	6	Start	07	2	+	IN_COM+
	06	3	+	IN_COM+	07	4	7	Reset
RB	08	1	8	MS1/2-RB	09	2	+	IN_COM+
	08	3	+	IN_COM+	09	4	9	MS3/4-RB
M1	0A	1	A	MS1	0A	2	-	OUT_COM-
	0B	3	B	MS2	0B	4	-	OUT_COM-
M2	0C	1	C	MS3	0C	2	-	OUT_COM-
	0D	3	D	MS4	0D	4	-	OUT_COM-
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-21 Terminal designation **SFL-SUB-MON**

The pins no. 1 and 3 or 2 and 4 of the I-P or O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring modules.

SFL-SUB-MON terminal diagram for Mode 1



Terminal block pin number mapping

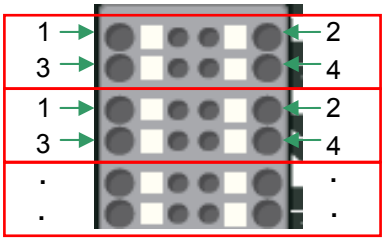


Figure 3-26 **SFL-SUB-MON** terminal diagram

3.11 Combined modules

3.11.1 General description

Of the combined modules the following 3 types are available

- **SFL-S-STP-E** inputs for floating sensors
- **SFL-S-STP-LC** inputs for non-floating sensors
- **SFL-S-STP-ELC** inputs for floating/non-floating sensors

If the **SFL** is in Mode 1, the outputs are switched to safe state as soon as an input is opened .

In order to activate the outputs, the reset input must be actuated on the CPU or SUB-MON module to reset the safety function.

If the **SFL** is in Mode 3, the combined modules act like a module with 3 x 2 inputs and 2 x 2 outputs.

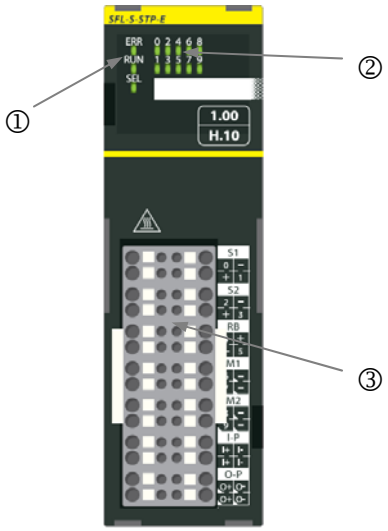
3.11.2 SFL-S-STP-E

Technical data

Position	Description
Name	**SFL-S-STP-E**
Operating voltage/current	24V DC ±10% / 70mA
Fusing	Internal fuse Input circuit: 3.2A Output circuit: 4.0A
Number of safety inputs	2 x 2 floating and 1 x 2 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Number of safety outputs	2 x 2
Output current	Max. 0.5A ohmic / output
Connection plug	7 x 4-pin plug MORIMATSU M800-S-01-xx 28-pin plug MORIMATSU M820A-07-xx
Dimensions / weight	30 x 100 x 80 mm / (W/H/D) / 185g

Table 3-22 Technical data **SFL-S-STP-E**

Housing description



① Status LED

ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display I/O

00 - 09h = status of I/O

③ Terminal

00h - 0Fh = connection sensor / actuator
I+, O+ = power supply (24V DC)
I-, O- = power supply (0V DC)

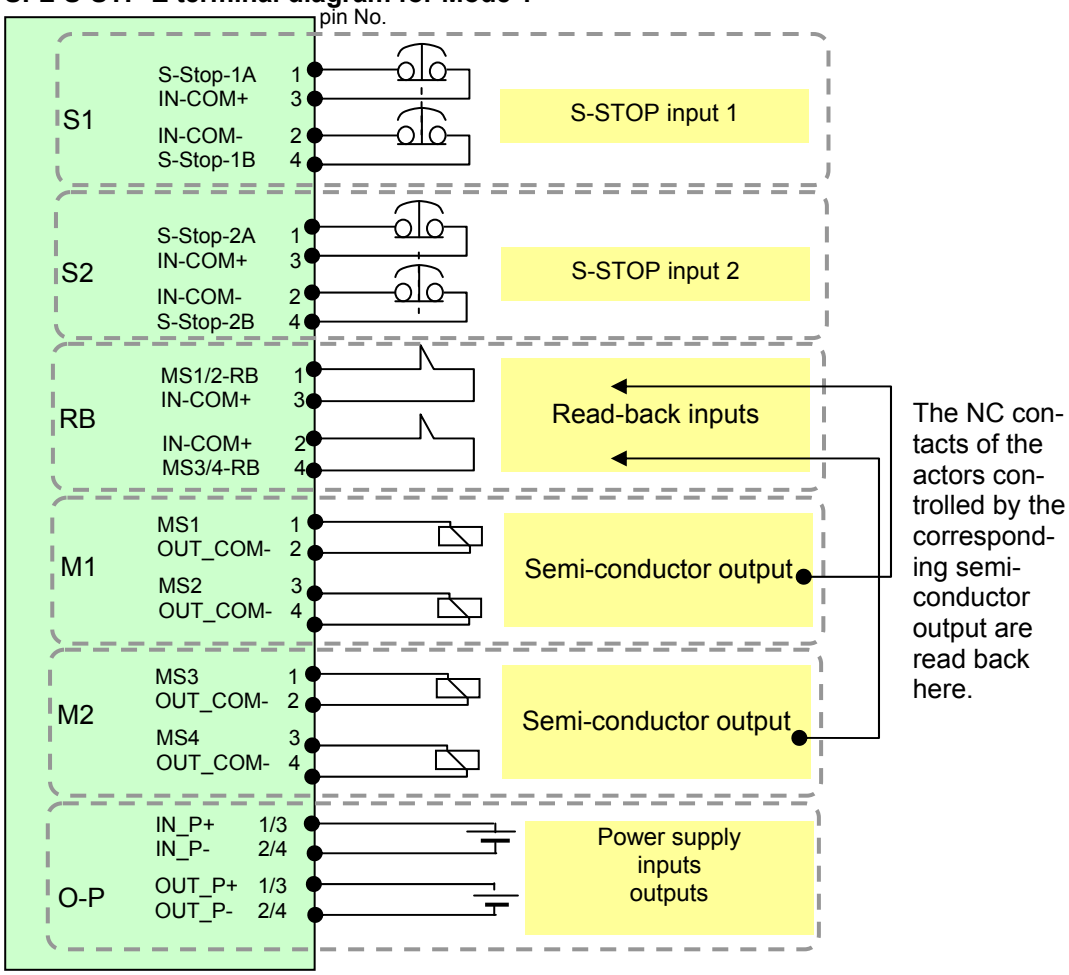
Figure 3-27 **SFL-S-STP-E**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
S1	00	1	0	S-Stop-1A	01	2	-	IN_COM-
	00	3	+	IN_COM+	01	4	1	S-Stop-1B
S2	02	1	2	S-Stop-2A	03	2	-	IN_COM-
	02	3	+	IN_COM+	03	4	3	S-Stop-2B
RB	04	1	4	MS1/2-RB	05	2	+	IN_COM+
	04	3	+	IN_COM+	05	4	5	MS3/4-RB
M1	06	1	6	MS1	06	2	-	OUT_COM-
	07	3	7	MS2	07	4	-	OUT_COM-
M2	08	1	8	MS3	08	2	-	OUT_COM-
	09	3	9	MS4	09	4	-	OUT_COM-
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-23 Terminal designation **SFL-S-STP-E**

The pins no. 1 and 3 or 2 and 4 of the I-P or O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring modules.

SFL-S-STP-E terminal diagram for Mode 1



Terminal block pin number mapping

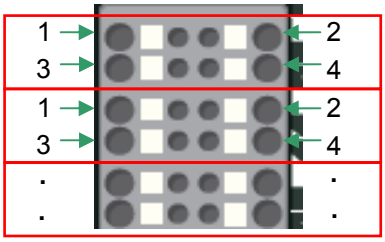


Figure 3-28 **SFL-S-STP-E** terminal diagram

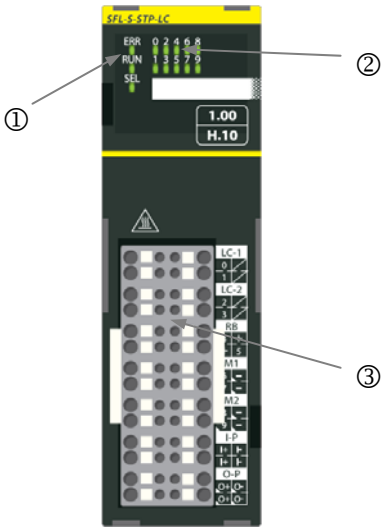
3.11.3 SFL-S-STP-LC

Technical data

Position	Description
Name	**SFL-S-STP-LC**
Operating voltage/current	24V DC ±10% / 70mA
Fusing	Internal fuse Input circuit: 3.2A Output circuit: 4.0A
Number of safety inputs	3 x 2 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Number of safety outputs	2 x 2
Output current	Max. 0.5A ohmic / output
Connection plug	7 x 4-pin plug MORIMATSU M800A-01-xx 28-pin plug MORIMATSU M820A-09-xx
Dimensions / weight	30 x 100 x 80 mm / (W/H/D) / 185g

Table 3-24 Technical data **SFL-S-STP-LC**

Housing description



① Status LED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display I/O
00h - 09h = Status der I/O

③ Terminal
00h - 0Fh = connection sensor / actuator
I+, O+ = power supply (24V DC)
I-, O- = power supply (0V DC)

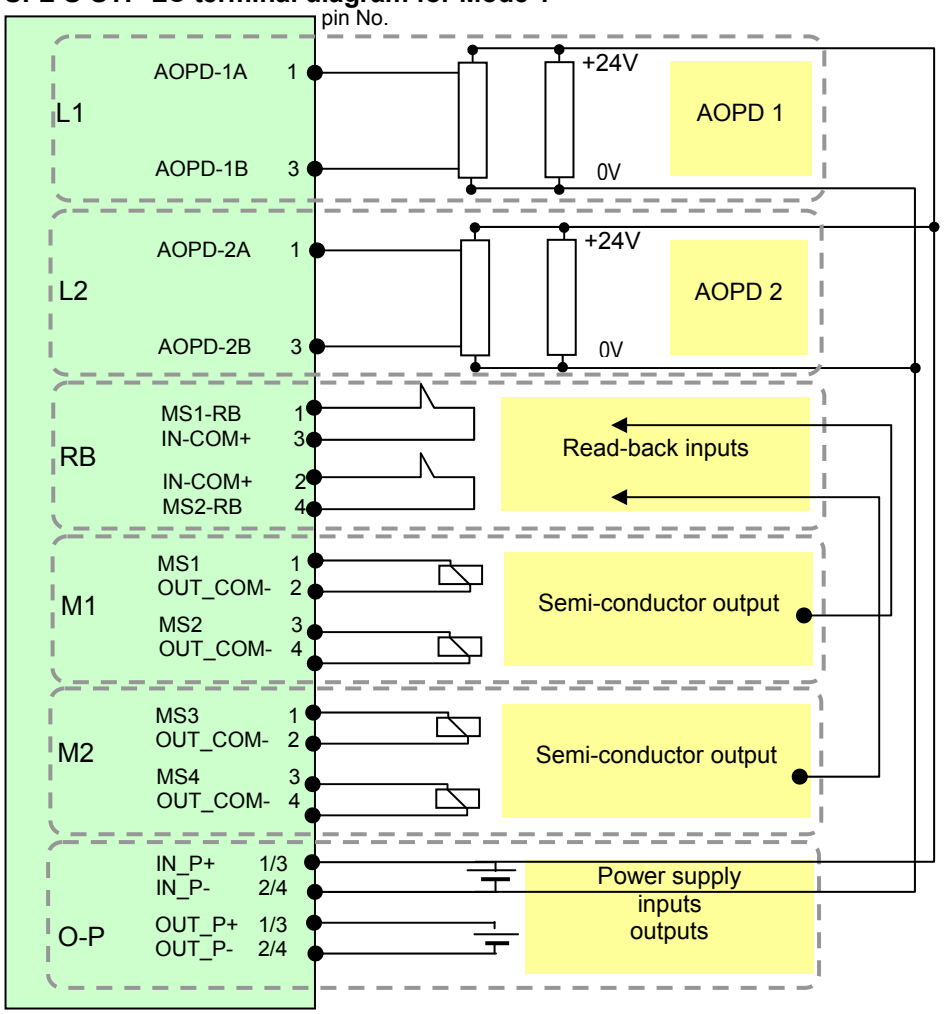
Figure 3-29 **SFL-S-STP-LC**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
L1	00	1	0	AOPD-1A		2		
	01	3	1	AOPD-1B		4		
L2	02	1	2	AOPD-2A		2		
	03	3	3	AOPD-2B		4		
RB	04	1	4	MS1-RB	05	2	+	IN_COM+
	04	3	+	IN_COM+	05	4	5	MS2-RB
M1	06	1	6	MS1	06	2	-	OUT_COM-
	07	3	7	MS2	07	4	-	OUT_COM-
M2	08	1	8	MS3	08	2	-	OUT_COM-
	09	3	9	MS4	09	4	-	OUT_COM-
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-25 Terminal designation **SFL-S-STP-LC**

The pins no. 1 and 3 or 2 and 4 of the I-P or O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring modules.

SFL-S-STP-LC terminal diagram for Mode 1



The NC contacts of the actors controlled by the corresponding semi-conductor output are read back here.

The power supply of the inputs must have the same potential as the power supply of the AOPDs.

Terminal block pin number mapping

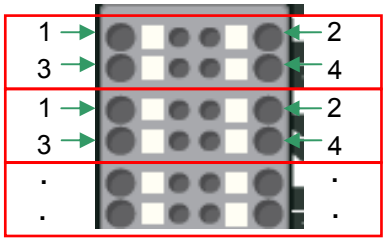


Figure 3-30 **SFL-S-STP-LC** terminal diagram

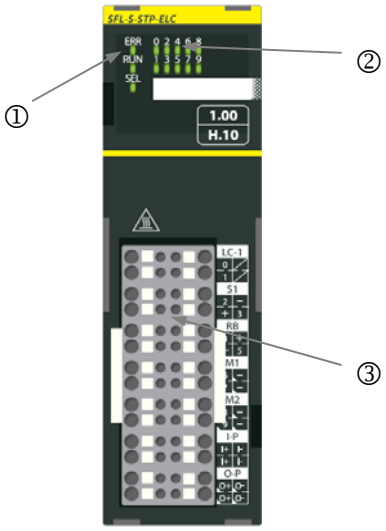
3.11.4 SFL-S-STP-ELC

Technical data

Position	Description
Name	**SFL-S-STP-ELC**
Operating voltage/current	24V DC ±10% / 70mA
Fusing	Internal fuse Input circuit: 3.2A Output circuit: 4.0A
Number of safety inputs	1 x 2 floating and 2 x 2 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Number of safety outputs	2 x 2
Output current	Max. 0.5A ohmic / output
Connection plug	7 x 4-pin plug MORIMATSU M820A-01-xx 28-pin plug MORIMATSU M820A-07-xx
Dimensions / weight	30 x 100 x 80 mm / (W/H/D) / 185g

Table 3-26 Technical data **SFL-S-STP-ELC**

Housing description



① Status LED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display I/O
00h - 09h = status of I/O

③ Terminal
00h - 0Fh = connection sensor / actuator
I+, O+ = power supply (24V DC)
I-, O- = power supply (0V DC)

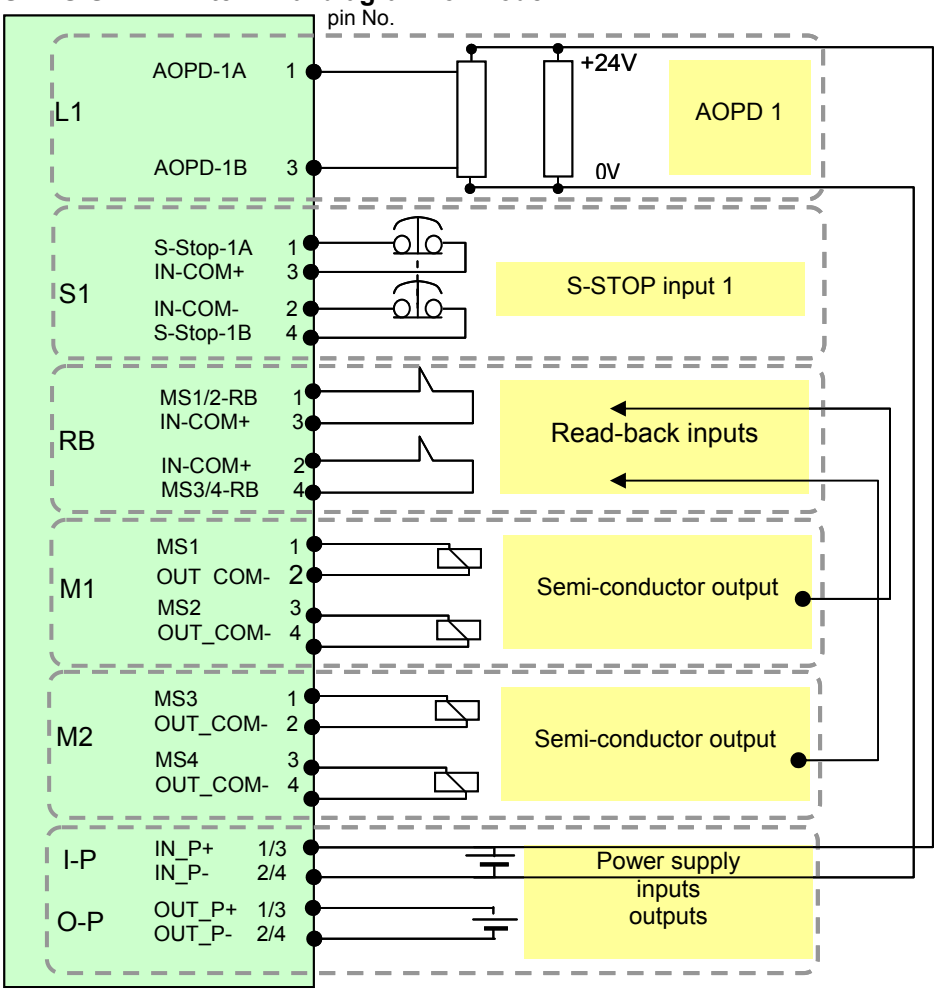
Figure 3-31 **SFL-S-STP-ELC**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
L1	00	1	0	AOPD-1A		2		
	01	3	1	AOPD-1B		4		
S1	02	1	2	S-Stop-2A	03	2	-	IN_COM-
	02	3	+	IN_COM+	03	4	3	S-Stop-2B
RB	04	1	4	MS1-RB	05	2	+	IN_COM+
	04	3	+	IN_COM+	05	4	5	MS2-RB
M1	06	1	6	MS1	06	2	-	OUT_COM-
	07	3	7	MS2	07	4	-	OUT_COM-
M2	08	1	8	MS3	08	2	-	OUT_COM-
	09	3	9	MS4	09	4	-	OUT_COM-
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-27 Terminal designation **SFL-S-STP-ELC**

The pins no. 1 and 3 or 2 and 4 of the I-P or O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring modules.

SFL-S-STP-ELC terminal diagram for Mode 1



The NC contacts of the actors controlled by the corresponding semiconductor output are read back here.

The power supply of the inputs must have the same potential as the power supply of the AOPDs.

Terminal block pin number mapping

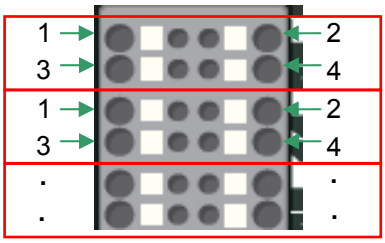


Figure 3-32 **SFL-S-STP-ELC** terminal diagram

3.12 Input modules

3.12.1 General description

Of the input modules the following 2 types are available

- **SFL-S-IN-E** inputs for floating sensors
- **SFL-S-IN-LC** inputs for non-floating sensors

If the **SFL** is in Mode 1, the outputs of the assigned group are switched to the safe state as soon as an input is opened.

If the **SFL** is in Mode 3, than the input modules act like modules with 8 x 2 inputs.

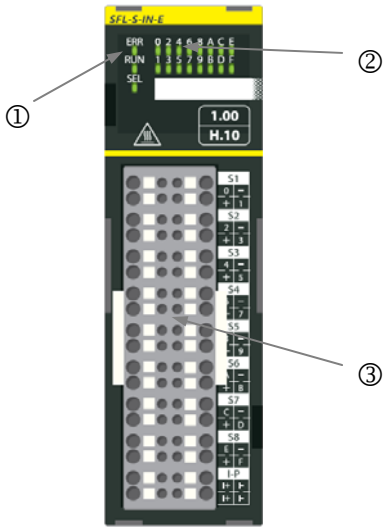
3.12.2 SFL-S-IN-E

Technical data

Position	Description
Name	**SFL-S-IN-E**
Operating voltage/current	24V DC ±10% / 62mA
Fusing	Internal fuse 3.2A
Number of safety inputs	8 x 2 floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Connection plug	9 x 4-pin plug MORIMATSU M820A-01-xx 36-pin plug MORIMATSU M820A-09-xx
Dimensions / weight	30 x 100 x 80 mm (W/H/D) / 190g

Table 3-28 Technical data **SFL-S-IN-E**

Housing description



① Status LED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display inputs
00h - 0Fh = status of inputs

③ Terminal
00h - 0Fh = connection sensor
I+ = power supply (24V DC)
I- = power supply (0V DC)

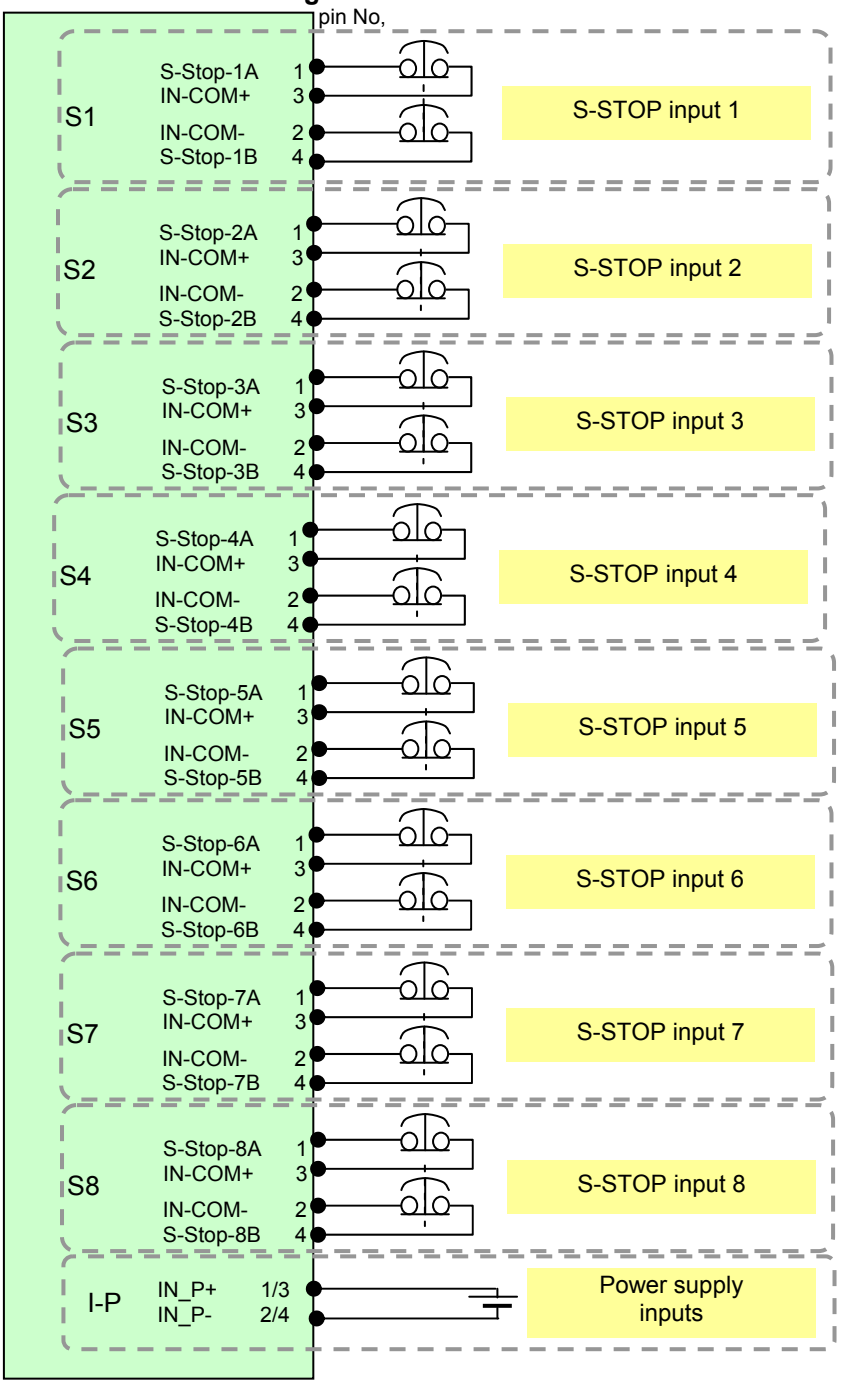
Figure 3-33 **SFL-S-IN-E**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
S1	00	1	0	S-Stop-1A	01	2	-	IN_COM-
	00	3	+	IN_COM+	01	4	1	S-Stop-1B
S2	02	1	2	S-Stop-2A	03	2	-	IN_COM-
	02	3	+	IN_COM+	03	4	3	S-Stop-2B
S3	04	1	4	S-Stop-3A	05	2	-	IN_COM-
	04	3	+	IN_COM+	05	4	5	S-Stop-3B
S4	06	1	6	S-Stop-4A	07	2	-	IN_COM-
	06	3	+	IN_COM+	07	4	7	S-Stop-4B
S5	08	1	8	S-Stop-5A	09	2	-	IN_COM-
	08	3	+	IN_COM+	09	4	9	S-Stop-5B
S6	0A	1	A	S-Stop-6A	0B	2	-	IN_COM-
	0A	3	+	IN_COM+	0B	4	B	S-Stop-6B
S7	0C	1	C	S-Stop-7A	0D	2	-	IN_COM-
	0C	3	+	IN_COM+	0D	4	D	S-Stop-7B
S8	0E	1	E	S-Stop-8A	0F	2	-	IN_COM-
	0E	3	+	IN_COM+	0F	4	F	S-Stop-8B
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-

Table 3-29 Terminal designation **SFL-S-IN-E**

The pins no. 1 and 3 or 2 and 4 of the I-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-S-IN-E terminal diagram for Mode 1



Terminal block pin number mapping

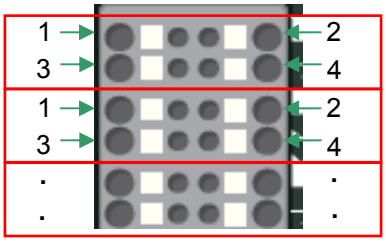


Figure 3-34 **SFL-S-IN-E** terminal diagram

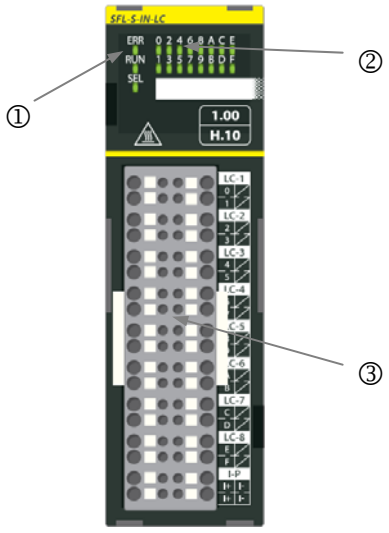
3.12.3 SFL-S-IN-LC

Technical data

Position	Description
Name	**SFL-S-IN-LC**
Operating voltage/current	24V DC ±10% / 62mA
Fusing	Internal fuse 3.2A
Number of safety inputs	8 x 2 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High/Low level	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Minimum pulse duration for safe detection	15ms
Connection plug	9 x 4-pin plug MORIMATSU M820A-01-xx 36-pin plug MORIMATSU M820A-09-xx
Dimensions / weight	30 x 100 x 80 mm / (W/H/D) / 190g

Table 3-30 Technical data **SFL-S-IN-LC**

Housing description



① StatusLED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Status LED
00h - 0Fh = status of I/O

③ Terminal
00h - 0Fh = connection sensor / actuator
I+ = power supply (24V DC)
I- = power supply (0V DC)

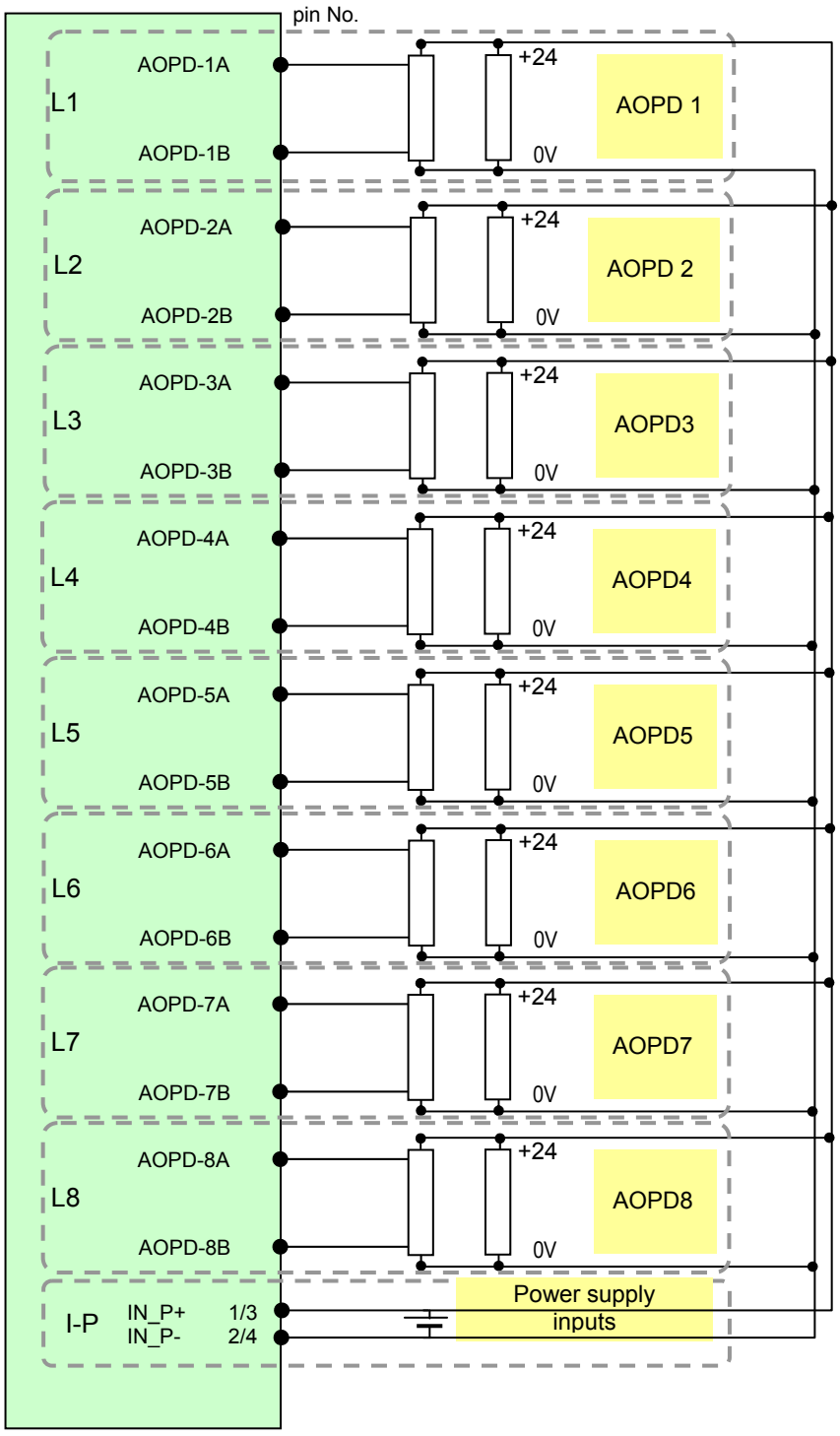
Figure 3-35 **SFL-S-IN-LC**

	I/O Address	Pin no.	Print name	Terminal name	I/O Address	Pin no.	Print name	Terminal name
L1	00	1	0	AOPD-1A		2		
	01	3	1	AOPD-1B		4		
L2	02	1	2	AOPD-2A		2		
	03	3	3	AOPD-2B		4		
L3	04	1	4	AOPD-3A		2		
	05	3	5	AOPD-3B		4		
L4	06	1	6	AOPD-4A		2		
	07	3	7	AOPD-4B		4		
L5	08	1	8	AOPD-5A		2		
	09	3	9	AOPD-5B		4		
L6	0A	1	A	AOPD-6A		2		
	0B	3	B	AOPD-6B		4		
L7	0C	1	C	AOPD-7A		2		
	0D	3	D	AOPD-7B		4		
L8	0E	1	E	AOPD-8A		2		
	0F	3	F	AOPD-8B		4		
I-P		1	I+	IN_P+		2	I-	IN_P-
		3	I+	IN_P+		4	I-	IN_P-

Table 3-31 Terminal designation **SFL-S-IN-LC**

The pins no. 1 and 3 or 2 and 4 of the I-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-S-IN-LC terminal diagram for Mode 1



Terminal block pin number mapping

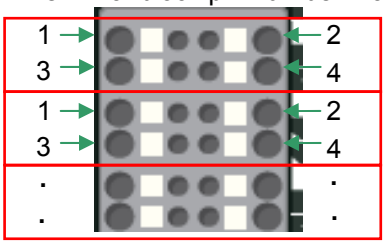


Figure 3-36 **SFL-S-IN-LC** terminal diagram

3.13 Relay module

3.13.1 General description

If the **SFL** is in Mode 1, the relay outpute are switched to safe state as soon as an input of the assigned group opens.

In order to reactivate the outputs, the reset input at the CPU module or SUB-MON module must be actuated to reset the safety function.

If the **SFL** is in Mode 3, then the output module acts like a module with 2 x 2 relay outputs.

The relay outputs must be opened at least once per year. This can be done manually by actuating the protective device on the input side or automatically in the user program.

The relay outputs must be fused by means of an appropriately dimensioned fuse (max. 4A slow blowing).

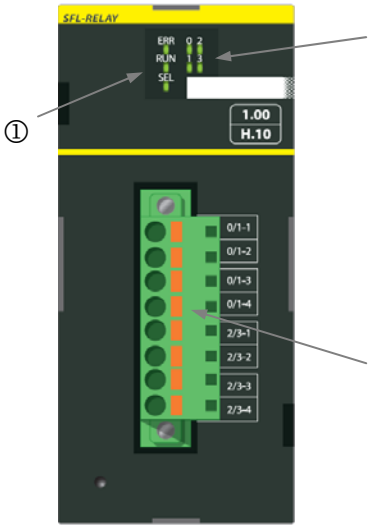
3.13.2 SFL-RELAY

Technical data

Position	Description
Name	**SFL-RELAY**
Operating voltage/current	24V DC ±10% / 107mA
Number of safety outputs	2 x 2 floating relay outputs
Output current	Max 4A ohmic / output External fuse: max. 4A slow blowing
Switching voltage	24V DC
Connection plug	8-Pin plug PHOENIX: FKC2.5/8-GF-5.08
Dimensions / weight	45 x 100 x 80 mm (B/H/ D) / 265g

Table 3-32 Technical data **SFL-RELAY**

Housing description



① Status LED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display I/O
00h - 03h = status of relay

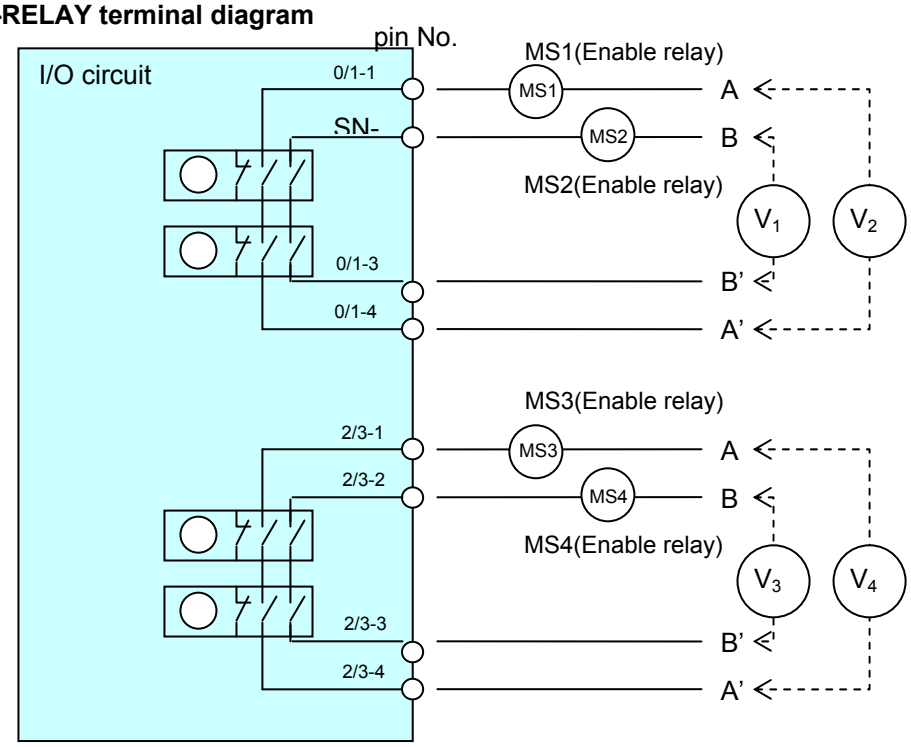
③ Terminal
00h - 03h = connection actuator

Figure 3-37 **SFL-RELAY**

I/O Address	Pin no.	Print name	Terminal name
00 01	1	0/1-1	RELAY OUT 1
	2	0/1-2	
	3	0/1-3	RELAY OUT 2
	4	0/1-4	
02 03	5	2/3-1	RELAY OUT 3
	6	2/3-2	
	7	2/3-3	RELAY OUT 4
	8	2/3-4	

Table 3-33 Terminal designation **SFL-RELAY**

SFL-RELAY terminal diagram



Terminal block pin number mapping

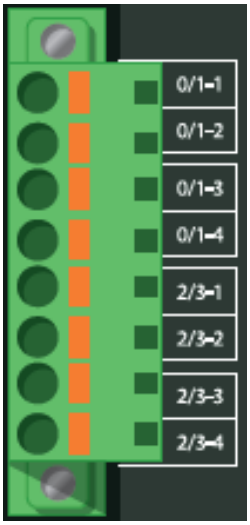


Figure 3-38 **SFL-RELAY** terminal diagram

3.14 Semi-conductor output module

3.14.1 General description

SFL is in Mode 1, the outputs are switched to the safe state as soon as one input of the assigned group is opened.

In order to reactivate the outputs, the reset input on the CPU or SUB-MON module must be actuated to reset the safety function.

If the **SFL** is in Mode 3, than the input module acts like a module with 8 x 2 inputs.

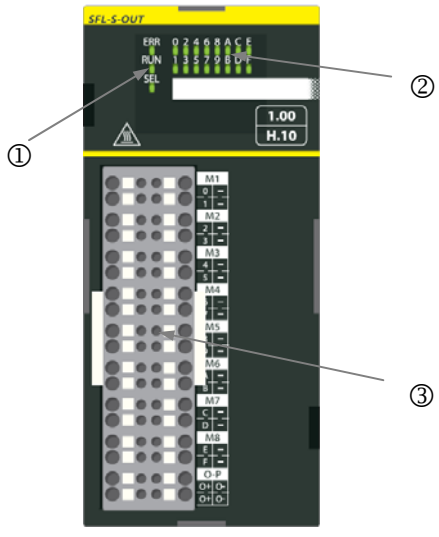
3.14.2 SFL-S-OUT

Technical data

Position	Description
Name	**SFL-S-OUT**
Operating voltage/current	24V DC ±10% / 74mA
Fusing	Internal fuse 4 x 4.0A
Number of safety outputs	8 x 2
Output current	Max. 0.3A ohmic load
Switching voltage	24V DC
Connection plug	9 x 4-pin plug MORIMATSU M820A-01-xx 36-pin plug MORIMATSU M820A-09-xx
Dimensions / weight	45 x 100 x 80 mm (B/H/ D) / 250g

Table 3-34 Technical data **SFL-S-OUT**

Housing description



① StatusLED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Status LED
00h - 0Fh = status of outputs

③ Terminal
00h - 0Fh = connection actuator
O+ = power supply (24V DC)
O- = power supply (0V DC)

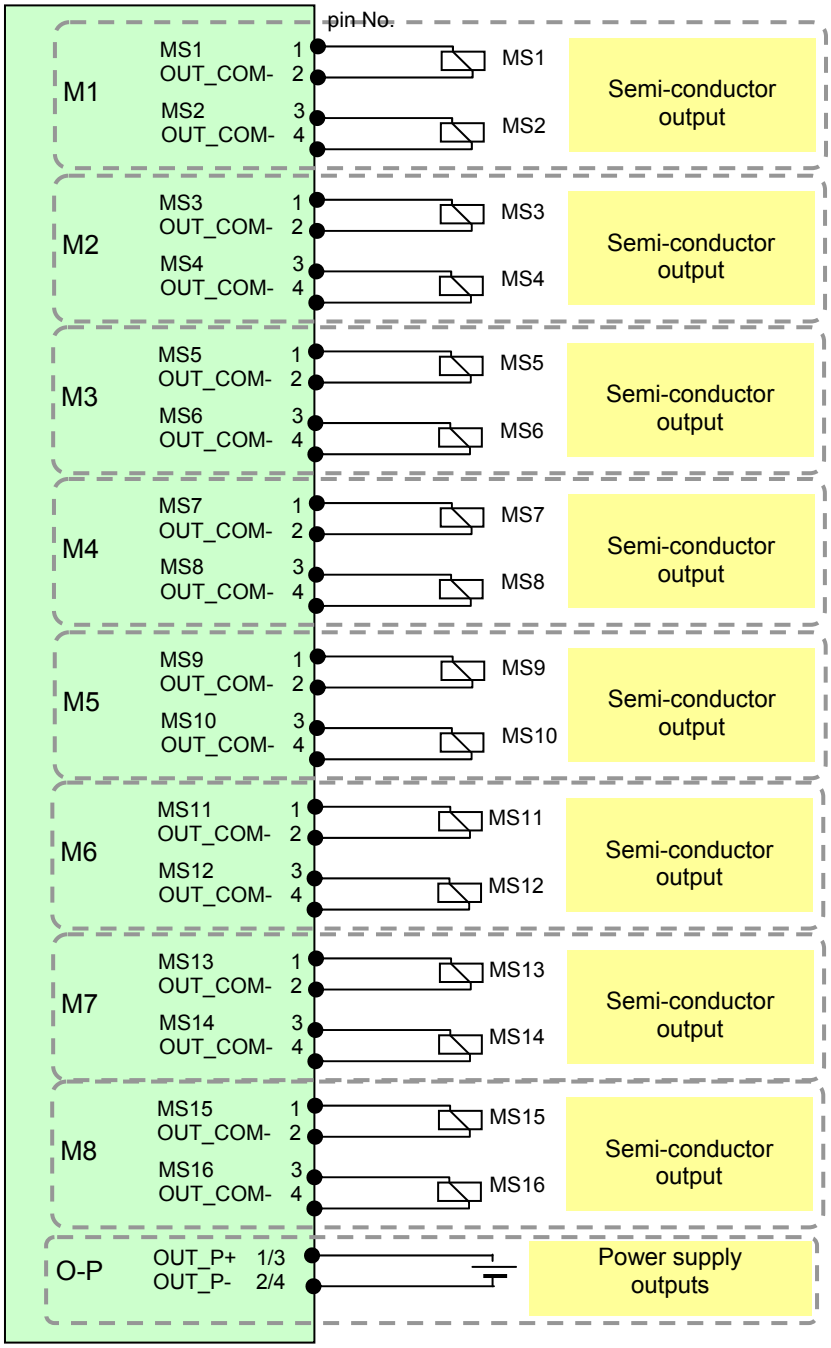
Figure 3-39 **SFL-S-OUT**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
M1	00	1	0	MS1	00	2	-	OUT_COM-
	01	3	1	MS2	01	4	-	OUT_COM-
M2	02	1	2	MS3	02	2	-	OUT_COM-
	03	3	3	MS4	03	4	-	OUT_COM-
M3	04	1	4	MS5	04	2	-	OUT_COM-
	05	3	5	MS6	05	4	-	OUT_COM-
M4	06	1	6	MS7	06	2	-	OUT_COM-
	07	3	7	MS8	07	4	-	OUT_COM-
M5	08	1	8	MS9	08	2	-	OUT_COM-
	09	3	9	MS10	09	4	-	OUT_COM-
M6	0A	1	A	MS11	0A	2	-	OUT_COM-
	0B	3	B	MS12	0B	4	-	OUT_COM-
M7	0C	1	C	MS13	0C	2	-	OUT_COM-
	0D	3	D	MS14	0D	4	-	OUT_COM-
M8	0E	1	E	MS15	0E	2	-	OUT_COM-
	0F	3	F	MS16	0F	4	-	OUT_COM-
O-P		1	O+	OUT_P+		2	O-	OUT_P-
		3	O+	OUT_P+		4	O-	OUT_P-

Table 3-35 Terminal designation **SFL-S-OUT**

The pins no. 1 and 3 or 2 and 4 of the O-P connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-S-OUT terminal diagram for Mode 1



Terminal block pin number mapping

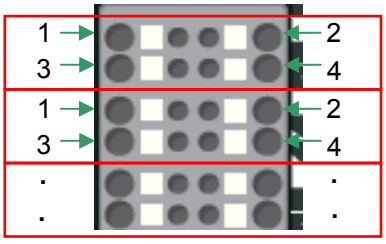


Figure 3-40 **SFL-S-OUT** terminal diagram

3.15 Operational input

3.15.1 General description

The operational input module only has one microprocessor. The inputs are operated using 1 channel only.

Short circuits between the inputs are not checked and additional self-tests are not performed.

Basic circuit diagram of operational input

The chart shows the basic structure of an operational input. The grey shaded part of the circuit multiply exists.

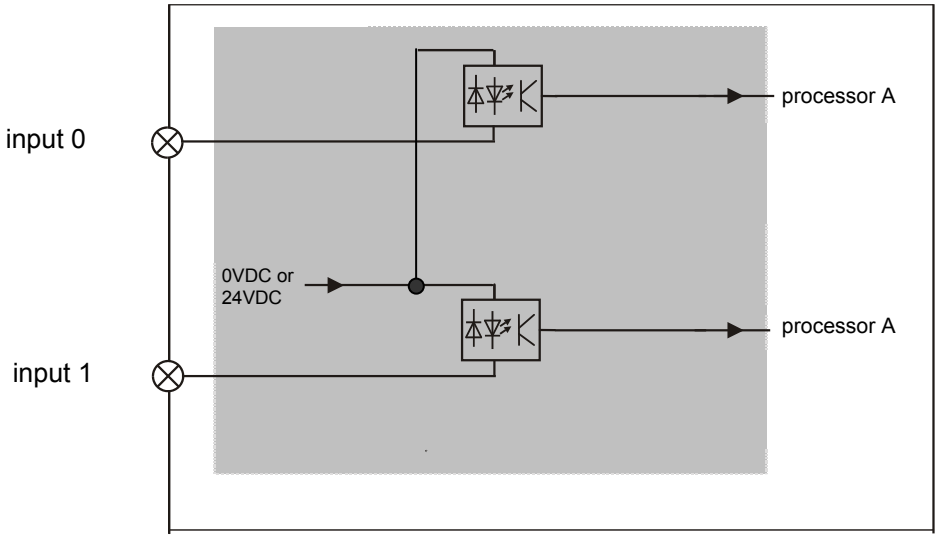


Figure 3-41 Basic circuit diagram for operational inputs

3.16 Operational output

3.16.1 General description

The operational output module only has one microprocessor. The outputs are operated using 1 channel only.

Short circuits between the outputs are not checked and additional self-tests are not performed.

Basic circuit diagram of operational output

The chart shows the basic structure of an operational output. The grey shaded part of the circuit multiply exists.

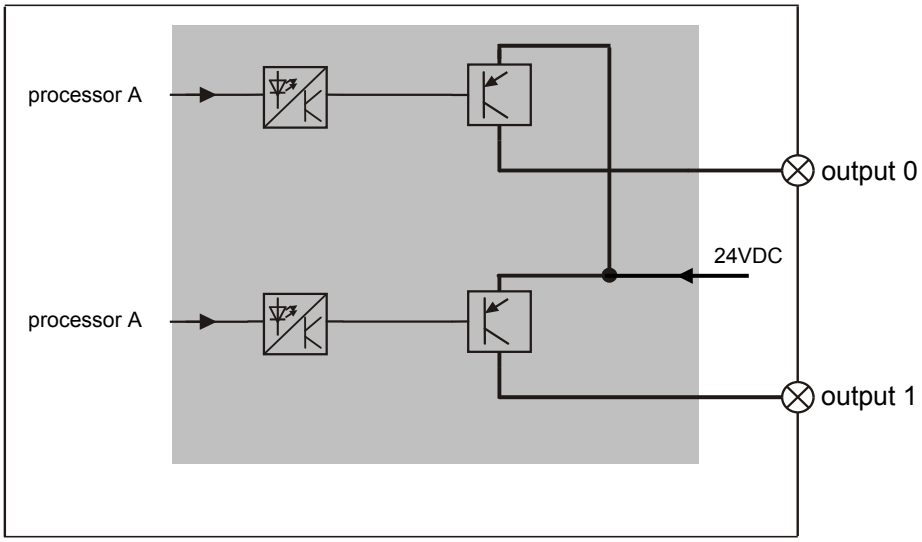


Figure 3-42 Basic circuit diagram for operational outputs

3.17 Operational input module

3.17.1 General description

If the **SFL** is in Mode 1, the operational input module is without function.

If the **SFL** is in Mode 3, then the operational input module acts as a module with 16 inputs.

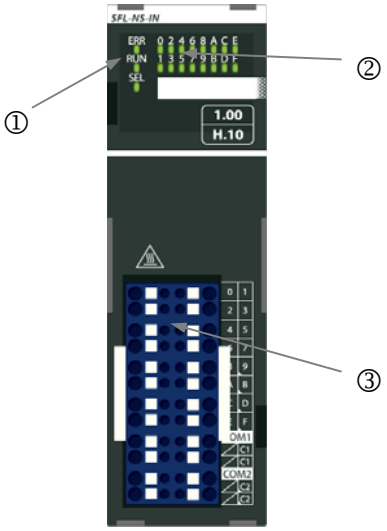
3.17.2 SFL-NS-IN

Technical data

Position	Description
Name	**SFL-NS-IN**
Operating voltage/current	24V DC ±10% / 25mA
Number of operational inputs	16 non-floating
Input resistance	Approx. 4.7kΩ
Input current	5mA
High-/Low-Pegel	H: >18V / >3.5mA L: <4.7V / <0.5mA
Minimum pulse duration for possible detection	1.4ms (input filter 0.7ms)
Connection plug	6 x 4-pin plug MORIMATSU M820A-01-xx 24-pin plug MORIMATSU M820A-06-xx
Dimensions / weight	30 x 100 x 80 mm (B/H/D) / 170g

Table 3-36 Technical data **SFL-NS-IN**

Housing description



① StatusLED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Display inputs
00h - 0Fh = status of inputs

③ Terminal
00h - 0Fh = connection sensor
C1, C2 = power supply

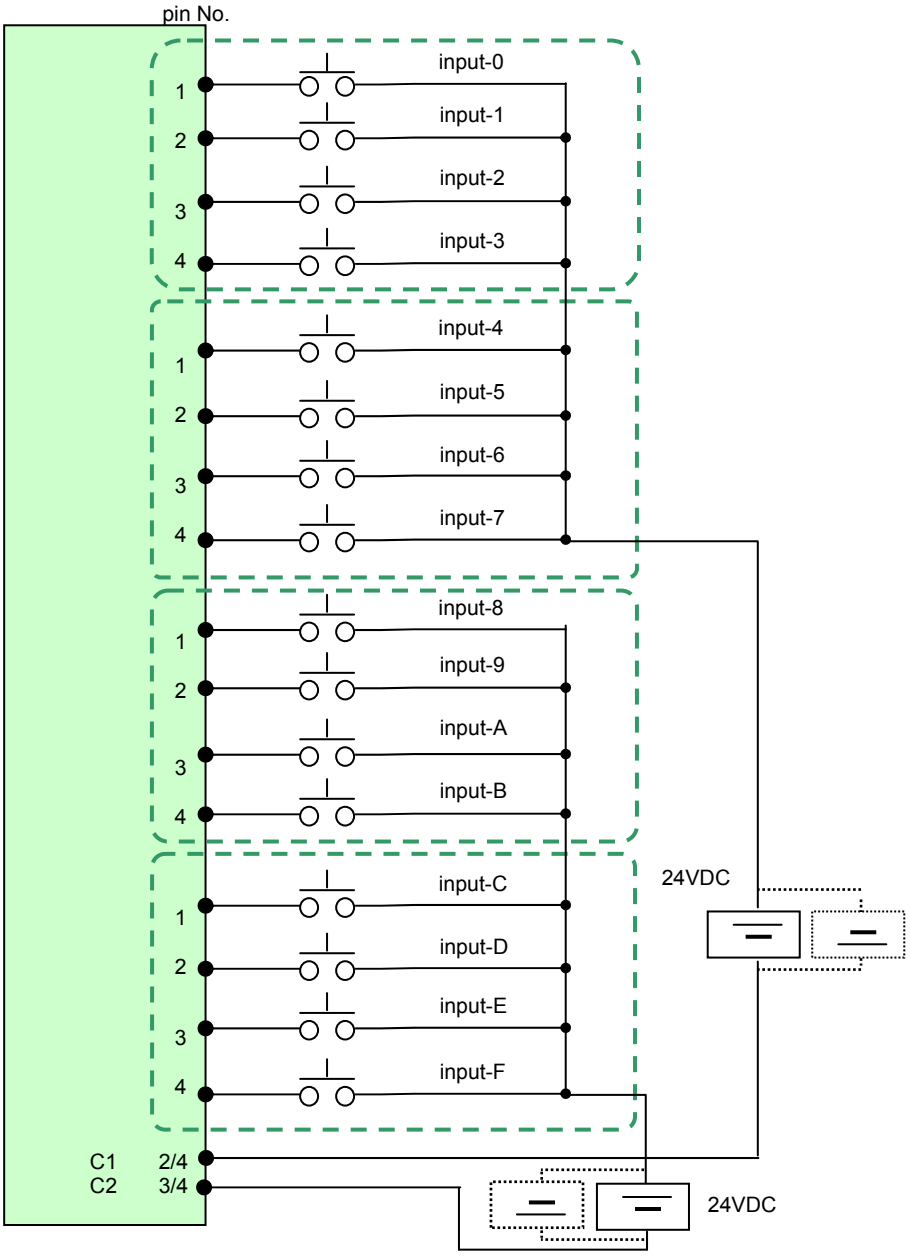
Figure 3-43 **SFL-NS-IN**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
	00	1	0	INPUT -0	01	2	1	INPUT -1
	02	3	2	INPUT -2	03	4	3	INPUT -3
	04	1	4	INPUT -4	05	2	5	INPUT -5
	06	3	6	INPUT -6	07	4	7	INPUT -7
	08	1	8	INPUT -8	09	2	9	INPUT -9
	0A	3	A	INPUT -A	0B	4	B	INPUT -B
	0C	1	C	INPUT -C	0D	2	D	INPUT -D
	0E	3	E	INPUT -E	0F	4	F	INPUT -F
COM1		1				2	C1	IN_P1
		3				4	C1	IN_P1
COM2		1				2	C2	IN_P2
		3				4	C2	IN_P2

Table 3-37 Terminal designation **SFL-NS-IN**

The pins no. 2 and 4 of the C1 or C2 connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-NS-IN terminal diagram



Terminal block pin number mapping

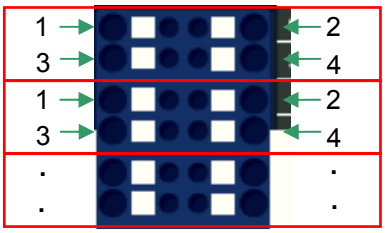


Figure 3-44 **SFL-NS-IN** terminal diagram

3.18 Operational output module

3.18.1 General description

If the **SFL** is in Mode 1, the operational output module is without function.

If the **SFL** is in Mode 3, then the operational output module acts as a module with 16 inputs.

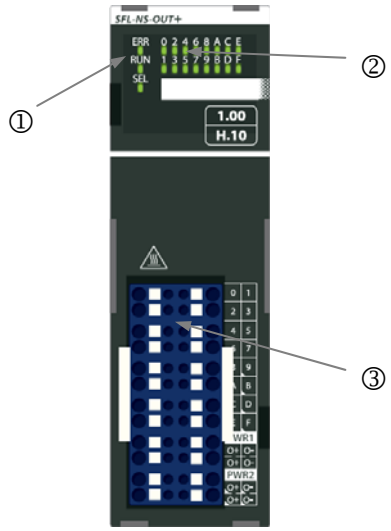
3.18.2 SFL-NS-OUT+

Technical data

Position	Description
Name	**SFL-NS-OUT+**
Operating voltage/current	24V DC ±10% / 48mA
Fusing	Internal fuse 4 x 3.2A
Number of operational outputs	16 (not safe)
Output current	Max 0.3A ohmic load
Connection plug	6 x 4-pin plug MORIMATSU M820A-01-xx 24-pin plug MORIMATSU M820A-06-xx
Dimensions / weight	30 x 100 x 80 mm (B/H/T) / 175g

Table 3-38 Technical data **SFL-NS-OUT+**

Housing description



① StatusLED
ERR = on : error / alarm
off : operation
RUN = lights up : user program active
off : user program inactive
SEL = initialisation phase

② Status LED
00h - 0Fh = status of inputs

③ Terminal
00h - 0Fh = connection actuator
O+ = power supply (24V DC)
O- = power supply (0V DC)

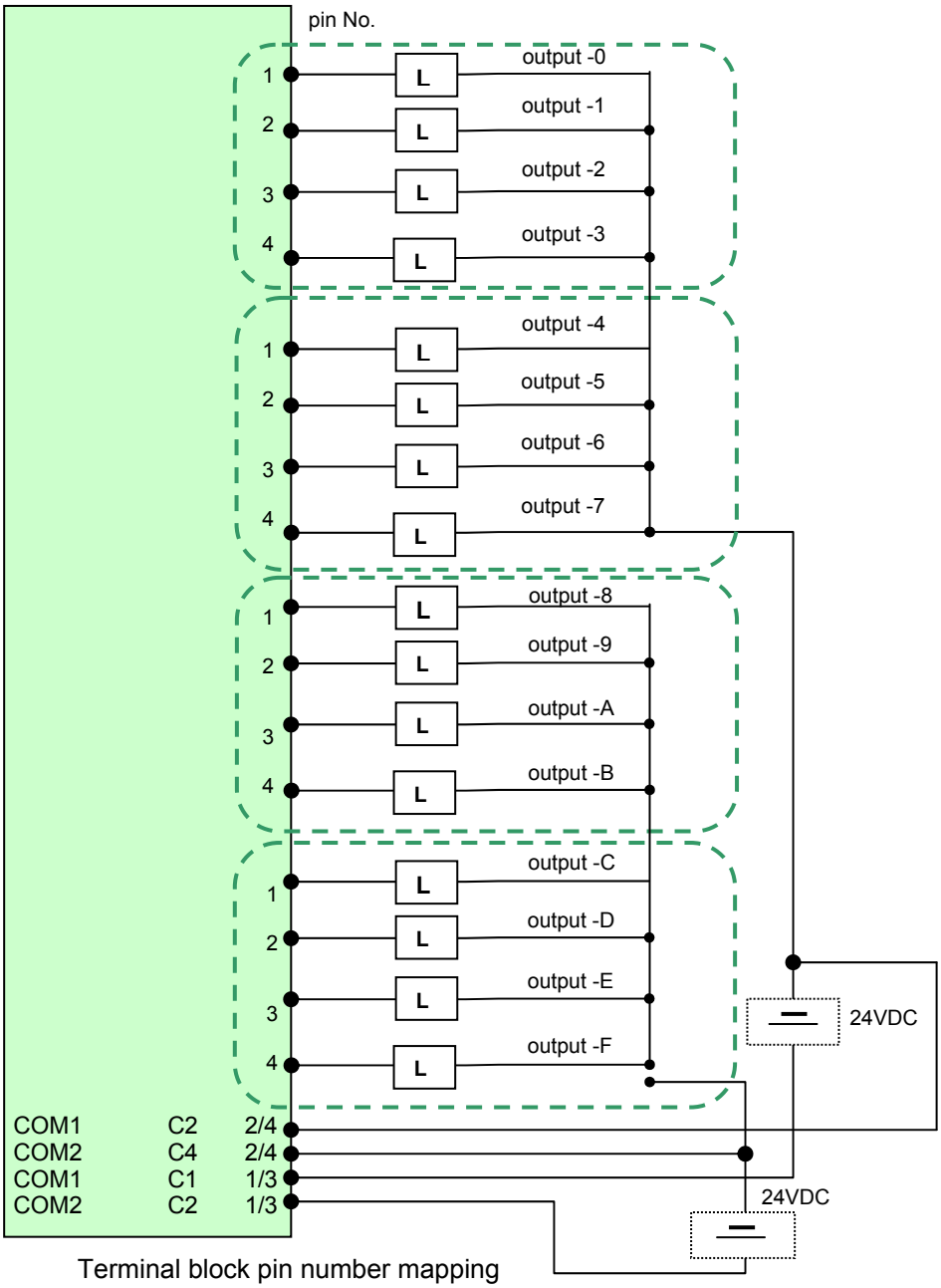
Figure 3-45 **SFL-NS-OUT+**

	I/O address	Pin no.	Print name	Terminal name	I/O address	Pin no.	Print name	Terminal name
	00	1	0	OUTPUT -0	01	2	1	OUTPUT -1
	02	3	2	OUTPUT -2	03	4	3	OUTPUT -3
	04	1	4	OUTPUT -4	05	2	5	OUTPUT -5
	06	3	6	OUTPUT -6	07	4	7	OUTPUT -7
	08	1	8	OUTPUT -8	09	2	9	OUTPUT -9
	0A	3	A	OUTPUT -A	0B	4	B	OUTPUT -B
	0C	1	C	OUTPUT -C	0D	2	D	OUTPUT -C
	0E	3	E	OUTPUT -E	0F	4	F	OUTPUT -F
PWR 1		1	O+	OUT_P1+		2	O-	OUT_P1-
		3	O+	OUT_P1+		4	O-	OUT_P1-
PWR 2		1	O+	OUT_P2+		2	O-	OUT_P2-
		3	O+	OUT_P2+		4	O-	OUT_P2-

Table 3-39 Terminal designation **SFL-NS-OUT+**

The pins no. 1 and 3 or 2 and 4 of the O+ or O- connections are internally bridged in order to facilitate the connection of the power supply to the neighbouring module.

SFL-NS-OUT+ terminal diagram



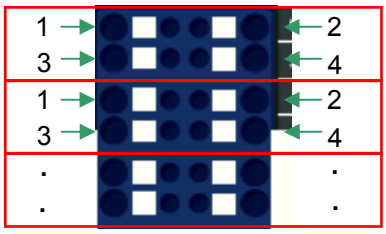


Figure 3-46 **SFL-NS-OUT+** terminal diagram

4 Installation / project planning

This chapter contains the description of the assembly and wiring of the **SFL** and circuitry examples of the most important basic circuits of safety protective devices.

4.1 Assembly

4.1.1 Ambient conditions

9.1 / 9-2

The following ambient conditions must be observed for the perfect operation of the **SFL**. Complete information is contained in Chapter 9.1.

Position	Description
Temperature range	0 to 55°C
Air humidity	30 to 85 % RH
Air pressure	86kPa bis 106kPa
Degree of pollution	2 to DIN EN 50178 (VDE 0160)
Place of installation	Earthed metal switch cabinet which may be closed with degree of protection IP 54 minimum
Installation condition	The influence should be within the limits expected in average environment. In installation, it is protected from strong (EX: installation place) influences (EX: vibration of a control cabinet etc.).

Table 4-1 Admissible ambient conditions

4.1.2 Assembly of the back plane

Installation position

In order to guarantee sufficient ventilation and comfortable assembly/dismantling of the individual modules, please leave a free space of at least 50mm above and beneath the station. A distance of 50mm must similarly be maintained to the left, right and to the front. The unit must be installed exclusively as shown below (horizontally suspended).

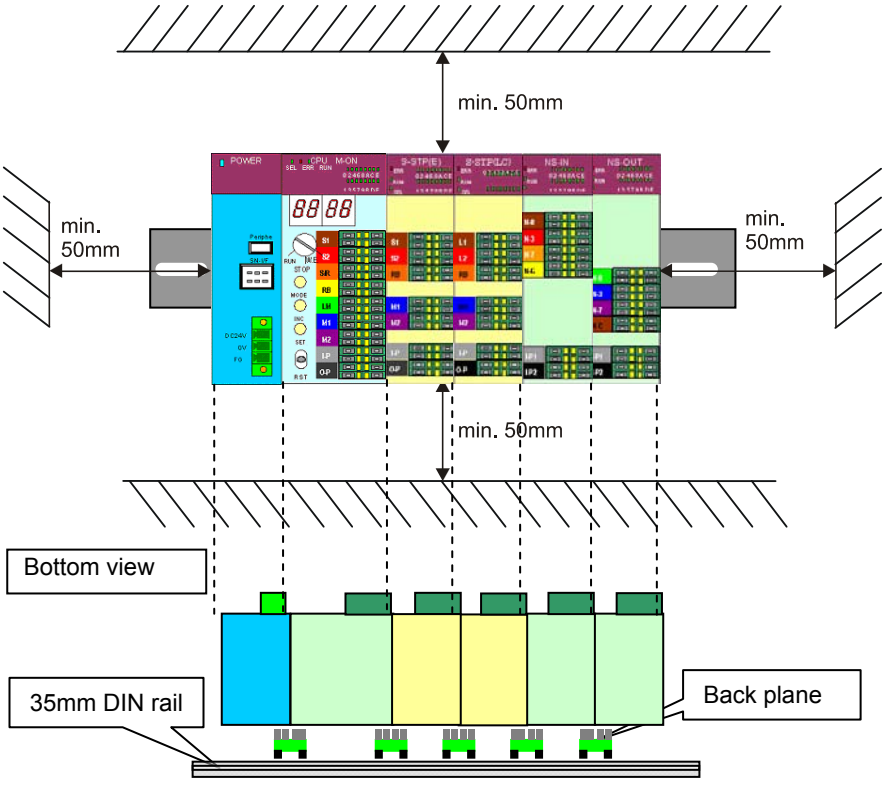


Figure 4-1 Installed position and position of the **SFL**

NOTE: In order to avoid interference with a base, the screws which fix a DIN rail should use low head screws, such as truss head screws.

Assembly

Select a location which is as far as possible away from the power level and other disturbing radiation and heat sources of the switch cabinet.

4.1.3 Assembly/dismantling of the individual modules

Assembly / dismantling work may only be performed in a de-energized state.

Assembly

1. Attach an earthed DIN top hat rail with a maximum distance of the screwed connection of 10cm. In order to attach the module to the top hat rail, the two interlocks must be tightened.

   The top hat rail should use TH35-7.5Al or TH35-15Fe (width: 35mm, height: 7.5mm) in accordance with IEC 60715 and fix it to control cabinet by sufficient strength to support the weight of the **SFL**.

   Consider that there is no influence by the vibration to the **SFL**. When influence remains, please use a sufficient number of screws, and fix the top hat rail.

NOTE: In order to avoid interference with **SFL-BASE**, the screws, which fix the top hat rail, should use low head screws, such as truss head screws.

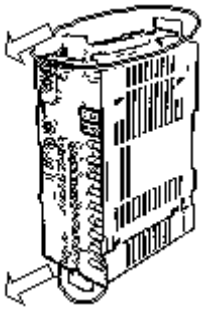


Figure 4-2 Assembly of a module (1/3)

2. Then carefully press the module against the top hat rail until a click is heard. If a further module is to be connected next to it, it must be ensured that the **SFL-BASE** is first installed.

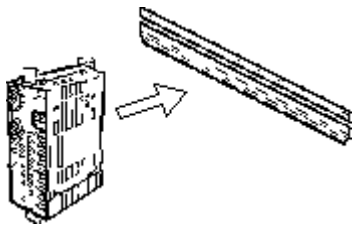


Figure 4-3 Assembly of a module (2/3)

3. Check that the module is fitted correctly and push the interlocks back where necessary.

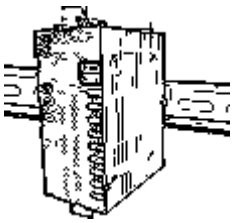


Figure 4-4 Assembly of a module (3/3)

4. After assembling all modules the system must be secured on both sides using fixing clips or comparable measures.

Demontage

Pull the two interlocks before removing the module.

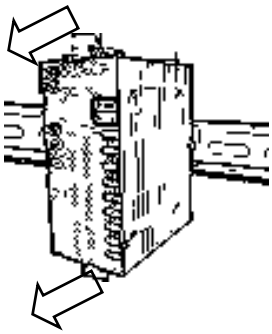


Figure 4-5 Dismantling a module

4.2 Wiring

4.2.1 General information on wiring

This chapter contains important information on the wiring of the **SFL**, which must be followed for safe and trouble-free operation.

The electrical equipment must be wired in conpliance with IEC 60439-1, 7.8.3.

The maximum admissible length of free hanging cable is 30cm.

Power supply

3.5.2 / 3-9 

The power supply units used for power supply must satisfy the requirements specified in Chapter 3.5.2 / 3-9. The power supply for the POWER / BOOSTER / CPU module and the safe/operational modules can be provided from the power supply unit of the module or separate power supply units. Chapter 4.2.4 / 4-10 contains a calculation example of the value of the fuse F2.

4.2.4 / 4-10 

Common supply

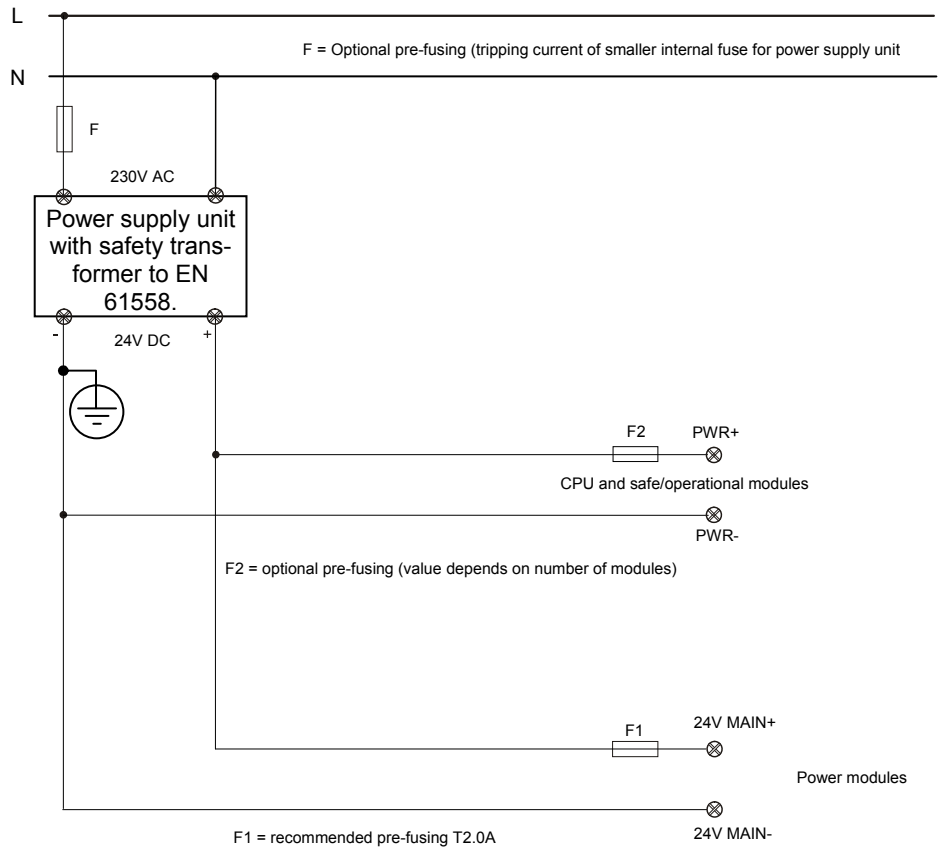


Figure 4-6 Power supply of the **SFL** common power supply unit with safety transformer

Separate supply

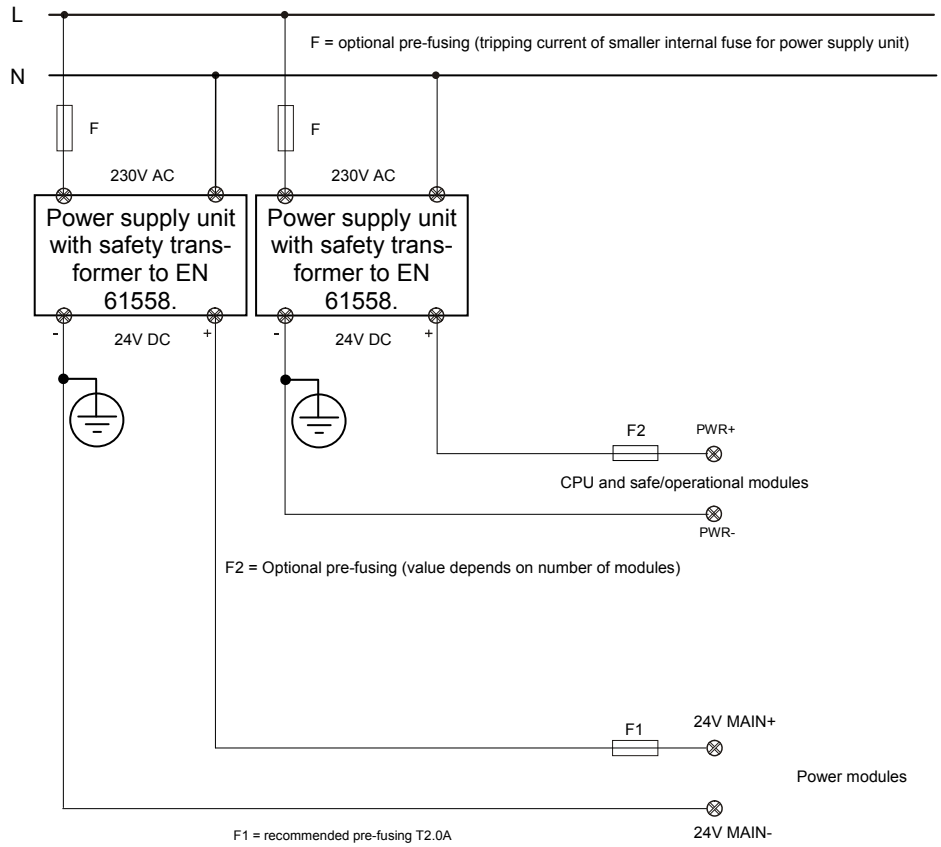


Figure 4-7 Power supply of the **SFL**, separate power supply units with safety transformer

The maximum cable lengths between the power supply unit and the **SFL** may not exceed 15m.

When two or more power supplies are used, the power supplies which connected to the CPU module, safety / un-safety I/O module(s) and the Booster module should be previously or simultaneously switch on than the power supply which connected to the Power module. If the power supply, which connected to the Power module, switch on earlier than the other's, an error may occur by a module monitoring function. The **SFL** moves to the safe state.

Leave the safe state is only possible by Power OFF/ ON or be activating the R.ST Button

Laying the Input/Output cables

The input and output cables must be at least 100mm away from high voltage / high current cables. In order to rule out cross-shorts, the cables must be laid in accordance with one of the following criteria:

1. permanently laid cables and protection against external damage
2. laying in different sheathed cables
3. laying within an electrical installation room and cables according to the requirements of DIN EN 60204-1.
4. cables whose shielding is individually earthed

Power supply cables

Ensure that power supply cables are at least 50mm away from the modules.

Connector strips

Connector strip			Recommended cable cross-section
Power	2 polig	PHOENIX FKC2.5/2-STF-5.08	0.75mm² for all connecting cables (apart from earth) In the case of Morimatsu connector strips: Remove 10mm of insulation sheath from the cable and use 10mm long wire-end ferrules
Sub-Master-On	36 polig	MORIMATSU M820A-09-xx	
Safety-Stop	28 polig	MORIMATSU M820A-09-xx	
Safety-Input	36 polig	MORIMATSU M820A-09-xx	
Safety-Output	36 polig	MORIMATSU M820A-09-xx	
Relay Output	8 polig	PHOENIX FKC2.5/8-GF-5.08	
Non-Safety Input	24 polig	MORIMATSU M820A-06-xx	
Non-Safety Output	24 polig	MORIMATSU M820A-06-xx	
CPU	36 polig	MORIMATSU M820A-09-xx	

Table 4-2 Connector strips used for the module

For reasons of shock hazard protection, all connectors must be fitted with the appropriate mating plugs.

Wire-end ferrules

All cables used must have wire-end ferrules (max. 0.75mm²; length 10mm, rectangular cross-section). An exception is the relay module who connector strips are designed to accommodate wire-end ferrules of up to 1.5mm². Ring cable lugs (M4) are to be used for the earth cables.
Recommended wire-end ferrules are below.

Manufacturer	Wire size		Type	No.
	mm^2	AWG		
Phoenix contact	0.75	18	AI0.75-10GY	3201288
	1		AI1-10RD	3200182

Cable ducts

Lay the input and output cables inside and outside the switch cabinet in separate cable ducts or similar. If cable ducts or pipes made of metal are used, they must be earthed.

Shielded cables

If input cables and output cables need to be laid together with power cables, use shielded cables and earth the shield.

Internal fusing

The CPU and the safe/operational modules, with the exception of the relay module, have internal fusing which is intended to protect the modules from destruction in the case of a short circuit. These fuses are not conceived as overload protection for normal operation. Please observe the prescribed specification for the connection of the modules. If the internal fuse is destroyed, the module no longer functions. The fuse can be replaced.

4.2.2 Power module

An external pre-fuse of 1.0A slow blowing is to be incorporated into the supply tables.

4.2.3 Assembly of the back-up battery

The CPU module is not supplied with a back-up battery. It must be assembled by the technician before commissioning the module.

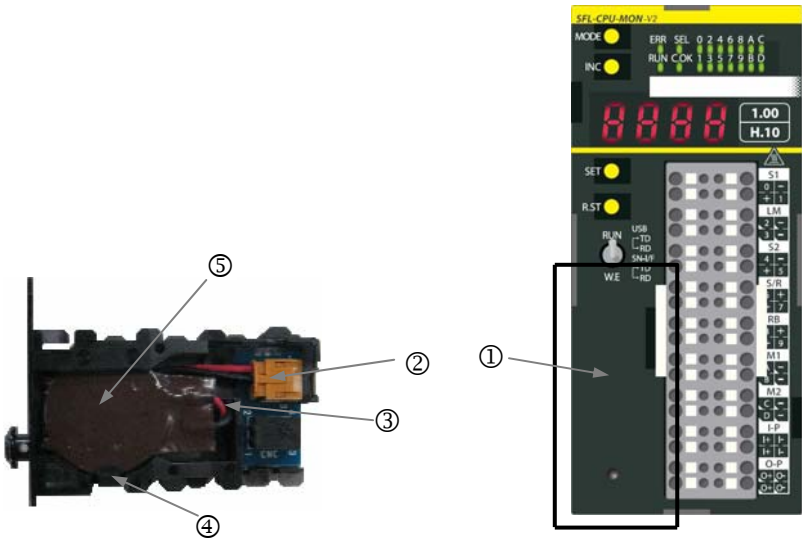


Figure 4-8 Assembling the back-up battery

1. Switch off the power supply.
2. Remove the battery holder ①.
3. Pull out the plug ②.
4. Remove the cable from the battery holder ③.
5. Press the holding lug ④down.
6. Remove the old battery ⑤
7. Take a new battery ⑤
8. Press the holding lug ④ to the right.
9. Insert the new battery ⑤ with the cabling to the top.
10. Guide the cable into the battery holder ③.
11. Insert plug ②.
12. Assemble the battery holder ①.
13. Mark the enclosed sticker ⑦ with the current date and attach it to the CPU module.
14. Switch on the power again.

4.2.4 Input/Output modules

The CPU module and the safe/operational modules require an external 24V DC supply voltage for the functionalities of the Input/Output level. This supply is to be secured by means of an external fuse. The tripping current for the fuse will depend on the number and type of modules supplied.

Example to calculate the tripping current

Assuming the **SFL** has the following modules:

2 x input module **SFL-S-IN-E** => 2 x 16 inputs 5mA

1 x output module **SFL-RELAY** => 4 outputs 3000mA

1 x output module **SFL-S-OUT** => 16 outputs 250mA

this will produce a maximum overall current of:

160mA + 12000mA + 4000mA = 16160mA=> fuse = 20A quick acting

When selecting the pre-fusing, please bear in mind that it must respond before the fuse of the power supply unit.

In order to prevent the internal fuse (output modules only) responding, an individual fusing of the modules is to be taken into consideration if the work is acceptable.

When connecting sensors/actuators, it must be taken into consideration that neighbouring inputs/outputs carry different potential.

When wiring the sensor/actuator supply voltage, observe the different connection assignments of the individual modules. Terminals that are not used may need to be interlinked under some circumstances.

Take suitable protective measures (free-running diode or similar) when switching inductive loads to protect the semi-conductor outputs from excessive voltage.

4.3 Circuitry examples

4.3.1 Information on circuitry examples

The safety categories specified for the circuitry examples only apply to the entirety of circuitry. This includes external circuitry, the properties of the external sensory system/actuator system and a correct user program in terms of safety aspects. Insofar as he does not have recourse to the program examples to which the circuitry examples make reference, the programmer must ensure that the user program he creates satisfies all requisite measures to achieve the foreseen safety categorisation.

RESET

7.11.1 / 7-52

A RESET is the manual resetting into the ready to operate state after a safety device has been triggered. The RESET must be monitored or controlled by a device categorised as safe.

In the event of EMERGENCY-OFF / EMERGENCY-STOP, the reset can be effected by the switch's mechanical latching function in accordance with IEC 60947-5-5. In this case the resetting is caused by the shut-down system itself.

START

7.11.2 / 7-53

A START command starts or restarts the operation of a machine or system monitored by the protective device.

4.3.2 EMERGENCY-OFF (EMERGENCY-STOP) circuitry

Start/reset level ①

Start button with integrated feedback circuit. The feedback circuit facilitates the activation of the circuit only if both actuators signal idle status via their auxiliary contacts (NC contacts). The reset is effected by means of the mechanical latching function of the EMERGENCY OFF (EMERGENCY STOP) switch.

Sensor level ②

2 channel EMERGENCY OFF/EMERGENCY STOP circuit to EN 418/EN 60947-5-5 with cross-short detection. The cross-fault detection function is only guaranteed if the sensor's channels switch against different potential, as shown below.

Actuator level ③

2 channel power level (series circuitry of the actuator contacts). A relay or contactor with positively driven contacts must be used.

Safety categorisation

Maximum advisable category is category 3 to DIN EN 954-1 or PL e to DIN EN ISO 13849-1 (in the case of series connection of the sensors, observe special features).

Remarks

Start-up occurs only after the release of the start button (monitored start) with the negative edge.

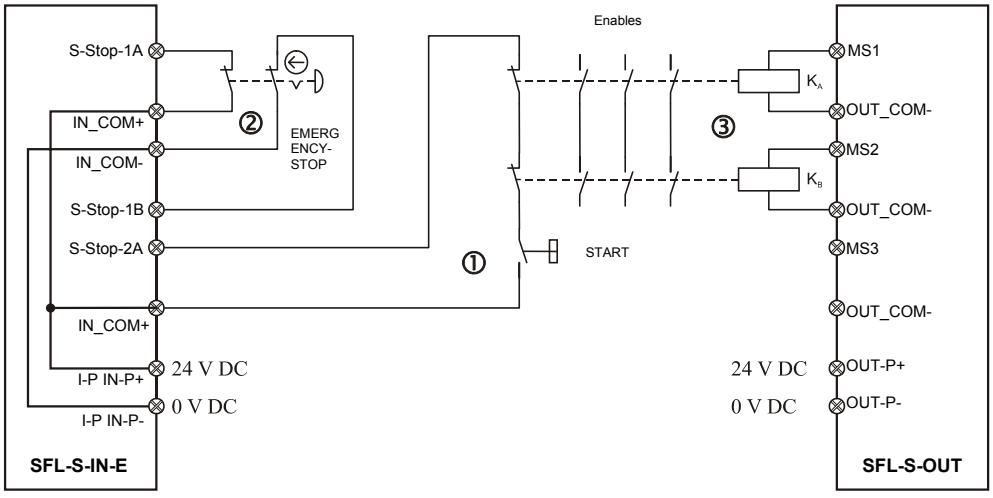


Figure 4-9 Circuitry example of an EMERGENCY-OFF (EMERGENCY-STOP) circuit

4.3.3 Guard monitoring

Start / reset level ①

Start button and reset of the safety functions. The feedback circuit facilitates the start of the circuit only if both actuators signal release state (de-energised state) via their auxiliary contacts (NC contacts).

Sensor level ②

2 channel guard monitoring to EN 1088 with at least one positively opening position switch with cross-fault detection. The cross-fault detection function is only guaranteed if the sensor's channels switch against different potential as shown below.

Actuator level ③

2 channel power level (series circuitry of the actuator contacts). A relay or contactor with positively driven contacts must be used.

Safety categorisation

Maximum advisable category is category 4 to DIN EN 954-1 or PLe to DIN EN ISO 13849-1 (in the case of series connection of the sensors, observe special features).

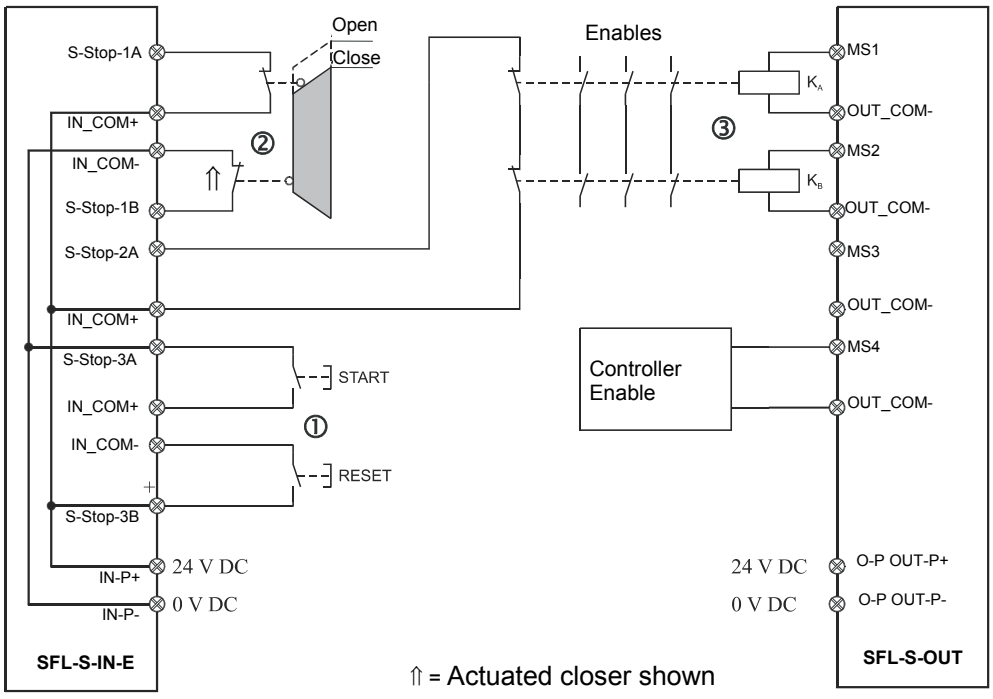


Figure 4-10 Circuitry example of guard monitoring

4.3.4 Guard monitoring with locking

Start / reset level ①

Automatic start and no reset of the safety function with the integrated feedback circuit. The feedback circuit facilitates the start of the circuit only if both actors signal the release state (de-energised state) via their auxiliary contacts (NC contacts).

Sensor level ②

2 channel guard monitoring to EN 1088 with solenoid actuated locking with cross-fault detection. The cross-fault detection function is only guaranteed if the sensor's channels switch against different potential, as shown below.

Actuator level ③

2 channel power level (series circuitry of the actuator contacts). A relay or contactor with positively driven contacts must be used.

Safety categorisation

The exact safety categorisation will depend on the application.

Remarks

This circuitry example is to be used exclusively for machine protection. Use for personal protection is only admissible with suitable additional measures due to a lack of locking in the de-energised state.

7.11.5 / 7-58 The "automatic start" function is realised here in connection with the feedback circuit in the user program (see Chapter 7.11.5).

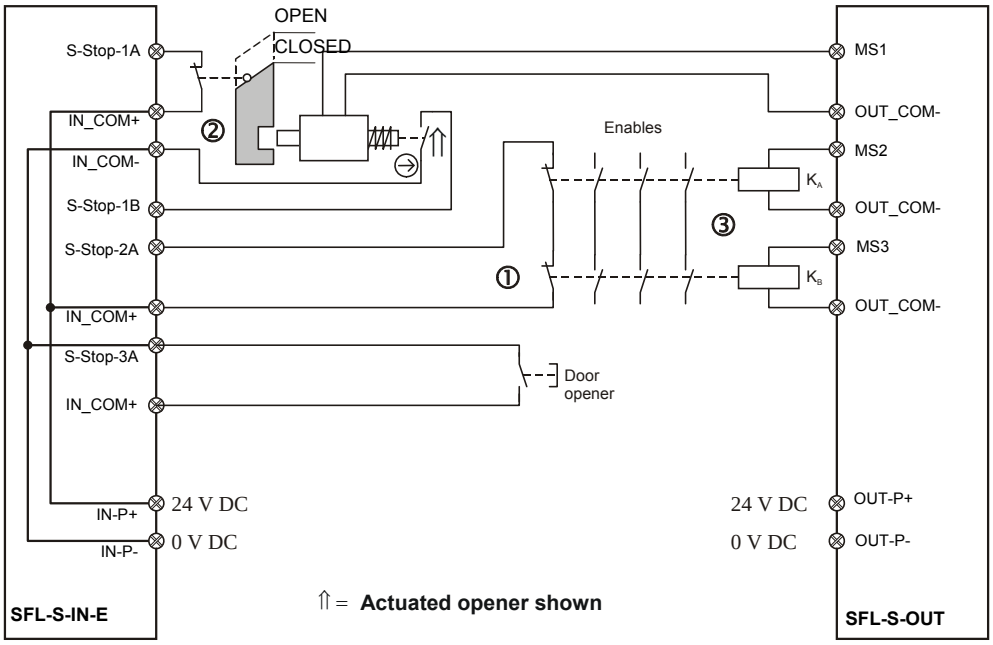


Figure 4-11 Circuitry example of guard monitoring with locking

4.3.5 Safety solenoid-operated switch

Start / reset level ①

Start button and reset of the safety function. The feedback circuit facilitates the start of the circuit only if both actuators signal release state (de-energised state) via their auxiliary contacts (NC contacts).

Sensor level ②

2 channel control with safety solenoid-operated switches to DIN VDE 0660-209 with cross-fault detection. The cross-fault detection function is only guaranteed if the channels of the sensor switch against the different potential, as shown below.

Actuator level ③

2 channel power level (series circuitry of the actuator contacts). A relay or contactor with positively driven contacts must be used.

Safety categorisation

Maximum possible category is category 4 to DIN EN 954-1 or PL e to DIN EN ISO 13849-1 (max. category 3 for series connection of the sensors).

Remarks

Please observe the C standard with respect to the admissibility of an individual switch or consult the manufacturer!

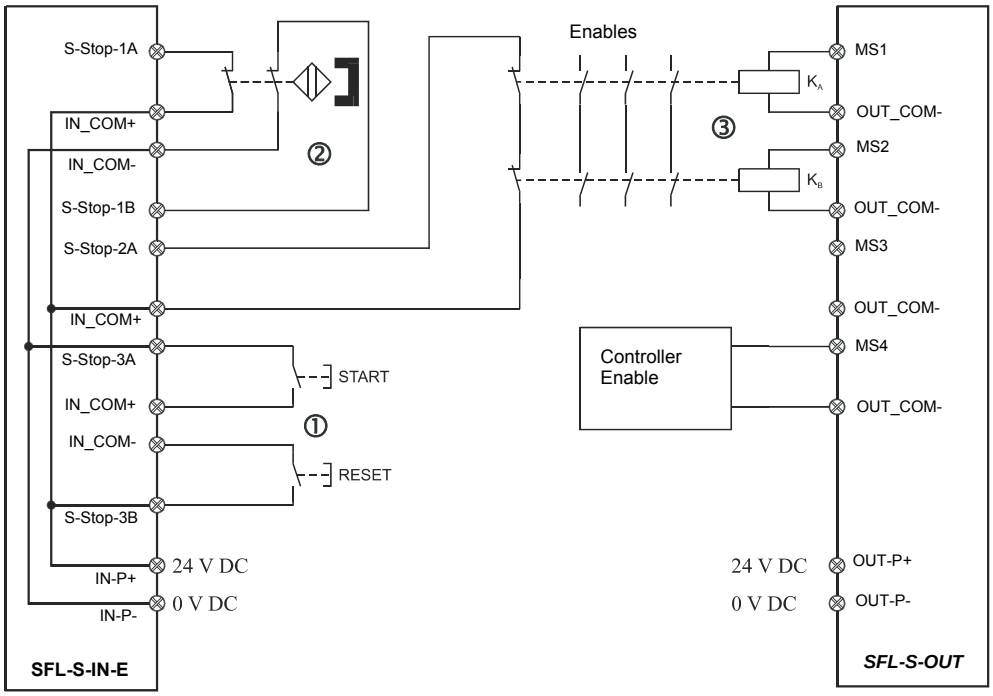


Figure 4-12 Circuitry example of safety solenoid-operated switch

4.3.6 Current-sourcing semiconductor

Start / reset level ①

The start is managed by the operational control, whereby the **SFL** controller receives the information on the status of the contactor via one of the enable contacts. There is no manual resetting of the safety function. The feedback circuit facilitates the start of the circuit only if both actuators signal the release state (de-energised state) via their auxiliary contacts (NC contacts).

Sensor level ②

2 channel control with safety-oriented current-sourcing semiconductor components, e.g. AOPDs to EN 61496. The cross-short detection must be performed by the AOPD. Each sensor with semiconductor output must have a 2 channel connection.

Actuator level ③

2 channel power level (series circuitry of the actuator contacts). A relay or contactor with positively driven contacts must be used.

Safety categorisation

Maximum realisable category is category 4 to DIN EN 954-1 or PLe to DIN EN ISO 13849-1 (max. category 3 for series connection of the sensors). The exact safety categorisation will depend on the sensor used. For categorisation in accordance with category 4 to DIN EN 954-1 or PLe to DIN EN ISO 13849-1 the sensor must have its own watchdog.

Remarks

The following applies to dangerous points/dangerous areas: the circuitry without reset is not admissible without additional measures if there is a risk of accessing the zone. It must be ensured that a restart of the plant is only possible if noone is in the hazard area.

4.3.3 / 4-13 

If, for structural reasons, the hazard situation requires a reset button, the start / reset circuit (①) and the corresponding user program is to be used analogously to the example 4.3.3 Guard monitoring.

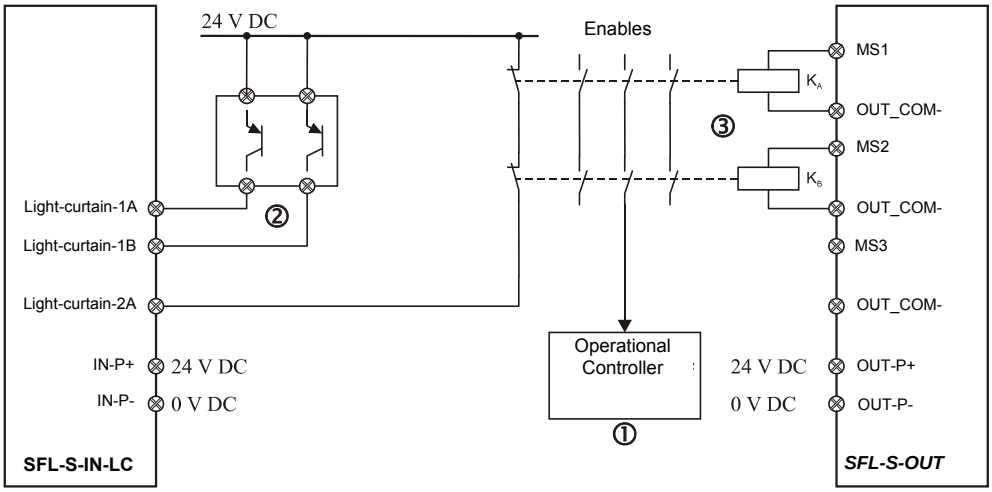


Figure 4-13 Circuitry example of current-sourcing semiconductor

4.3.7 Circuitry of the actuator level

Semiconductor outputs

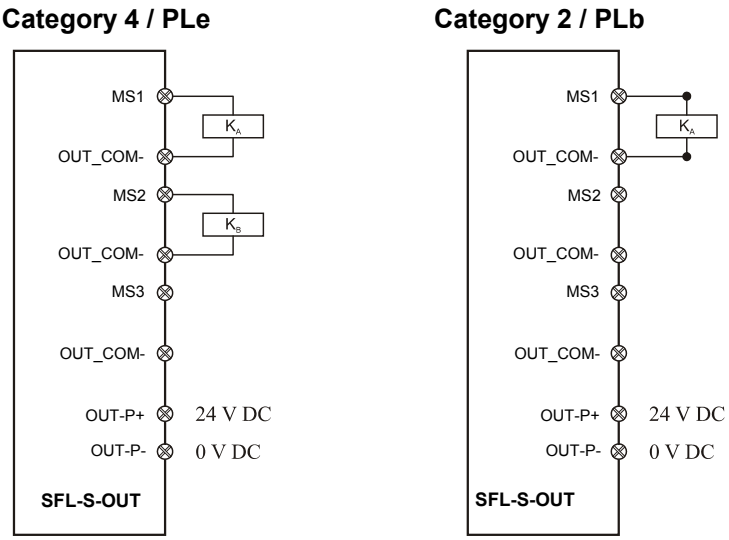


Figure 4-14 Circuit types for actuators

The second shut-down path is not shown in the example for category 2 / PLb.

Relay exit

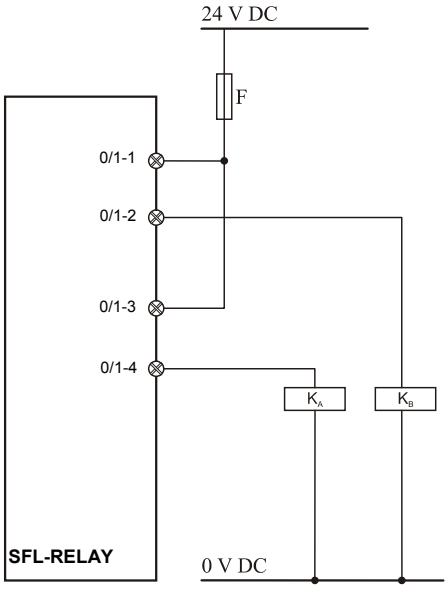


Figure 4-15 Types of circuit for actuators

If the actuator does not have its own fuse, a fuse must be incorporated in the supply line for the actuators.

4.3.8 Checking

Assembly

Is there a free space of at least 50mm above and beneath the modules?

Are all modules correctly locked in and screwed?

Do all modules have a minimum distance of 50mm to power lines?

Power supply

Does the power supply comply with the requisite requirements (refer to Chapter 3)?

Is an external fuse incorporated?

Laying of cables

Do the Input/Output cables have a minimum distance of 100mm to the power lines?

Has a shielded cable been used if a minimum distance of 100mm could not be observed?

Wiring

Do all connection lines have wire-end ferrules?

Are all connections correctly wired and poled?

Do all Input/Output modules have a 24V DC supply?

5 Operating mode 1

This chapter describes operation of the **SFL** in operating mode 1.

5.1 Overview

Implementation in operating mode 1 requires the setting of a PIN code and confirmation of the configuration.

3.4 / 3-6 The PIN code is set in the CPU module. Detailed information can be found in Chapter 3.4.

The channel difference time (see Chapter 7.9.4) of all 2 channelled inputs is 1.0s.

In modules where monitoring of the feedback circuits takes place, e.g. **SFL-CPU-MON-V2**, **SFL-S-STP-LC**, the maximum delay between the triggering of the actuator and the expected change of the feedback circuit may be 0.5s.

5.2 Project planning

In order to facilitate implementation in operating mode 1, the following sequence must be adhered to.

4.1 / 4-2 **1.** Configuration and assembly
4.2 / 4-5 **2.** Wiring
5.2.1 / 5-2 **3.** Setting/changing the PIN code
5.2.2 / 5-5 **4.** Confirmation of the configuration

Refer to the respective chapters for details.

5.2.1 Setting/changing the PIN code

The PIN code is set/changed using the MODE / INC buttons and the SET button.

3.4.2 / 3-8 The PIN code in Mode 0 (see Chapter 3.4.2) is “0000”. The 4 digit decimal numbers “0001” to “9999” may be selected as PIN code.

2.1.7 / 2-4 See Chapter 2.1.7 for further information on the PIN code.

Setting the PIN code

The setting of a new PIN code must be completed before a new configuration can be confirmed.

For this the RUN/W.E. switch on the CPU module must be set to “W.E.” and the PIN code entered.

e.g. PIN code = 1234

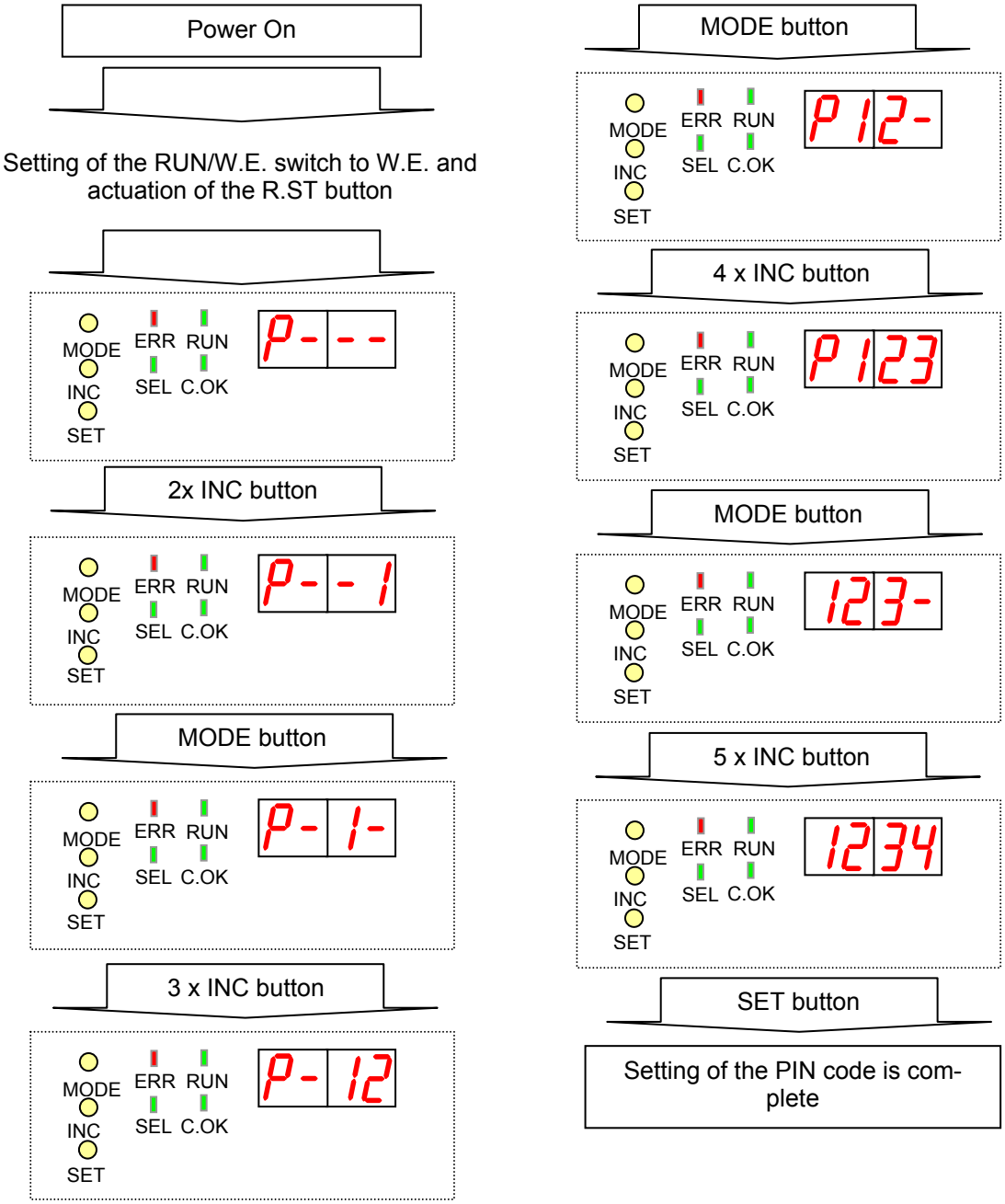


Figure 5-1 Setting the PIN code

Changing the PIN code

The current PIN must first be entered in order to change the PIN code. If the current PIN code is entered incorrectly, it must be entered again.

Example: **Old** PIN code 1234,
New PIN code 5678

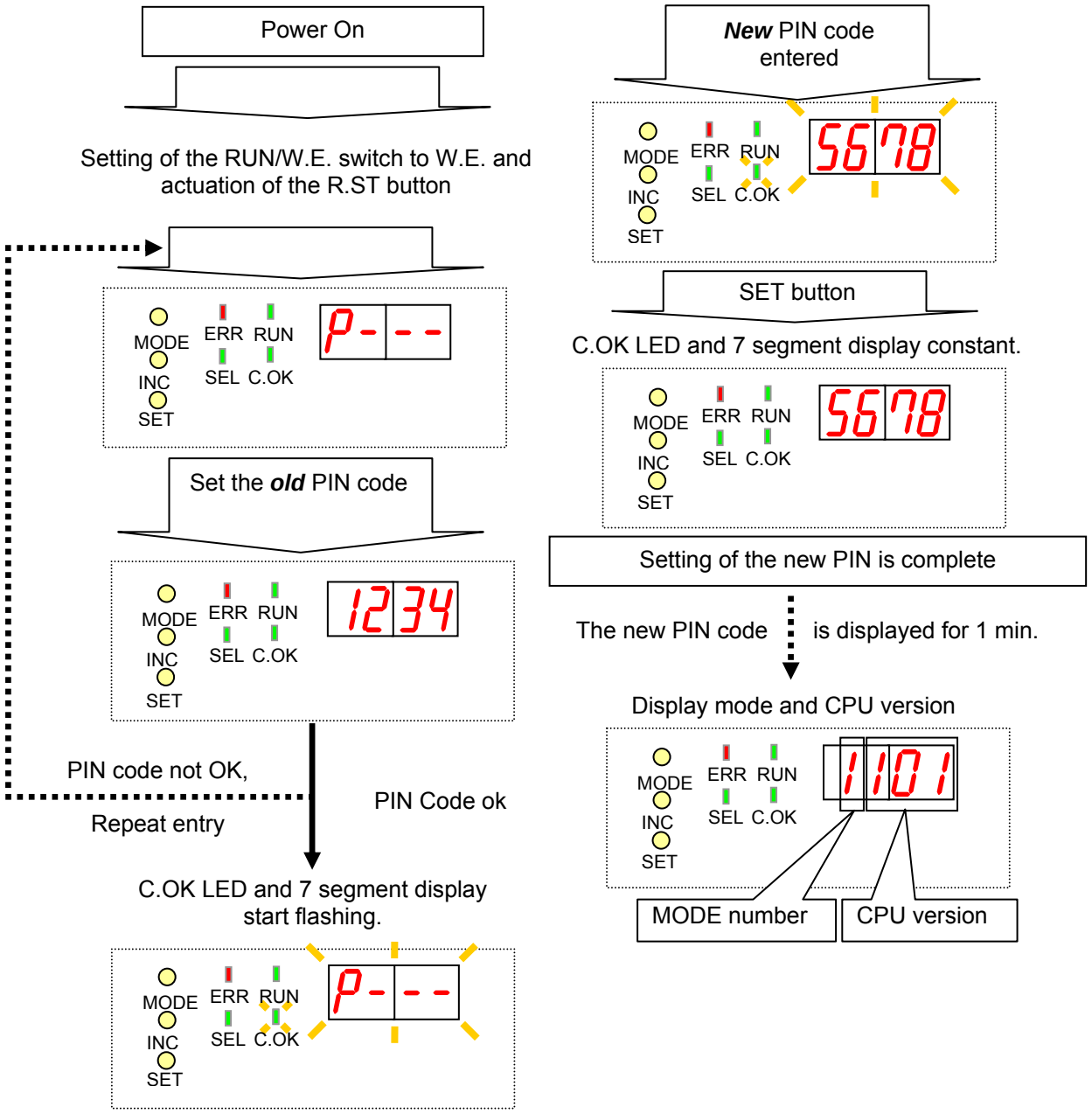


Figure 5-2 Changing the PIN code

5.2.2 Confirmation of the configuration

The configuration must be confirmed in order to work in Mode 1. During confirmation of the configuration the number of the slot and the ID code of the installed module are displayed on the 7 segment display.

Conformity between the assembled module and the displayed slot /ID code must be individually confirmed during confirmation.

3.3 / 3-6 Details on the ID code of the modules can be found in Chapter 3.3.

Delivery status

3.4.2 / 3-8 If a CPU module is to be moved from Mode 0 (see Chapter 3.4.2 to Mode 1, the PIN code must first be set, and only then is it possible to confirm the configuration.

The precise sequence is described in Figure 5-4.

5.2.1 / 5-2 Details on setting the PIN code can be found in Chapter 5.2.1.

Example

Configuration: CPU(MON) – S-STP(E) – RELAY – S-OUT

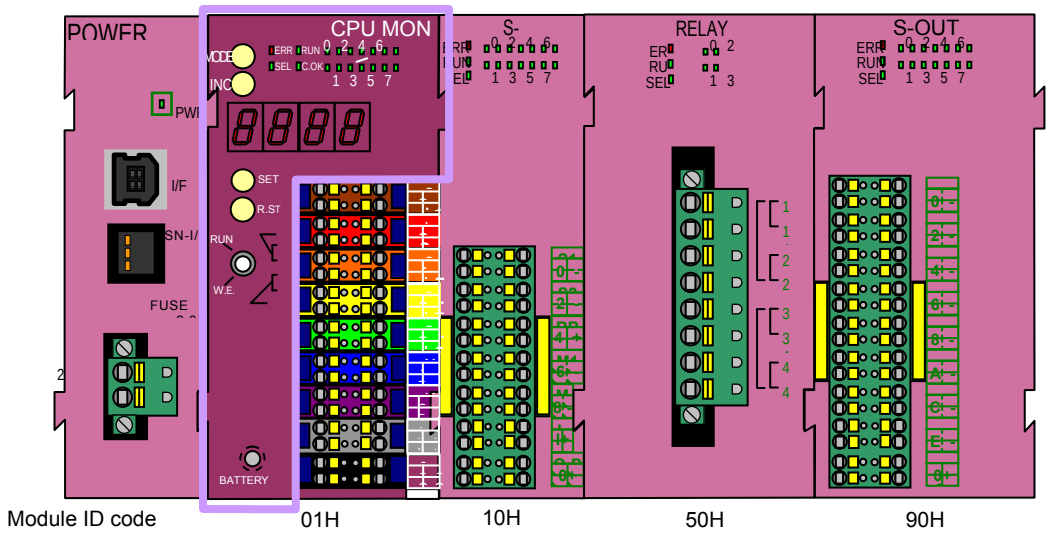


Figure 5-3 Module configuration (delivery status → Mode 1)

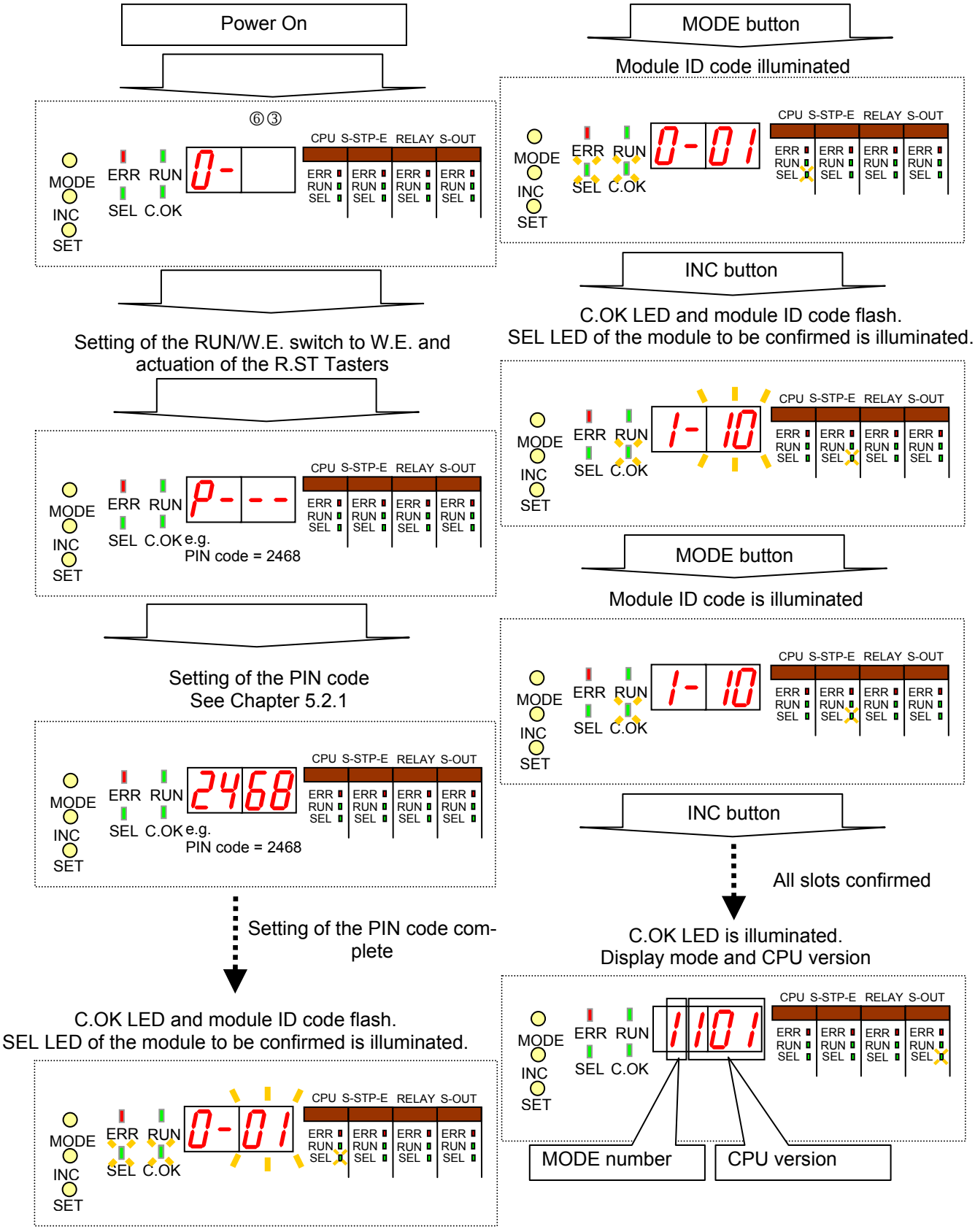


Figure 5-4 Confirmation of the configuration (delivery status → Mode 1)

5.2.3 Changing the configuration

The valid PIN code must be entered before the changed configuration is confirmed. The current configuration is then deleted and the new configuration can be confirmed.

An example is shown in Figure 5-7.

5.2.1 / 5-2  More precise information for the confirmation of the configuration can be found in Chapter 5.2.1.

Old configuration: CPU(MON)– S-STP-E – RELAY – S-OUT

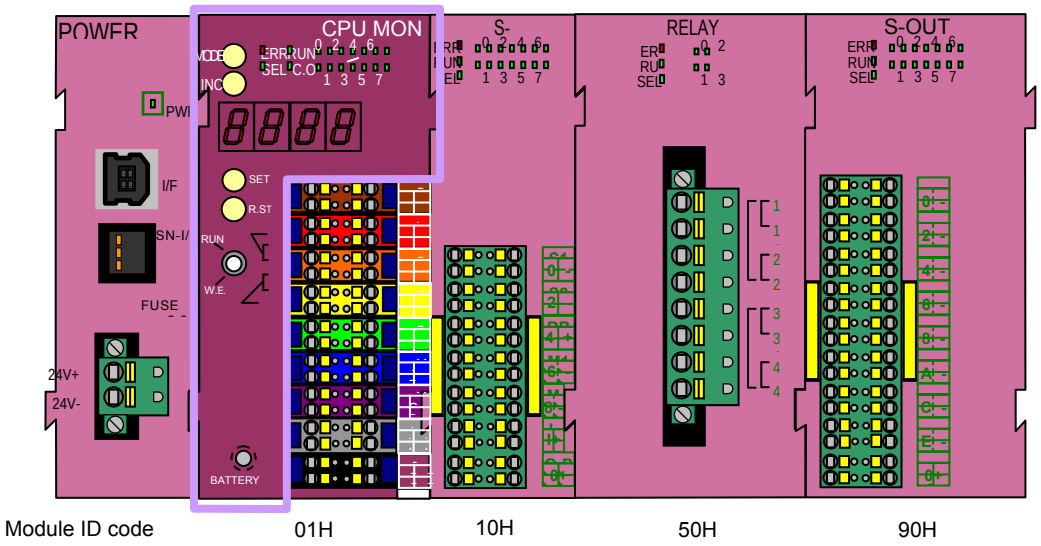


Figure 5-5 Old configuration

Changing the input module

New configuration: CPU – S-IN-E – RELAY – S-OUT

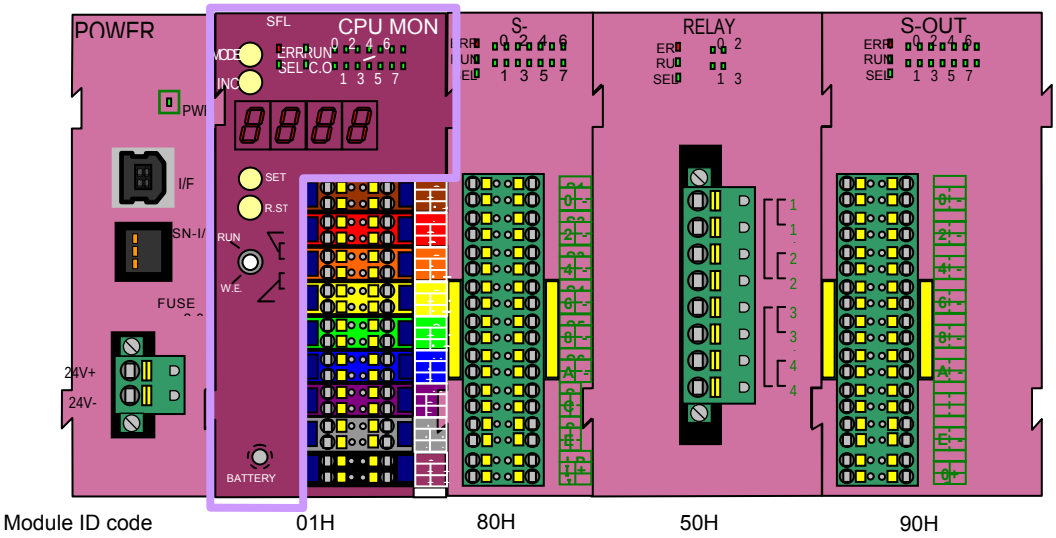


Figure 5-6 New configuration

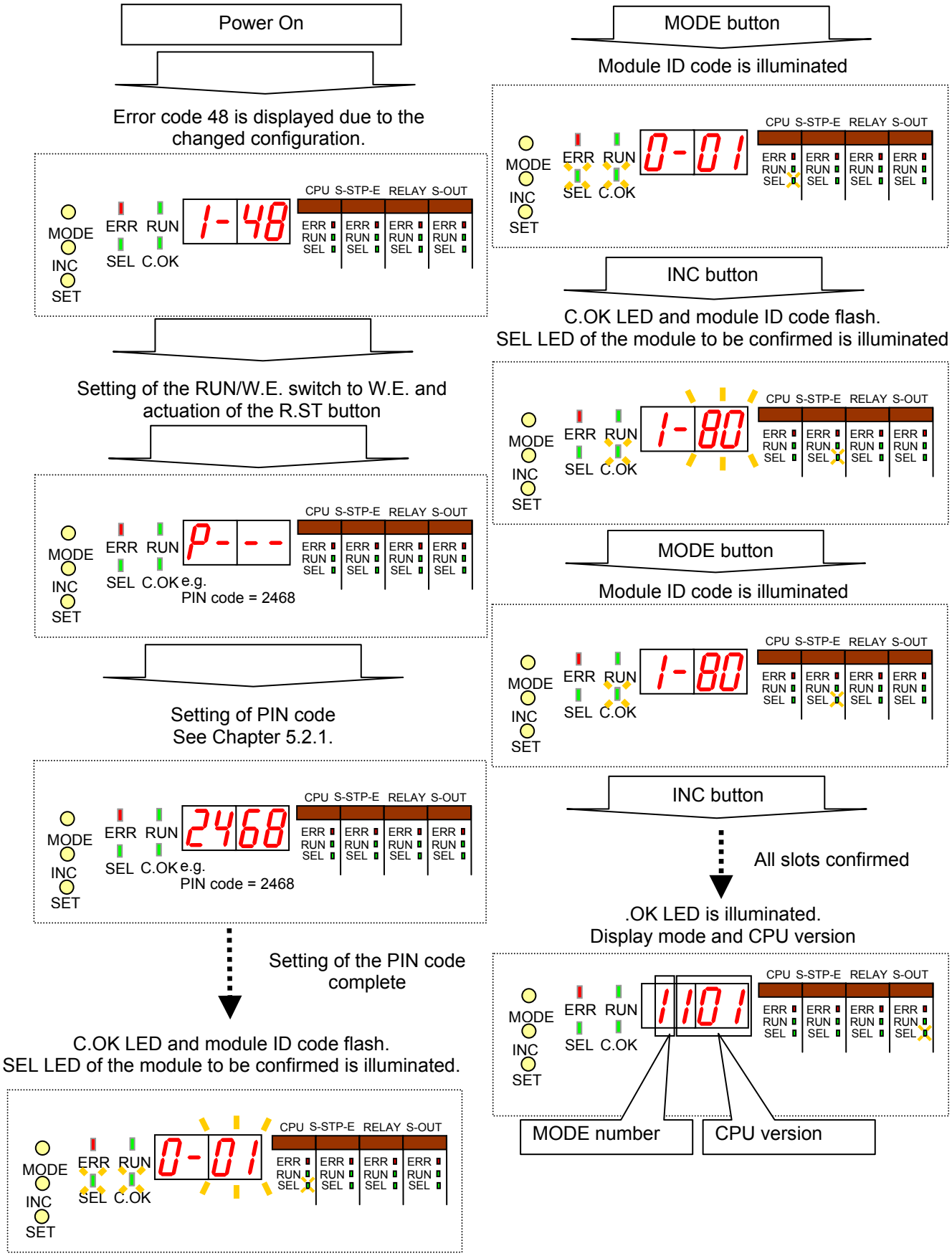


Figure 5-7 Confirmation of the configuration during configuration change

Transfer Mode 3 → Mode 1

In order to switch from Mode 3 to Mode 1, the CPU module must be returned to the delivery status (Mode 0).

The following options are available for attaining the delivery status:

5.4 / 5-9

- Switching to the delivery status (Mode 0) via the CPU module is described in Chapter 5.4.

7.18 / 7-105

- Switching to the delivery status (Mode 0) via **SFL-SOFT** is described in Chapter 7.18 .

5.3 Monitoring

7.15 / 7-95

In Mode 1 monitoring by **SFL-SOFT** is available. See Chapter 7.15. for more detailed information.

5.4 Initiation of Mode 0 (delivery status)

In order to place the CPU module in Mode 0 (delivery status), the PIN code “0000” must be entered.

5.2.1 / 5-2

See Chapter 5.2.1 for more information on setting the PIN code.

6 Operating mode 3

This chapter describes operation of the **SFL** in operating mode 3.

6.1 Overview

8.2 / 8-2

Implementation in Mode 3 requires a user program which must be created using **SFL-SOFT**. The creation and transfer of the user program are described in Chapter 8.2.

3.4 / 3-6

The PIN code is set using **SFL-SOFT**. Further information on Mode 3 can be found in Chapter 3.4.

6.2 Project planning

In order to facilitate implementation in operating mode 3, the following sequence must be adhered to.

7.10 / 7-31
7.13 / 7-82
7.7 / 7-11

1. Creation of the user program
2. Saving / logical checking of the user program
3. Transfer to the **SFL**. Confirmation of the PIN code is necessary.
4. Confirmation of the error-free transfer.
5. Checking of the program in the application.

Further details on **SFL-SOFT** can be found in Chapter 7.

The programming of the inputs/outputs of Mode 1 is adopted when switching from Mode 1 to Mode 3. This should be observed during further project planning. A programmer needs to mind about this, and needs to change and add.(1/2 channel setting of I/O setup are succeeded as it is) .

7 Programming / parameter assignment

7.1 Overview

SFL-SOFT will provide assistance in the creation of safe/operational programs for the **SFL**.

Using the **SFL-SOFT** programming software the programmer creates the user program in the form of a ladder diagram (to IEC 61131). For the purposes of verification the ladder diagram is converted to a statement list in the PS program. After checking the PS program by the programmer with the assistance of the programming software, the PS program is translated into a format which can be read by the CPU module and can then be transferred to the CPU module by means of a USB interface. The PN program is directly converted into a format which can be read by the CPU module and transmitted by means of the USB interface.

After successfully transfer to the CPU module, it sends the PS program to the programming software again for checking purposes which then releases the user software for the CPU module for operation after comparison with the transmitted program.

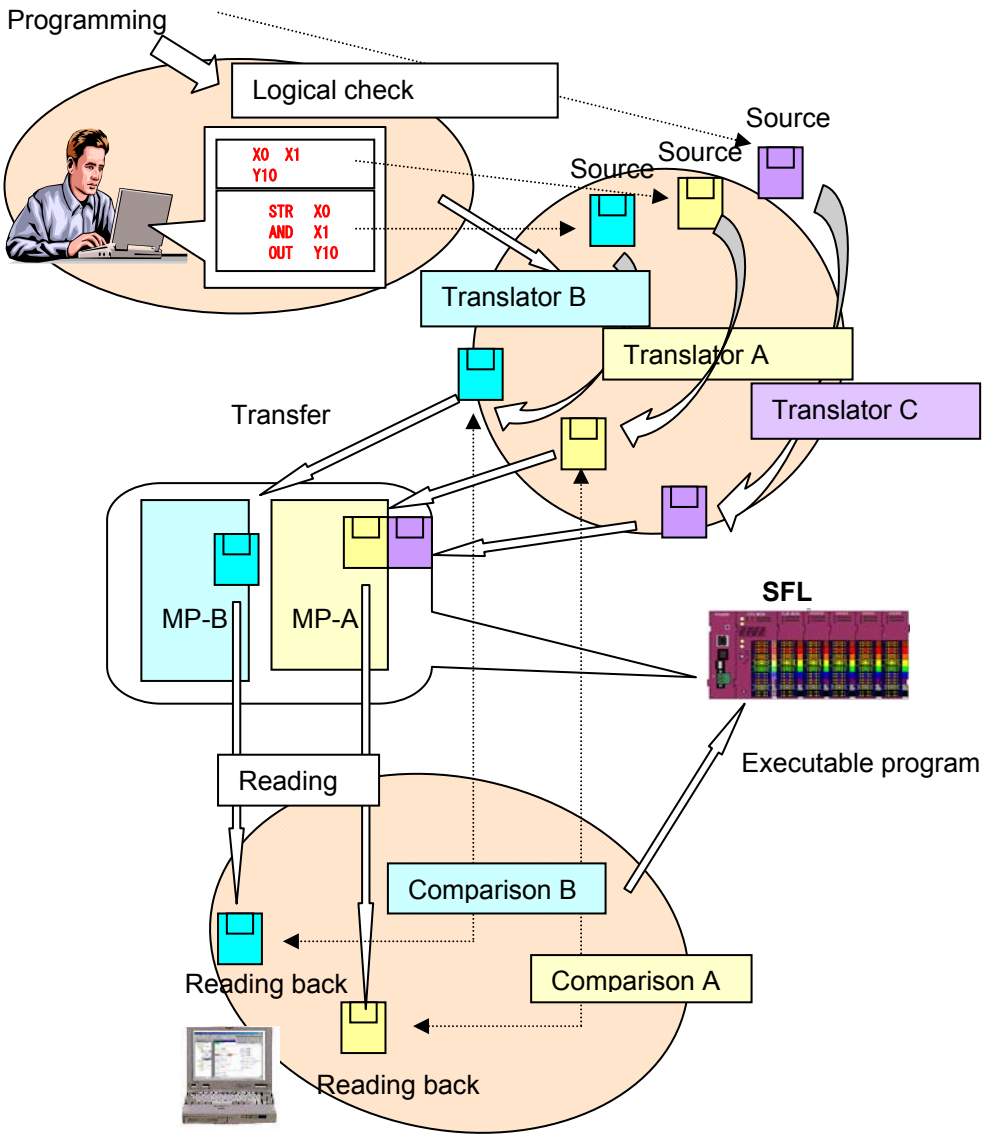


Figure 7-1 Creation of a PS program

7.2 Functionality

The functionality of **SFL-SOFT** is shown in the following table.

Functionality		SFL-SOFT	
		No password	Password
User program	Read	✓	✓
	Edit		✓
FB library	Read		✓
	Edit		✓

Table 7-1 Functionality of the **SFL-SOFT**

7.3 System requirements

The system requirements described below are minimum requirements for the operation of **SFL-SOFT**. The values in brackets are recommended values for a better performance to ensure a smooth work flow.

7.3.1 Operating system

Microsoft Windows 2000/XP/Vista/7 with the rights

„Administrator“

„Power User“

7.3.2 Hardware

CPU : Pentium 200MHz (preferably 500MHz or higher)

Memory : at least 128MB (preferably 256MB or higher)

Graphics : VGA Graphics with at least 800 x 600 pixels

Hard disk : at least 80MB free memory

7.4 Installation

Before starting installation please close all active programs and deactivate your anti-virus software (if present).

1. Insert the CD with the **SFL-SOFT** software.
2. Start the 'Setup.exe' file.
3. Follow the instructions of the installation routine.

After the first connection of the CPU module with the PC via the standard USB interface, the USB driver is first installed on the PC. The necessary driver ('ump') will usually be located in the directory …/windows/system32/drivers.

7.5 Password protection

The creation of new programs, the modification of existing projects and their transfer to **SFL** for existing projects is protected by a password query. This ensures that unauthorized persons are refused active access. After the first time the program is started, a window indicates that no password has yet been entered and that standard passwords are generated.

The entry of a password is necessary for the following processes:

1. Saving of the PS/PN program
2. Editing of programs (PS program /FB library)
3. Editing of parameters
4. Editing of commentaries (PS program)
5. Writing the PS program to the CPU module

7.5.1 Standard password

The following passwords are pre-set in the factory for first use:

SFLSUNX1, SFLSUNX2, SFLSUNX3, SFLSUNX4

After installation or during first use, all four standard passwords must be replaced (see Chapter 7.5.2).

If the password is forgotten, deinstall and install **SFL-SOFT** once again. The standard passwords will then be available again.

7.5.2 Changing the password

It is only possible to change the password if the edit mode has not yet been released (a password has not yet been entered for an open project). In order to change the password, use the menu entry [Setup] - [Password] – [Change] or the tool button . Now enter the old password (on first start-up **SFLSUNX1**...**SFLSUNX4**) and the new password in password dialogue. For safety purposes the new password must be confirmed once again. The password must have at least 5 characters and may be a maximum of 10 characters long. Small and capital letters are viewed to be different characters.

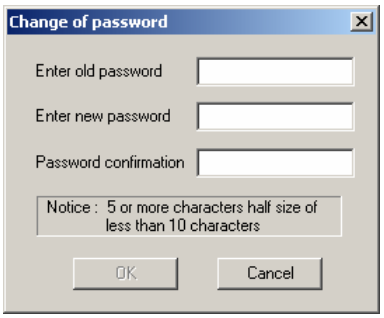


Figure 7-2 Password dialogue (Changing the password)

7.5.3 Release to edit

After starting the program every time, a valid password must be entered in order to release a project for editing. This can either be done under the menu option [Edit] [Start Edit Mode] or the tool button .

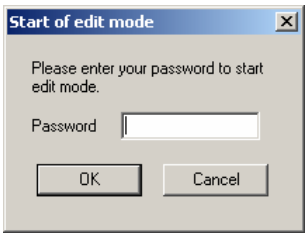


Figure 7-3 Password entry

7.5.4 Program lock

Manual program lock

In order to stop editing after the password has been entered (e.g. in the case of brief absence), the complete program can be locked against any entry. Use the menu option [Option] [Lock **SFL-SOFT**] or alternatively the tool button . The lock can only be released using the password used to log in (the password with which editing was released) but not with one of the other three..

Figure 7-4 Program lock dialogue

Automatic program lock

This option automatically locks the program after an adjustable time of inactivity (no mouse or keyboard action). The time period is activated and set in password dialogue [Setup] – [Password]. The waiting time is freely adjustable between 1 and 120 minutes.

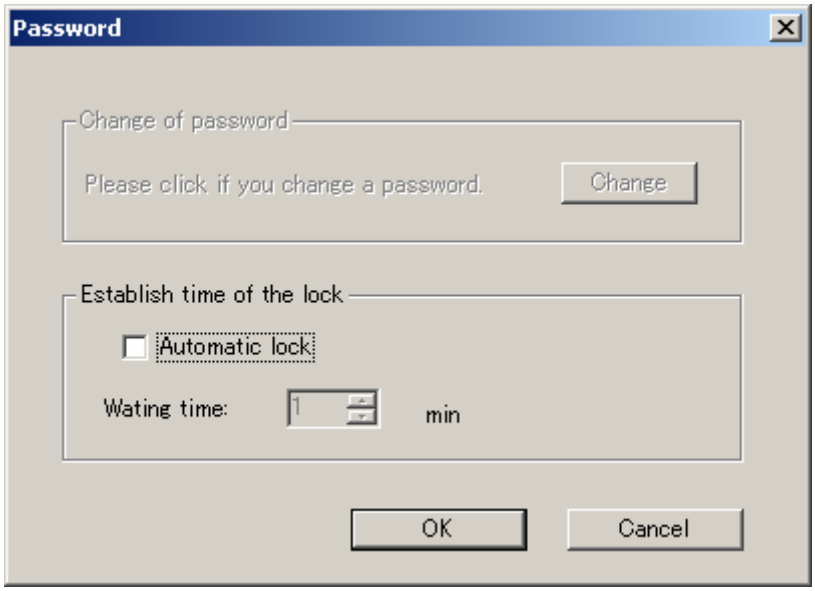


Figure 7-5 Password dialogue (Automatic program lock)

7.5.5 Setting the protection (program password)

This function makes it possible to protect the current program with an individual password. The transfer to the CPU module is then no longer possible.

The password may have 5 to 20 characters. After entry of the password the protection is activated after having saved the program.

If the protection is active, a key symbol appeas in the project window to display the status (see Figure 7-18). It is not possible to edit the marked areas.

In order to protect the program use the menu option [Option] – [Setup Protect protection].

Available options are the protection of

- “PS + Parameter(PS)” or
- “PS + PN + Parameter”.

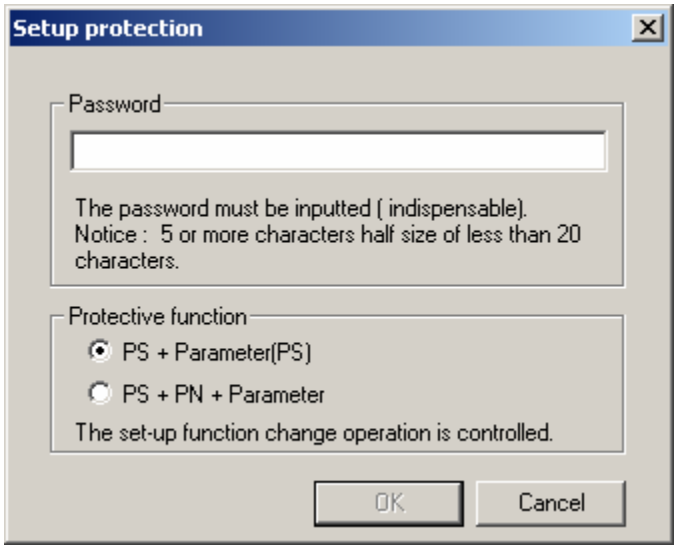


Figure 7-6 Program protection dialog

It is possible to cancel the protection of a program by entering the password. Use the menu option [Option] – [Reset protection] for this purpose.

Figure 7-7 Dialogue Program protection dialog

7.6 PIN Code

When transferring the PS program the dialog to enter the PIN code is displayed. This dialog is used to enter or alter the PIN code.

The PIN code is ‘0000’. The user can select the figures ‘0001’ to ‘9999’ as PIN code.

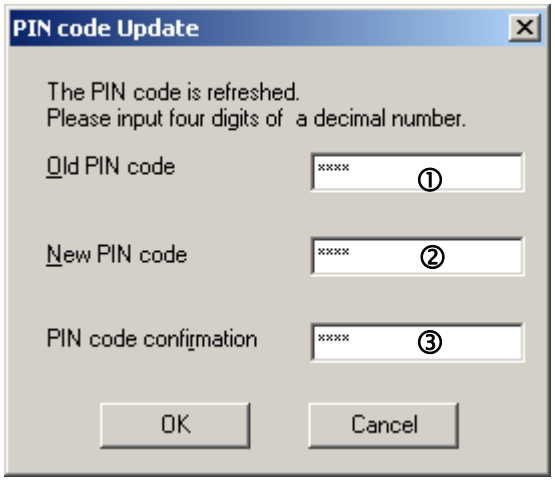


Figure 7-8 Entry of the PIN code

① Old PIN code
Entry of the current PIN code. The PIN code is displayed as ‘*’.

② New PIN code
If necessary, a new PIN code can be entered. The old PIN code can be entered here if no change of the PIN code is desired. The PIN code is displayed as ‘*’.

③ PIN code confirmation
The new PIN code ② must be entered again for confirmation. It is again displayed as ‘*’.

7.7 Program/data transfer

7.7.1 Connecting to SFL

The connection to **SFL** is made via a standard USB printer cable.

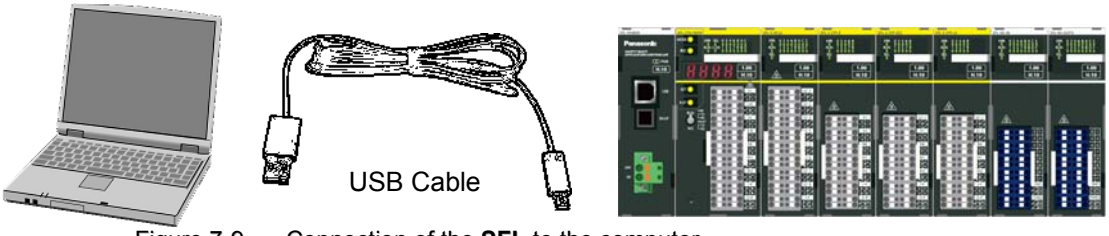


Figure 7-9 Connection of the **SFL** to the computer

It is only possible to connect the **SFL** with a different device via a USB cable for parameter assignment or for short-term trouble shooting.

A permanent connection is not permitted.

Recommended USB cable

USB Cable			CLAMP FILTERS	
Manufacturer	Type	Cable length	Manufacturer	Type
ELECOM CO.,LTD	USB2-FS05	0.5m	unnecessary	
	USB2-FS15	1.5m		
	USB2-FS3	3.0m		
	USB2-50BK	5.0m	TDK Corporation	ZCAT2032-930

7.7.2 Setting time and date

After you have performed a first connection test, you should set the real time clock of the **SFL**. You can reach the dialogue via the main menu [CPU] - [Setup Date/Time]. The real time clock of the **SFL** can be read out here and either the time and the date entered manually or the time and the date taken from your computer.

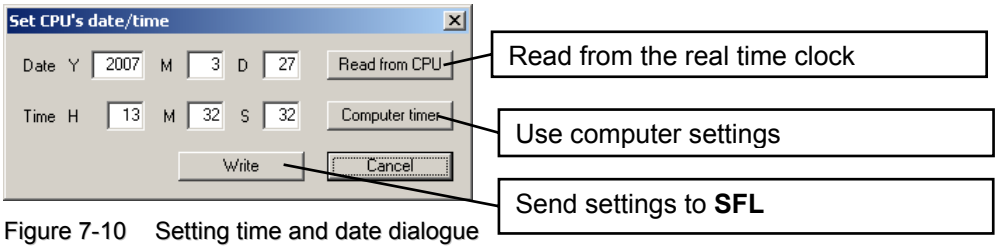


Figure 7-10 Setting time and date dialogue

7.7.3 Communication settings

Before you can exchange data with the, you must first select the correct communication settings (interface).

Creating the communication settings

Select the entry [Setup Communication Port] from the [Setup] menu or use the tool button . The following dialogue appears :

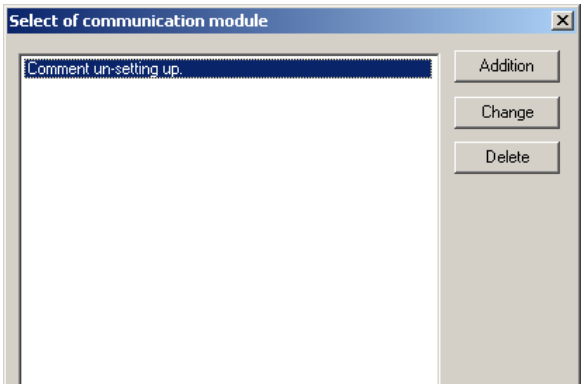


Figure 7-11 communication dialogue **SFL**

You can change, delete or add a new communication setting here. A name can be assigned to each setting. After the first program start (providing you did not install the **SFL-SOFT** beforehand) a setting appears with the name „Comment un-setting up“. You can edit this setting or add a new one.

Communication setup

After clicking the ‘change’ button, the following dialogue appears.

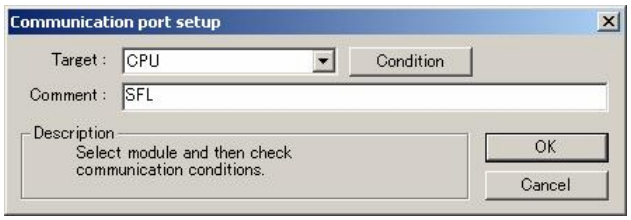


Figure 7-12 Communication setup dialogue

- First select a communication target (currently only CPU module) and then enter a description comment.
- Open the dialogue [Communication modul setup] by clicking the ‘Condition’ button and select the interface.

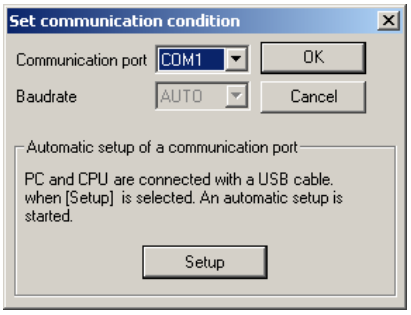


Figure 7-13 Dialog Attitudes Communication

7.7.4 Program transfer

3.7.3 / 3-18

Before a program can be transferred to the **SFL**, you must first release the writing process (see Chapter 3.7.3).

After program creation, logical check and storing of the project, the program can then be transferred to the **SFL**. The menu entry [CPU] - [Write Data] offers several options as to which data should be transferred to **SFL**.

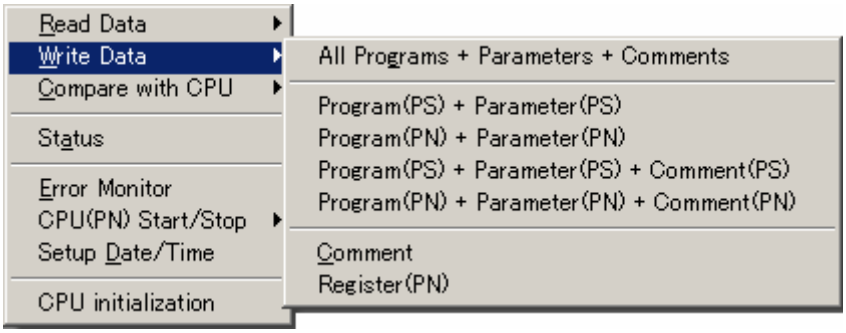


Figure 7-14 Menu to transfer the program

[All Programs + Parameters + Comments]

Transfers the user programs (PS and PN) (machine code and source text) including the machine code of the function blocks (if present), the parameter data and the comments.

[Program(PS) + Parameter(PS)]

Transfers the PS program (machine code and source text) including the machine code of the function blocks (if present) and the specific parameters of the PS program.

[Program(PN) + Parameter(PN)]

Transfers the PN program (machine code and source text) including the machine code of the function blocks (if present) and the specific parameters of the PN program.

[Program(PS) + Parameter(PS) + Comment(PS)]

Transfers the PS program (machine code and source text) including the machine code of the function blocks (if present), the specific parameters and the comments of the PS program.

[Program(PN) + Parameter(PN) + Comment(PN)]

Transfers the PN program (machine code and source text) including the machine code of the function blocks (if present), the specific parameters and the comments of the PN program.

[Comment]

Only transfers the comments.

[Register PN]

Only transfers the memory areas determined for the PN program (EL00W..ELBFW, EM00W..EM7FW, PN-P00W..PN-P1FW, PN-K00W..PN-K2FW, PN-V00W..PN-V0FW, PN-TC00W..PN-TC1FW). This makes it possible to initialise data for the PN program.

SFL-SOFT now prepares for program transfer. Finally, a dialogue appears that shows you the project to be transferred once again and the selected transfer option. You can now decide whether you would like to continue or discontinue transfer.

The transfer procedure contains a so-called read-back routine for safety relevant data. After transfer the data are read back once again and compared with the data sent.

7.7.5 Program comparison

The menu entry [CPU] - [Compare with CPU] offers several options to perform a data comparison:

[Program + Paramter]

Compares the PS and PN user programs (machine code and source text), the function blocks (if present) and the parameters with the data stored in the CPU.

[Register]

Compares all registers (with the exception of the R-register)

[PSV]

Compares only the PSV (Program Specific Value) of the user program.

7.16.1 / 7-97

The PSV can also be read out directly on the CPU modul by **SFL-SOFT**: **SFL-SOFT** automatically inserts the PSV into the cover sheet of the documentation.

7.7.6 CPU Status

The menu entry [CPU] - [Status] shows the current status of the CPU.

Figure 7-15 Display of the current CPU status

7.8 Program description

7.8.1 User interface

The program user interface consist of 4 areas.

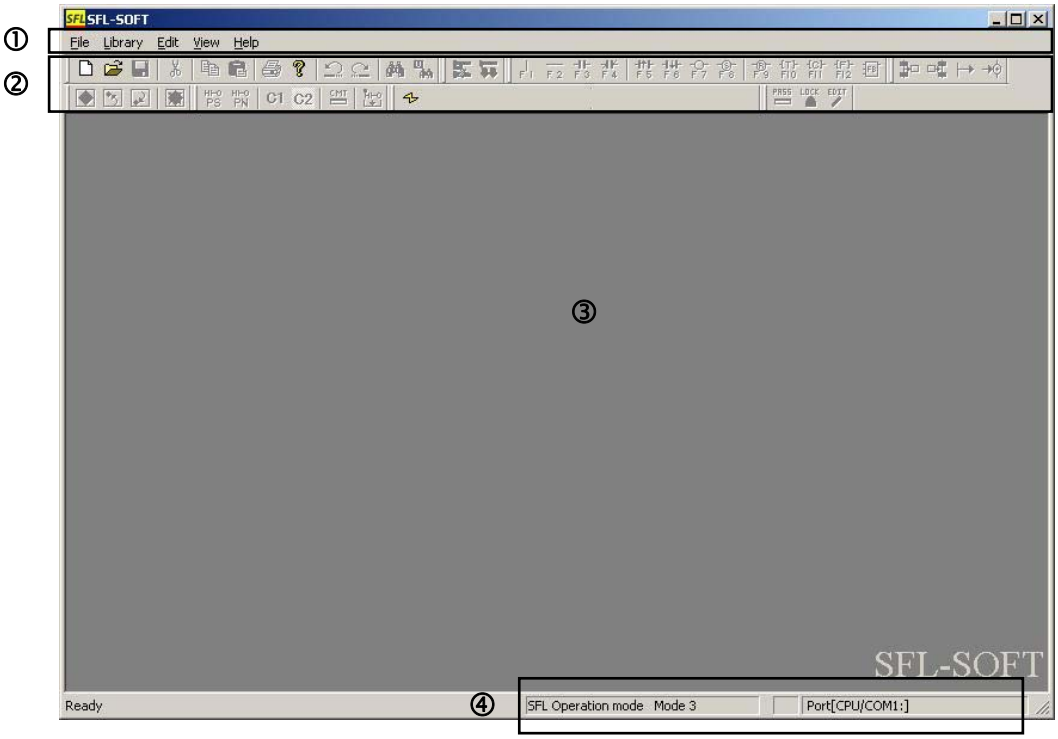


Figure 7-16 **SFL-SOFT** program user interface

① Main menu

7.8.2 / 7-16

Provides access to the program functions. Depending on the program status or possible actions, not all menu points will be available (shown in grey or faded out).

② Tool bar

The tool bar provides fast and easy access to the majority of functions in the form of icons, so-called tool buttons. Depending on program status and possible actions, some buttons may be faded out. The tool bar is subdivided into individual groups. Each group has a grip (vertical line) for positioning within the tool bar. For this purpose the grip must be clicked with the left mouse button. By holding down the mouse button the group can now be positioned within the tool bar. A group can also be pulled out of the tool bar and is then displayed as an independent window.

7.8.3 / 7-18

③ Client area

7.8.4 / 7-21

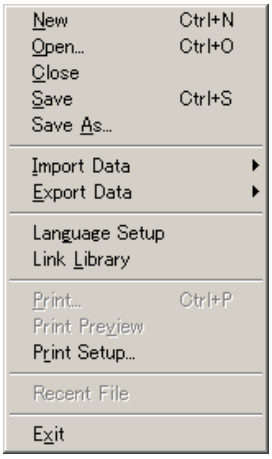
This is where the project window, the ladder diagram and the parameter window are displayed. Both windows can be freely arranged.

④ Status bar

Shows additional information on the active menus/buttons and status information.

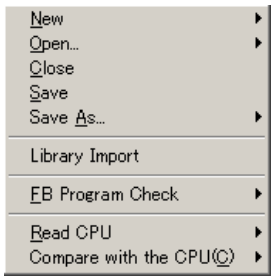
7.8.2 Main menu

File



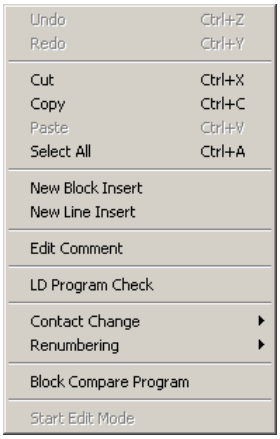
➔Creates a new project
➔Opens an existing project
➔Closes an opened project
➔Saves an opened project under current name
➔Saves an opened project under a new name
➔Imports programs/comments/register data
➔Exports comments/register data
➔Setup languages for comments
➔Integrates a library
➔Prints (draft)
➔Prints (documentation)
➔Selects and sets-up printer
➔Shows the last opened projects (max. 4)
➔Exits the program

Library



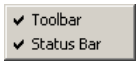
➔Creates a new library
➔Opens an existing library
➔Closes an opened library
➔Saves an opened library under the current name
➔Saves an opened library under a new name
➔Imports library
➔FB Program Check
➔Reads library from CPU
➔Compares a library with the library saved in the CPU

Edit



➔Undoes the last action performed
➔Restores an undone action
➔Cuts out a selected object
➔Copies a selected object
➔Inserts a copied or cut-out object
➔Selects all objects
➔Inserts a new block
➔Inserts a new line
➔Edits comments
➔Checks the program (ladder diagram)
➔Changes the type of contact
➔Renumbers the edge address (P)
➔Compares two programs block for block
➔Enters the password for the edit mode

View (LD window inactive)



➔Displays/conceals tool bar
➔Displays/conceals status bar

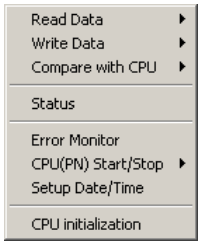
View (LD window active)

➔View of the ladder diagram with / without comments
➔Selects type of block numbering
➔Displays block comments
➔Displays identifier
➔Searches for contacts/functions and comments
➔Searches for U marks
➔Jumps to specific position in ladder diagram
➔Edits bookmark
➔Displays use status of the memory
➔Displays the links of the contacts in tabular form
➔Displays the property dialogue for contacts and functions
➔Displays comment 1 (language 1)
➔Displays comment 2 (language 2)

Converting

➔Converts modified blocks into machine code
➔Defragments the FB execution memory (Chapter 7.12.8).
➔Checks program size (Chapter 7.13.3)

CPU

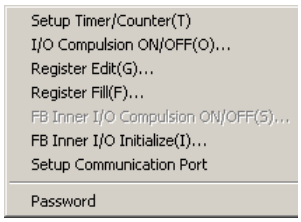


➔Reads data from the CPU module.
➔Transfers data to the CPU module
➔Compares data of the CPU module
➔Displays current operating status of the CPU module
➔Reads out the error memory of the CPU module
➔Starts/stops the CPU module (PN Program only)
➔Sets the real time clock of the CPU module
➔Initialises the CPU module

Monitor

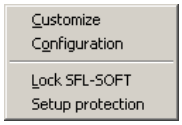
➔Starts/stops monitoring of the program (Chapter 7.15)
➔Start/stops monitoring of the register (memory area)

Setup



➔Not available
➔Sets memory areas of the PN program (Chapter 7.10.3)
➔Edits memory areas of the PN program
➔Not available
➔Not available
➔Resets all flags in the FB
➔Selects communication module (Chapter 7.7.4).
➔Changes password (Chapter 7.5.2).

Options

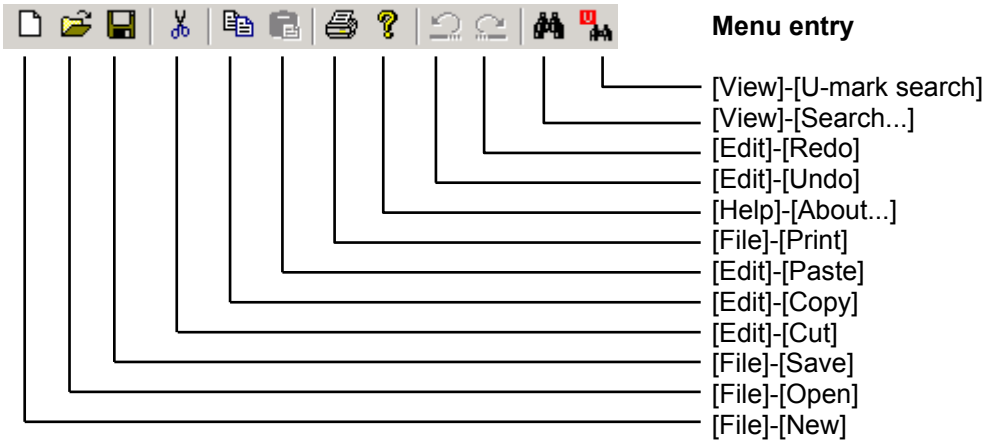


➔Modifies the presentation of the program user interface (Chapter 7.17.1).
➔Modifies program settings (Chapter 7.17.2).
➔Locks programs for editing (Chapter 7.5.4).
➔Sets program protection (Chapter 7.5.5).

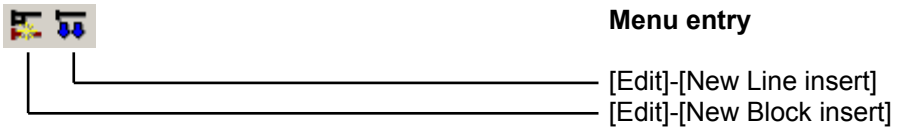
7.8.3 Tool bar

The functions of the tool bar are found as subentries in the main menu. If you move the mouse over the individual tool buttons, an information window appears with a brief description of the action associated with the tool button. In addition, a brief description of the function is displayed in the status bar. A reference to the menu function belonging to the tool buttons is shown below.

Edit

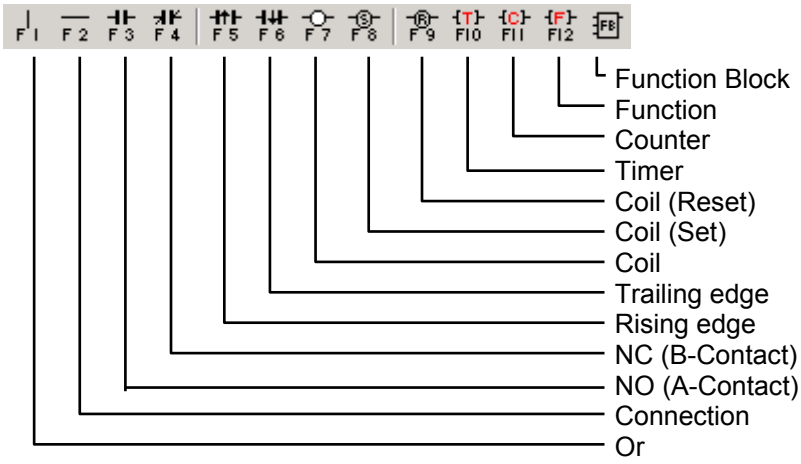


Ladder diagram

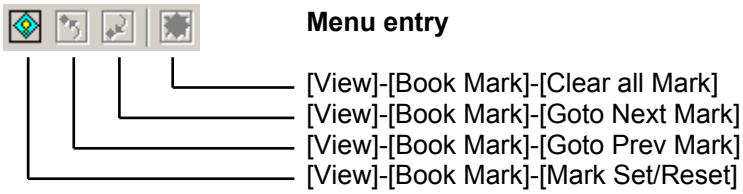


Contacts

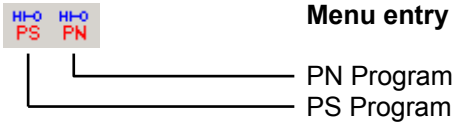
The symbols cannot be accessed via a menu entry. To select a contact, you can also use the function buttons in addition to the left mouse button. The parameters of the selected contact can be set in the property dialogue (Chapter 7.10.11). This is opened by double clicking on the appropriate contact (already marked) or via the pop-up menu (right mouse button). A detailed description of the individual contacts is provided in 7.10.11.



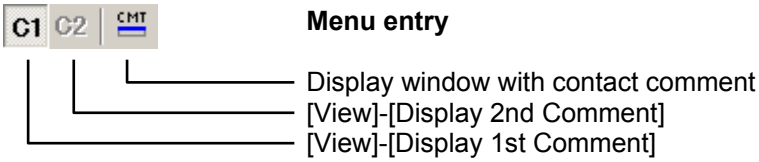
Book Mark



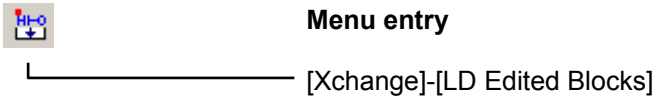
PS/PN



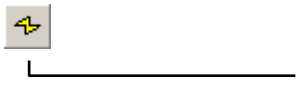
Comments



Xchange



Communication



Menu entry

[Setup]-[Setup Communication Port]

Password

Menu entry

[Edit]-[Start Edit Mode]
[Option]-[Lock SFL-SOFT]
[Setup]-[Password]

7.8.4 Client area

Project window

7.12.2 / 7-63

Different project settings can be made here. The individual options are accessible via a tree structure. A description of the tab [Library] can be found in Chapter 7.12.2.

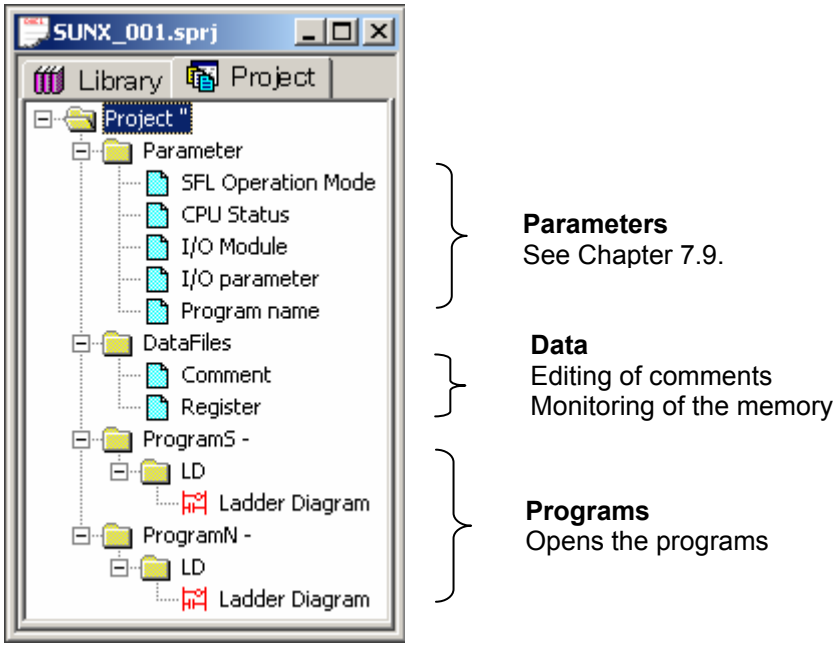


Figure 7-17 Project window

If the program is protected, the ‘lock’ symbol for the protected elements is displayed. See Figure 7-18.

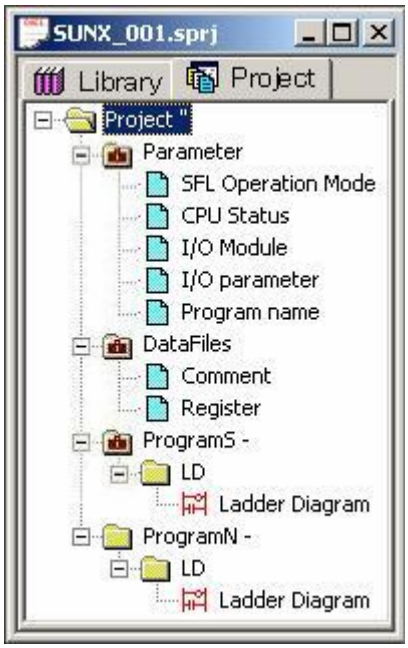


Figure 7-18 Project window with protective program

Ladder diagram

The program is actually created in the LD window. The symbols are inserted via the 'contacts' tool bar or property dialogue. The program is executed block by block from the top to the bottom and once it reaches the bottom end, starts from the top again. Within a block the individual instructions are processed from left to right. Only contacts can be inserted on the left hand side. Coils and functions are always arranged to the far right. Every block has a block number which represents a check step during logical program checking (see Chapter 7.13.2 / エラー! ブックマークが定義されていません。). A comment can be entered for every block and for every contact/coil/function. Block comments (maximum 4 x 50 characters long) are displayed directly in the LD window and contact comments (maximum 100 characters long) in their own window. The contact comments window can be displayed/concealed with the tool button .

7.13.2 / 7-84

The display of the ladder diagram differs as follows for the PS and PN program:

- PS program : background WHITE
- PN program : background GREEN

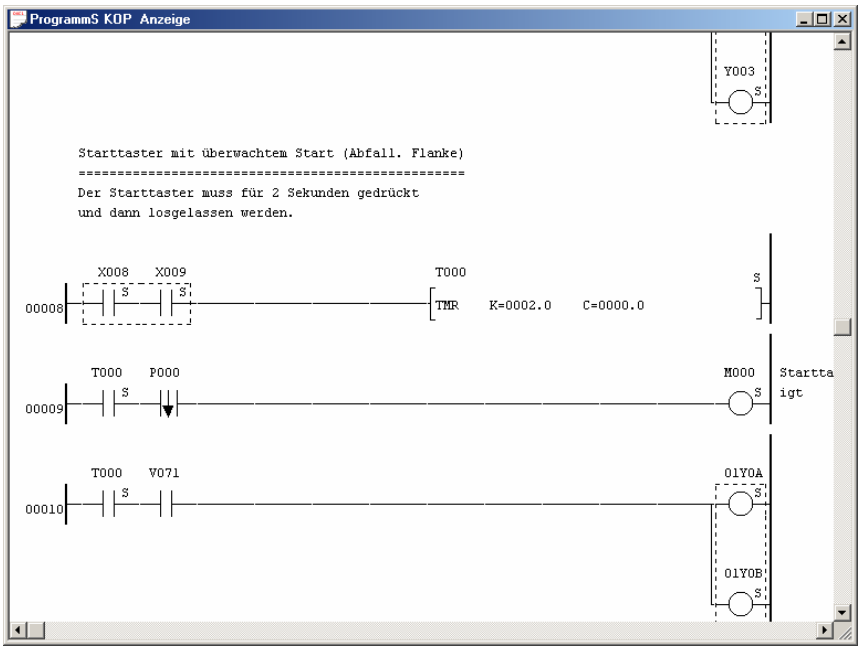


Figure 7-19 PS ladder diagram window

7.9 Parameter assignment

7.9.1 SFL operating mode

The operating mode of the **SFL** can be displayed under **SFL** operating mode in the project window. Setting the operating mode is planned for future extensions.

3.4 / 3-6

More detailed information on the operating modes is provided in Chapter 3.4.

Figure 7-20 Operating modes of the **SFL**

The current operating mode of the **SFL** is displayed in the status bar of **SFL-SOFT**.

Figure 7-21 Operating modes of the **SFL**

7.9.2 CPU program execution

The execution of the PS and of the PN program can be determined in this dialogue (project window under the entry [CPU Status]).

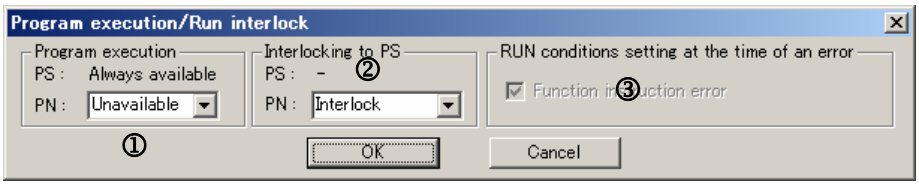


Figure 7-22

① Program execution

The PS program must always be available and is always executed. If a PN program is to be executed, the PN program must be created and the program execution mode is to be set to 'available' for the PN program.

The default setting of the PN program is 'not available', i.e. an existing PN program is not executed in this case.

② Interlocking to PS

The execution of the PN program can be influenced by **SFL-SOFT** in the menu option [CPU] - [CPU (PN) Start/Stop]. Depending on the option selected a Start/Stop of the PN program will have the following effects on the PS program:
'Locking': when the PN program is stoped, the PS program is also halted. All outputs of the PS program switch to the safe state; the outputs of the PN program are frozen. It is only possible to leave this state by actuating the R.ST. button on the CPU module.
'No locking': Stop/Start of the PN program has no influence on the PS program.

③ Troubleshooting in the PN program

This function is planned for future extension of troubleshooting in the PN program. If an error arises in the PN program, then the PS and PN programs are stopped. All the safe outputs move to the safe state. The operational outputs are 'frozen'. It is only possible to leave the state via R.ST. or Power ON/OFF. The error code 71 is shown on the CPU.

7.9.3 I/O Modules

Which modules are assigned to which individual stations is specified in the project window under the [I/O-Module] entry.

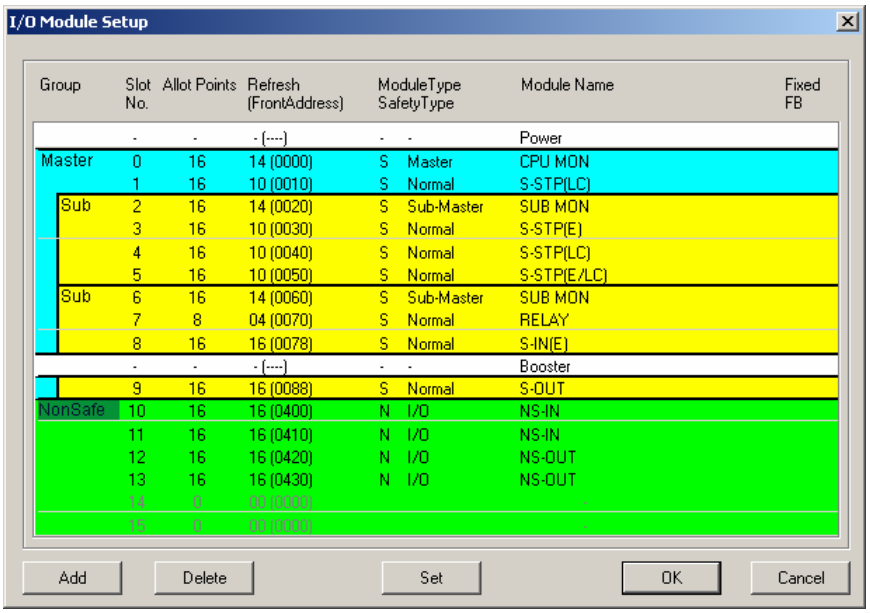


Figure 7-23 I/O module setup dialogue

The individual elements are marked in different colours as follows:

- Master group blue
- Sub-Master groups yellow
- Operational I/O's green

Please bear in mind that in the case of discrepancies between the configuration set here and the actual configuration it is possible to create a program and to transfer it but not to execute it. **SFL-SOFT** has no information on your hardware configuration. Before creating the program, compare your settings with the hardware to ensure that, for example, an input has not been configured as an output module or a non-existent slot has not been configured by mistake.

Group

Display of the corresponding group

Master : group controlled by the CPU module
Sub : group controlled by the SUB-MON module
NonSafe : group of the operational I/O

Slot no.

Display of the slot

Reserved channels

Number of the reserved channels of the module. In order to simplify the counting of the I/O's, a module always occupies a multiple of 8 channels. This ensures that the counting of the I/O's of each module always starts with ..0 or ..8.

Channels used

Number of physically existent I/O channels of the module.

Module Type

Specification of the module status

- S : safe I/O module
- N : operational I/O module

Specification of the module type

Master : controls the entire system
Sub-Master : controls the assigned group
Normal : I/O module of a group
I/O : operational I/O

The check is executed in the described way only in Mode 1. The functions are freely available in Mode 3.

Module name

Name of the module

Registered FB

This display is planned for future extensions.

Selection of the I/O module

A new module may be added by selecting [Add].

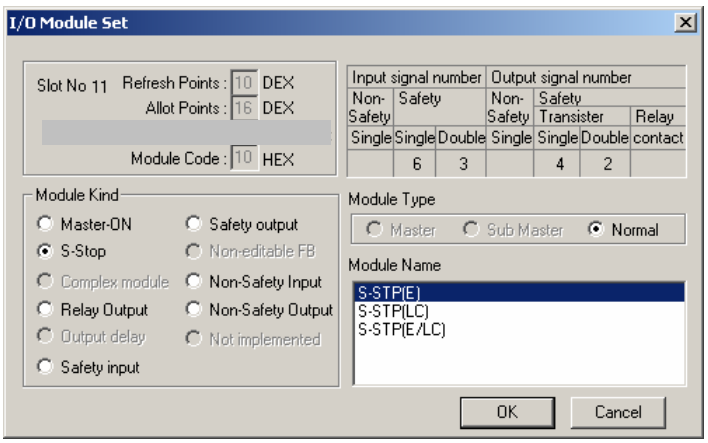


Figure 7-24 Selection of a module

The required module is to be selected and the selection confirmed with [OK].

Channels used

Number of the physically existing channels of the module.

Reserved channels

The sum of the channels used

Identification code

ID code of the module

Module identification

Selection of the module type.

Module type

Specification of the module type.

Module name

Name of the module to be selected.

7.9.4 I/O parameters

The settings of the individual modules are made in the project window under the [I/O Parameter] entry.

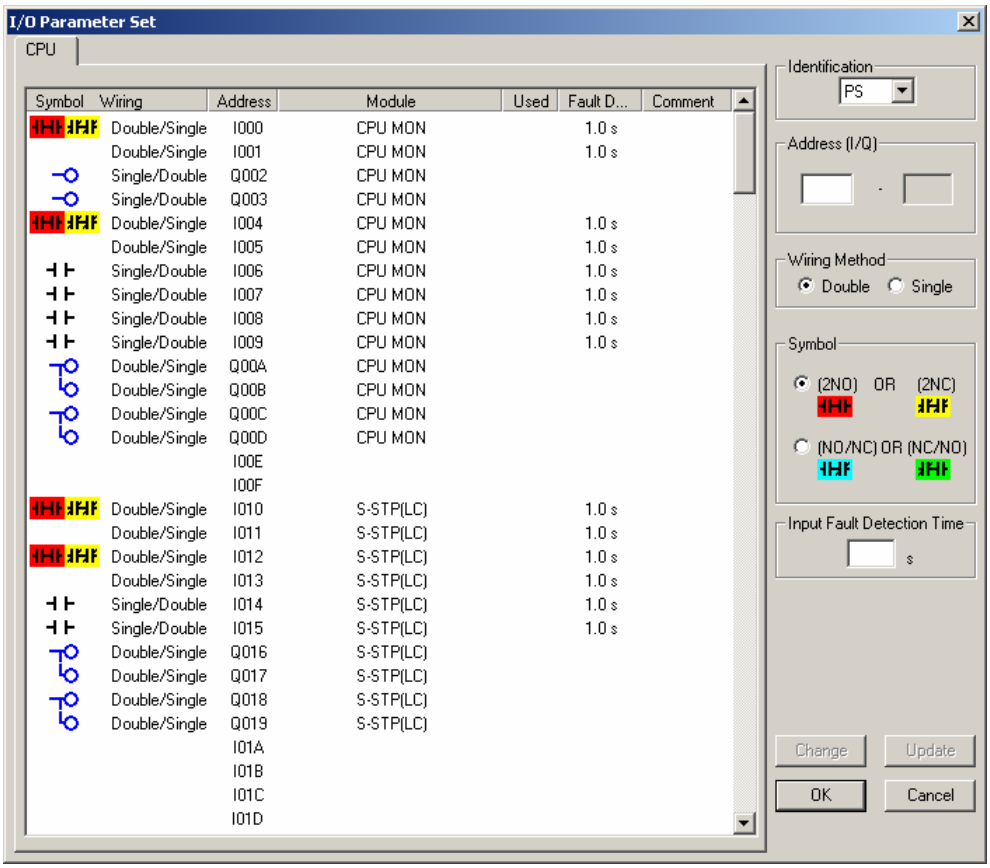


Figure 7-25 I/O parameter dialogue

The left part of the window lists all I/O channels of the module used in tabular form. The settings for one or several selected I/O channels may be modified in the right part of the window.

Table of the I/O channels

□ ① □② ③ ④ ⑤ □⑥

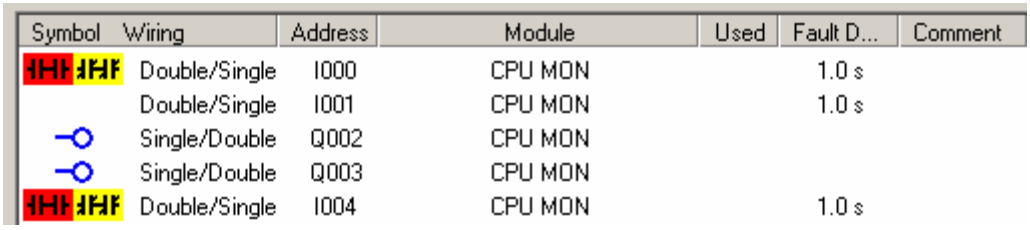
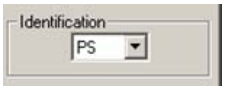


Figure 7-26 Channel table

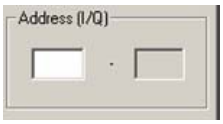
① Type of contact shown and wiring of the contact in symbolic form
② Address of the channel within the system
③ Type of module belonging to the channel
④ Shows whether the channel is used in the program
⑤ The channel difference time set
⑥ Comment

PS/PN switch over

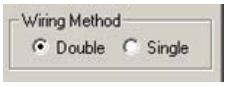


Switches between the parameters of the PS and PN program.

Channel settings

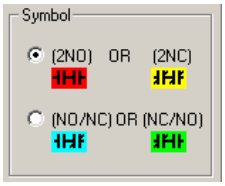


In order to select several channels you can enter an address area here alternatively to the left mouse button whilst simultaneously pressing the SHIFT button.



Select here whether the selected channel is to be wired in a double or single manner.

In the case of safety-oriented circuitry starting from SIL 2, category 3/PLd, a 2 channel wiring (wiring method = double) will be necessary.



This option is only available in the case of 2 channel wired inputs. You can select here between valent (NC / NC, NO / NO) or exclusive (NC / NO, NO / NC) combinations.

The channel difference time permits a time delay in the switching sequence between the individual channels with 2 channel circuitry. The setting range is between 1.0 and 9.9 seconds.

The default of the ‘channel difference time’ is 1.0 seconds. Only change this default if it can be seen that the sensor used will not be able to observe this time.

7.9.5 Program name

The program name is entered in the project window under the entry [Program name].

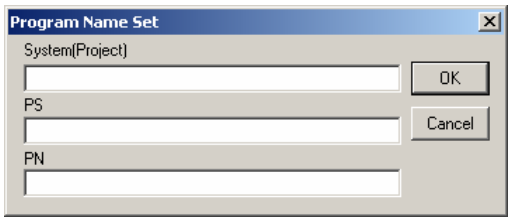


Figure 7-27 Program name dialogue

Using this dialogue, a name can be assigned for the system, the PS and the PN program. The assigned name for the system (project) is displayed in the project window.

7.10 Programming

The programs are created exclusively graphically in the form of a ladder diagram. Since the **SFL-SOFT** programming software is primarily intended for the creation of user programs for the safety-orientated use of the **SFL** only those instructions and functions of IEC 61131 are available which can be checked in terms of safety using simple means. The statement list programming language used to check the user program contains a few necessary additions which are not a part of the IEC 61131.

It is assumed that the programmer has already gathered appropriate experience with the programming languages of ladder diagram (LD) and statement list (STL). If the requisite basic know-how does not exist, intensive familiarisation with the theme is necessary before creating the program. Even if the **SFL** controller satisfies the highest safety requirements in machine / man protection, this is no protection against faulty (from a safety point of view) user programs. The integrated program check can merely test the logical structure of the user program.

7.10.1 Basic program flow

Before executing the user program, the states of the inputs are queried and stored. The inputs are not queried during execution of the user program.

The user program is subdivided into several numbered contact blocks. Each block can consist of several lines. A line constitutes the equivalent of an electrical connection from the input (left) to the output (right). An input is always a contact (reading of an address) and the output is always a coil / function (writing of an address).

A block consists of at least one line with at least one coil. The starting point of a block is always 1/high. A block can consist of a maximum of 22 lines and a total of 256 symbols (contact, coil, function). The number of blocks is only restricted by the maximum admissible program size or run time.

7.13.3 / 7-90

Please bear in mind during program creation that the user program is always worked through from top to bottom. The unfavourable placement of the individual program blocks has a very negative effect on the reaction time. This applies in particular to the use of flags. Therefore always switch safety enables (coils) directly and ensure that the safety enables are shut down after a change in level at the input in the same program run.

7.10.2 Creating the PS program

SFL-SOFT is required to create the PS program. After stipulating the system configuration, the ladder diagram can be selected in the project window in the [ProgramS / LD / Ladder Plan] path.

After creation the program must be stored whereby a logical check and translation is performed.

7.10.3 Creating the PN program

SFL-SOFT is required to create the PN program. After stipulating the system configuration, the ladder diagram can be selected in the project window in the [ProgramS / LD / Ladder Plan] path.

After creation the program must be stored whereby a translation is performed.

3.7.6 / 3-26

For debug purposes it is possible under menu entry [Setup] - [I/O Compulsion ON/OFF] to set the following specifiers (see Chapter 3.7.6) insofar as these have not been set in the PN program.

- M flag
- K flag

7.10.4 Commands / symbols

The following table provides an overview of the existing ladder diagram symbols and their equivalent representation in the 'statement list' programming language used for checking purposes. Not all symbols in the ladder diagram column are directly available to the programmer. A few are available from the user program created by the programmer. The description of the symbols available for program creation and the respective statement list of the timer and functions are provided following this table.

7.13.2 / 7-84 The translation process of the 'ladder diagram' programming language to the 'statement list' programming language necessary to check the program is described in Chapter 7.13.2 using a detailed example.

Translation table

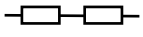
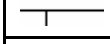
Ladder diagram	Statement list	Function
	STR	Start of an operation with NO (A contact)
	STR NOT	Start of an operation with NC (B contact)
	AND	Series connection (A contact)
	AND NOT	Series connection (B contact)
	OR	Parallel connection (A contact)
	OR NOT	Parallel connection (B contact)
	AND STR	Series connection of blocks
	OR STR	Parallel connection of blocks
	OUT	Coil output
	PTS	Edge detection, rising edge
	NTS	Edge detection, trailing edge
	FPS	Start of branching double coil
	FRD	Branching of double coil
	FPP	End of branching double coil
	FST	Unconditional output
	NOT	Inversion
		Connection

Table 7-2 Translation table for ladder diagram/statement list instructions

Description

Markings

As the **SFL** can also be used for control tasks that are not safety-oriented, all ladder diagram symbols in the PS program (contact, coil, timer …) have a marking to show the programmer which symbols may be used for safety functions.

Marking	Safe	Description
“U”	no	A safe symbol was overwritten by a non-safe symbol. The symbol is not safe and may not be used for safety functions.
“ ”	no	Symbol is not safe and may not be used for safety functions.
“S”	yes	Symbol is safe and may be used for safety functions.
“S2”	yes	Symbol ist safe in accordance with the requirements of category 2 and may be used for safety functions. The second set down path is to be realised in accordance with Chapter エラー! 参照元が見つかりません。.

Table 7-3 Markings of the ladder diagram symbols

In order to find the cause of a ‘U’ mark more easily, these are highlighted in two different background colours

U	RED	appears in the line containing the cause of the ‘U’ mark.
U	YELLOW	appears in the line that contains a ‘U’ mark due to an existing ‘U’ mark.

The symbol marking is performed in accordance with a system of rules of which the main rules are described in the following.

The observation of the rules is ensured by **SFL-SOFT** and need not be understood by the programmer in all cases. However, an understanding makes it easier for the planning/programming in advance to attain the desired safety.

1. Inverted inputs (B contact) are **not safe** and are marked **‘ ’**.
2. Self-holding (activation by own state) is **not safe** and marked with **‘ ’**.
3. A parallel connection (OR operation) from one or more **‘S’** or **‘S2’** and at least one **‘ ’** or at least one **‘U’** produces a **‘U’**.
4. A branch containing a **‘U’** (direct or indirect) always gives rise to a **‘U’**.
5. A series connection (AND operation) from **‘ ’** and at least one **‘S’** produces an **‘S’** or an **‘S2’**.

For safety-oriented control tasks only ladder diagram symbols to switch outputs marked with an ‘S’ or an ‘S2’ may be used. This information does not relieve the programmer of the responsibility to carry out his own evaluation of the user program.

The ‘S’ or the ‘S2’ mark does not made any declaration about the reached SIL, category or an PL. Therefore the combination of Sensor - **SFL** – Actor under provision of the **SFL** Application program must be taken in consideration.

Connection

Horizontal connection

Passes the left state on to the right connection in an unmodified form.

Vertical connection (OR)

Initiates an OR logical operation.

Negation

Inversion of the signal level.

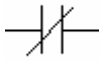
A wrong use of negation can but need not lead to an infringement of the closed-circuit current principle. An assessment can only be made if the entire PS program is viewed together with the application. In order to simplify this assessment, all negations used are highlighted with a red background.

Contacts (inputs)

NO (A contact)

Passes the left state on to the right connection if the contact state is TRUE. Otherwise right is always FALSE.

NC (B contact)



Passes the left state on to the right connection if the contact state is FALSE. Otherwise right is always FALSE.

Edge detecting contacts

Edge detecting contacts always keep their TRUE state for only one program cycle. Further processing must be made after these contacts therefore.

Rising edge

The right connection is only TRUE if the left contact state has a rising (positive) edge. Otherwise right is always FALSE.

Trailing edge

The right connection is only TRUE if the left contact state has a trailing (negative) edge. Otherwise right is always FALSE.

Coils (outputs)

Coils

Passes the left state on to the address allocated to the coil.

7.10.5 Timer

7.10.2 / 7-49

The number of available timers is 256 in the PS program and 512 in the PN program. They are selected by assigning an address with the prefix 'T' (PS:T000…T0FF, PN:T000…T1FF). A timer passes a TRUE at its input to its output with a time delay. Depending on the requirements of the circuitry, a selection can be made between a self-holding and a non-self-holding timer. Both types are available with 2 different increments (10ms and 100ms). The adjustable ranges are between 0.1s…6553.5. During address assignment (PS:T000…T0FF, PN:T000…T1FF) the timer is automatically assigned an address from the 'N' memory area with the same address. This register serves the timer as counting register and permits the current counter reading to be read out.

Depending on the capacity stage of the **SFL** there are different tolerance ranges for the timer.

Timer	Increments	Tolerance
TMRH	10ms	-25ms…+15ms
TMR	100ms	-115ms…+15ms

Table 7-4 Timer tolerances

Please observe these tolerances when creating programs and use the TMRH timers to produce more exact time delays.

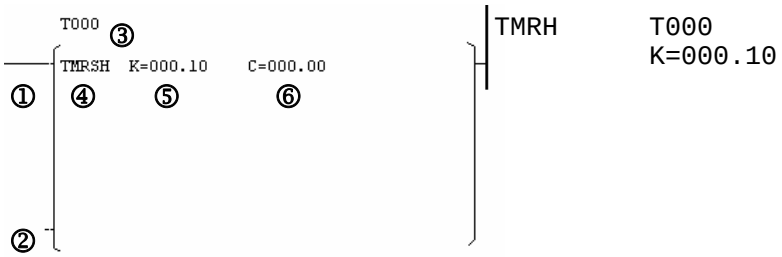


Figure 7-28 Ladder diagram symbol/statement for function timer

① **counter input**

② **reset input (only for self-holding TMRS/TMRSH)**

③ **address**

④ **type of timer**

TMRH = Incrementing in 10ms steps (0,01s…655,35s)
TMR = Incrementing in 100ms steps (0,1s…6553,5s)
TMRHS = Incrementing in 10ms steps; self-holding (0,01s…655,35s)
TMRS = Incrementing in 100ms steps; self-holding (0,1s…6553,5s)

⑤ **set time in seconds**

⑥ **current time under monitoring**

TMR/TMRH

This timer passes on a TRUE to its input after expiry of the delay time at its output and maintains this state until its input resumes the FALSE state.

TMRS/TMRSH

This timer passes on a TRUE at its input to its output after expiry of the delay time and maintains this state (independent of the state of the input) until its R-input assumes the HIGH state. Due to its self-holding function this timer works like an integrator. Therefore, it also counts pulsed input signals or adds the individual pulse lengths together until the preset total time is reached.

In the case of short pulse times considerable differences may arise between the actual and the measured added time due to the increment of 15ms.

7.10.6 Counters

Counters may only be used in PN program. They are selected by assigning an address with the prefix **'C'** (000…1FF). The adjustable range is between 1 and 65535. A counter passes on a TRUE to its input if the number of counter pulses exists.

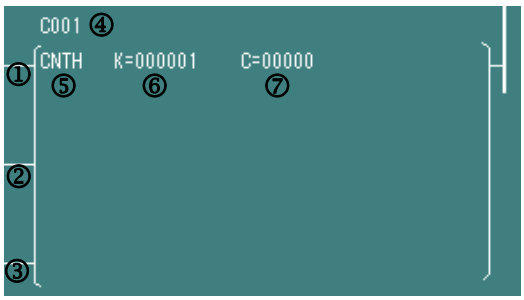


Figure 7-29 Ladder diagram symbol/counter function statement

① counter input
② up/down input (only CNTH)
③ RESET input
④ address
⑤ type of counter
⑥ set value
⑦ current value of the counter

CNT

This is an incrementing counter. If the RESET input is LOW, the value is incremented every time if a rising edge is at the counter input. The output is HIGH if the counter value reaches the set limit value. In the case of RESET input HIGH, the internal counter value is set to 0 and the output to LOW.

CNTD

This is a decrementing counter. If the RESET input is LOW, the value is decremented every time if a rising edge is at the counter input. The output is HIGH if the counter value reaches the value 0. As soon as the RESET input is HIGH, the internal counter value is set to the default value of the output to LOW.

CNTH

This is an incrementing/decrementing counter. If the up/down input is HIGH, it acts like an incrementing counter, and if it is LOW it acts like a decrementing counter.

7.10.7 Functions

7.10.12 / 7-50

A function is selected in the [Property] dialogue (to be reached via the main menu [View] - [Property] or alternatively with a right mouse click on a symbol). In addition to its function name, every function is defined by a clear function number. The functions are stored internally in the **'F'** memory area which is not accessible to the programmer.

Flow control

START

Figure 7-30 Ladder diagram symbol/statement for START function

Marks the program start. All blocks between START and END are executed in an endless loop.

In order to execute blocks only once directly after initialisation (during the first program run), you can also place your statements before the START function as an alternative to interrogating the special flag VO6.

END

Figure 7-31 Ladder diagram symbol/statement for END function

Marks the end of the main program. All statements between END and PEND can only be achieved by branches or subprogram calls.

This function has only been implemented for reasons of compatibility with other control types and is without significance to the programming of the **SFL**.

PEND

Figure 7-32 Ladder diagram symbol/statement for PEND function

Marks the end of the program. Statements after this function are not taken into consideration.

If you do not wish to execute specific program parts for the time being during the program test or during trouble-shooting, move these to behind the END or PEND function. This will save you the necessity of storing several test versions.

Application instructions (PN program only)

Comparison

Command	OP1	OP2	Function
=H W=H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '=' is satisfied, then the result is HIGH.
=N W=N	S_1	S_2	
<>H W<>H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '<>' is satisfied, then the result is HIGH.
<>N W<>N	S_1	S_2	
>H W>H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '>' is satisfied, then the result is HIGH.
>N W>N	S_1	S_2	
>=H W>=H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '>=' is satisfied, then the result is HIGH.
>=N W>=N	S_1	S_2	
<H W<H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '<' is satisfied, then the result is HIGH.
<N W<N	S_1	S_2	
<=H W<=H	S_1	H	Comparison of a register with a constant or of two registers. If the condition '<=' is satisfied, then the result is HIGH.
<=N W<=N	S_1	S_2	
Agenda $S_{1,2}$: register, H : hexadecimal constant			

Table 7-5 Overview of the comparison functions

Examples

Set flag M013 if the WORD value of flag K010 is larger than or equal to the WORD value of D0019.

Set flag M014 if the Low BYTE value in the special register D0019 is smaller than or equal to 2Ah.

Data transfer

Instruction	OP1	OP2	Function
MOV WMOV	H	S_1	Load hexadecimal constant H to S_1.
MOVE WMOVE	S_1	S_2	Load decimal constant S_1 to S_2.
BCD WBCD	S_1	S_2	Translate data of S_1 from binary to BCD and load these data to S_2.
BIN WBIN	S_1	S_2	Translate data of S_1 from BCD to binary and load these data to S_2.
Agenda $S_{1,2}$: register H : hexadecimal constant D : decimal constant			

Table 7-6 Overview of the data transfer functions

Examples

Write 10h as WORD value in flag K000.

Write the LOW Byte of the special register D0004 to the LOW Byte of flag K001.

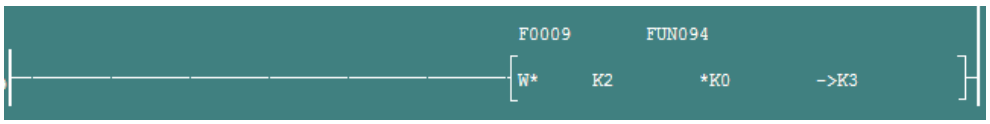
3.7.6/ 3-26 In case of reading a Counter Value with one of the Data Transfers commands in a Function block, the Variable of the Counter Value must be preceded by an “&”.

Arithmetical functions

Instruction	OP1	OP2	OP2	Function
W+	S_1	S_2	S_3	$S_3 = S_1 + S_2$
W-	S_1	S_2	S_3	$S_3 = S_1 - S_2$
W*	S_1	S_2	S_3	$S_3 = S_1 * S_2$
W/	S_1	S_2	S_3	$S_3 = S_1 / S_2$ Store quotients in S3 and the remainder in $S_3 + 1$.
WINC	S_1	S_2		Incrementing of S2 and comparison of the incremented value with S1. If both values are equal, then the result is HIGH. The data are treated as BCD values.
WDEC	S_1			Decrementing the content of S_1. The data are treated as BCD values.
Agenda $S_{1,2,3}$: register				

Table 7-7 Overview of the arithmetical functions

Examples



Multiply the content of the flag K002 with the content of the flag K000 and write the result to flag K003.

Logical functions

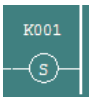
Command	OP1	OP2	OP2	Function
AND	S_1	S_2	S_3	S_3= S_1AND S_2
OR	S_1	S_2	S_3	S_3= S_1OR S_2
NOT	S_1	S_2		S_2= NOT S_1. Bit by bit inversion of the content of S_1
XOR	S_1	S_2	S_3	S_3= S_1XOR S_2 Bit by bit XOR operation of S_1 AND S_2
Agenda $S_{1,2,3}$: register				

Table 7-8 Overview of the logical functions

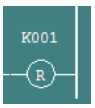
Examples

OR operations of the LOW Byte of flag K003 with the LOW Byte of flag K001 and saving of the result in the LOW Byte of flag K004.

Setting / resetting flags



The assigned flag is set the first time that a HIGH condition arises.



The assigned flag is reset as soon as a HIGH condition arises.

7.10.8 Status flags

The status flags are stored in the 'V' memory area in special flags V50 to V56. Their state is dependent on the result of an executed function. The meaning of the individual flags corresponds to that of the status flag of a processor. An exception is provided by the 'ER' (ERROR) flag. It is set if the **SFL** recognises an invalid operation. When an ER flag is set, the **SFL** controller moves immediately to safe state (all outputs shut down) and transmits an appropriate error message.

Symbol	Name	Address
CY	Carry flag	V56
BO	Borrow flag	V55
Z	Zero flag	V54
ER	Error flag	V50

Table 7-9 Status flags

7.10.9 Memory

3.7.6 / 3-26 

An overview of the individual memory areas and their use is contained in Chapter 3.7.6.

The following tables should contain incomplete address areas. Non-executed addresses shall not be used for programming. They are either reserved for internal use by the **SFL** or are not defined. A reading process of these addresses provides an unforeseeable state.

Communication between PS and PN

The EM (000..7FF) memory area is available for the operational communication between the PS and the PN program.

Range	Direction
EM 000 - 3FF (BIT)	PS➔PN
EM 400 - 7FF (BIT)	PN➔PS

Table 7-10 Communication between PS and PN N

Special flags

8.3.3 / 8-5 

An assignment between set special flag and any accompanying error message is to be found in Chapter 8.3.3.

Address	Function	Description
V001	Serious error	Indicates the occurrence of a serious error (see 8.3.1)
V002	Slight error	Indicates the occurrence of a slight error (see 8.3.1)
V003	Warning	Indicates the occurrence of a warning (see 8.3.1)
V004	Always TRUE	Always has the state TRUE
V005	Always FALSE	Always has the state FALSE
V006	First run	Is deleted after reset once the 'END' function has been achieved.
V027	Program active	Set if user program active
V040	PS program active	Set if PS program active
V041	PN program active	Set if PN program active
V04D	Voltage backup	Set if switched over to backup power
V04E	Reset I/O error	Performs software reset with a rising edge
V050	Error flag	Set if error in function (OR, AND)
V054	Zero flag	Set if result of an operation is zero
V055	Borrow flag	Set if result of an operation is negative
V056	Carry flag	Set with overflow of a function result
V070	Timer 0.1s	Changes state every 50ms
V071	Timer 0.2s	Changes state every 100ms
V072	Timer 1.0s	Changes state every 500ms
V073	Timer 2.0	Changes state every 1000ms
V074	Timer 60s	Changes state every 30000ms
V078	Program run	Changes state each run of the program loop
V080	Reset link error	Performs the software with rising edge
V0C0	CPU error	Set if error in CPU module detected
V0C1	Supply voltage error	Set if power interruption > 10ms
V0C2	Memory error	Set if CRC error flash ROM detected
V0C3	I/O-BUS error	Set if error on backplane BUS detected
V0C4	Special module error	Set if error in special module
V0C5	Parameter error module	Set if parameter assignment in **SFL-SOFT** faulty
V0C6	Parameter assignment error	Set if an error is discovered in the parameter assignment.
V0C7	Error in the I/O module	Set if error in the I/O module is detected (fusing defective, ...).
V0C8	Configuration error module	Set if configuration in **SFL-SOFT** does not comply with the actual state of the hardware
V0C9	Program error	Set if program error recognised
V0CA	Memory error	Set if writing error internal flash ROM recognised
V0CB	Battery error reset	Error after failure of the backup battery not reset with the **SFL-SOFT**.
V0EO	Self diagnosis error	Set if error in the self diagnosis (module) detected
V0E1	Execution time exceeded	Set if the maximum execution time is exceeded
V0E2	PN program error	Set in the case of an error A/B 071
V0E8	PS program error	Execution time of the PS program exceeded
V0F0	Battery error	Set if the voltage for the backup battery too low
V0F5	Real time clock error	Set if error in real time clock detected

Table 7-11 Memory area 'V' (special flags)

Any special flag address that is not listed has the value of 0.

Special registers

Address	Function		Description
S001	Maximum program run time		Constant value 15ms
S002	Minimum program run time		
S003	Actual program run time		
S004	Time (seconds)		Current date and time of the internal real-time clock. The data are stored in BCD format with 1 number per BYTE. Example: '0102' => 12 minutes
S005	Time (minutes)		
S006	Time (hours)		
S007	Time (day)		
S008	Time (month)		
S009	Time (year)		
S00A	Time (day of the week)		
S019	Time (minutes, seconds)		Current date and time of the internal real-time clock. The data are stored in BCD format with 2 numbers per BYTE.Example: '1234' => 12 minutes and 34 seconds
S01A	Time (day, hours)		
S01B	Time (year, month)		
S050 S06F	Error state modules		Description see Table 7-14
S0C0	Initialisation program run time		Constant value 15ms
S0C1	Maximum program run time		
S0C2	Minimum program run time		
S0C3	Actual program run time		
S0E0	Program/ parameter transfer entry 1	Transfer	Bit A = PN program, Bit 9 = PS program
S0E1		min/sec	Date and time of the modification The data are stored in BCD format with 2 numbers per BYTE. Example: '1234' => 12 minutes and 34 seconds
S0E2		day/hour	
S0E3		year/month	
S0E4... S0E7	Program/parameter transmission entry 2 (structure as S0E0...S0E3)		
S0E8... S0EB	Program/parameter transmission entry 3 (structure as S0E0...S0E3)		
S0EC... S0EF	Program/parameter transmission entry 4 (structure as S0E0...S0E3)		
S0F0... S0F3	Program/parameter transmission entry 5 (structure as S0E0...S0E3)		
S0F4... S0F7	Program/parameter transmission entry 6 (structure as S0E0...S0E3)		
S0F8... S0FB	Program/parameter transmission entry 7 (structure as S0E0...S0E3)		
S0FC... S0FF	Program/parameter transmission entry 8 (structure as S0E0...S0E3)		
S200... S24F	Error memory		See Chapter 8.3.2
S2D0	CPU ID		Identification number of the CPU
S2D1	CPU version		Version number of the CPU
S2D2	Version date		Month and day of the version
S2D3	Version date		Year of the version

Table 7-13 Memory area 'S' (special registers)

The memory area S050 contains the error information of all modules. A set bit shows that there is an error at a station in a module at the appropriate slot position.

Address	MSB						Bit								LSB		Station
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
S050	-	-	-	-	11	10	9	8	7	6	5	4	3	2	1	0	CPU

Table 7-14 Error status of the modules

Any special register address not listed here has the value 0. The application can only have access to the areas S0050 and S0130. The contents of the other special registers serve the user as diagnosis aid if he creates a visual presentation of the special register, for example with **SFL-SOFT**.

7.10.10 Addressing

7.9.3 / 7-25 

Addresses are assigned continuously in ascending order according to used slots of the respective back plane. The assignment is solely dependent on the order of assembly of modules and their number of channels. The absolute slot number is not incorporated in this process. Refer also to Chapter 7.9.3 / エラー! ブックマークが定義されていません。.

	0	1	2
Module	CPU	Relay	S-STP-E
Channels	16		
Addresses	I/Q000 to I/Q00F	I/Q010 to I/Q017	I/Q018 to I/Q027

Table 7-15 Address assignment within the station

7.10.11 Comment

Comments can be inputted about a contact, a coil, etc. The comment inputted in different languages can be displayed on a ladder screen by language setting. The language setting is implemented as [File] - [Language Setup]. A language setting is reflected also for printing.

In order to display comments of different language, it is necessary to install the related language fonts beforehand. Please refer to the Windows instructions manual for fonts import.

7.10.12 Program creation

Please bear in mind during program creation that the user program is always worked through from top to bottom. If the individual program blocks are unfavourably placed, the reaction time may be substantially worsened. This applies in particular to the use of flags. Therefore always switch safety enables (coils) directly and ensure that the safety enables are shut down after a change in level at the input in the same program run.

Procedure

The following sequence is to be maintained as a basic rule when creating the user program:

7.9.5 / 7-30
7.9.4 / 7-28
7.7.4 / 7-13

1. Creation of a new project [File] - [New].
2. Stipulation of the program name.
3. Stipulation of the hardware configuration (parameter assignment).
4. Creation of the program (see below).
5. Saving / logical check of the program.
6. Transfer to **SFL**.
7. Confirmation of faultless transmission.
8. Checking of programming in the plant.
9. Printing out and completing documentation.

Inserting a block

In order to position symbols, you must first insert a new block. Select the block under which the new block is to be inserted by clicking on the left mouse button and select [Edit] - [New Block Insert] in the main menu or use the tool button . A new block always consists of one line (connecting path from left to right).

By selecting [Options] - [Configuration] - [Operation] in the main menu, you can select whether new blocks are to be inserted beneath or instead of the selected block. A description of these and other options is provided in Chapter 7.17.2.

Inserting a line

A new line must be inserted for every additional connecting path within a block. For this purpose select the required block by clicking with the left mouse button and select [Edit] - [New Line Insert] in the main menu or alternatively use the tool button

Positioning the symbols

7.17.2 / 7-102

Move the cursor to the required position and select the symbol you wish to insert from the tool bar or alternatively use the function buttons. Depending on options set in the configuration dialogue (see Chapter 7.17.2) the property dialogue is automatically opened. In order to manually display this dialogue, press either on the symbol using the right mouse button and select the [Property Dialogue] entry or use the entry [View] - [Property Dialogue] from the main menu.

Assigning properties

The property dialogue consists of several tabs using which you can select the symbol group.

Contact

A contact corresponds to the reading of an address. The address must be entered hexadecimally as a bit address. In the case of addresses from the word memory area (D, N, S) the corresponding bit position is selected within the address with a suffixed '-', whereby the '□' stands for the corresponding BIT within the WORD.

Address	Description
I001	Input 001 on master station
Q00C	Output 00C on master station
T035	Timer 035
N035-4	BIT 4 of the current counter reading of the timer 035

Table 7-16 Address assignment for contacts

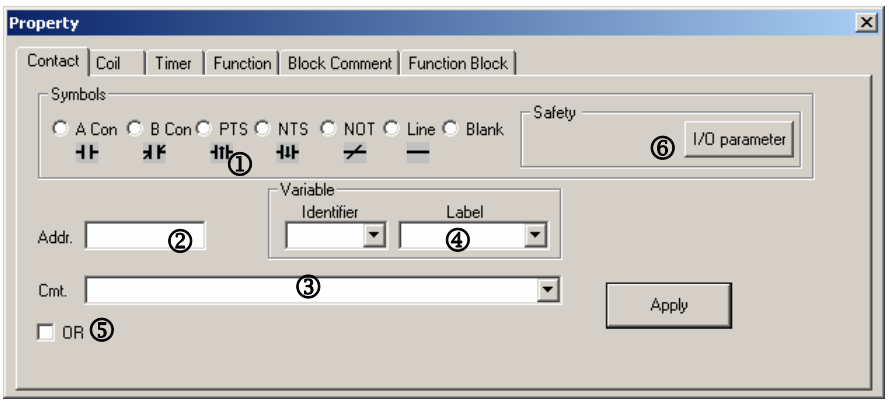


Figure 7-33 Property dialogue for contacts

① Type of contact.
② Entry field for address.
③ Entry field for comment (max. 100 characters).
④ Entry fields for identifier and label (equipment label).
⑤ Simultaneous insertion of OR operation of a contact with the block above the contact at the current position.
⑥ Opens the I/O parameter dialogue (see Chapter 7.9.4)

Coil

A coil corresponds to the writing to an address. The address must be entered hexadecimally as a bit address. In the case of addresses from the word memory area (D, N, S) the corresponding bit position is selected within the address with a suffixed '-', whereby the '□' stands for the corresponding BIT within the WORD.

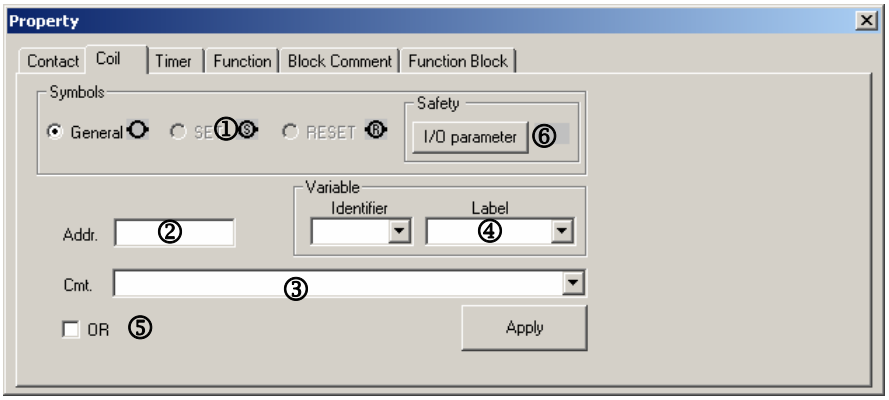


Figure 7-34 Property dialogue for coils

① Selection of the coil.
② Entry field for address.
③ Entry field for comment (max. 100 characters).
④ Entry fields for identifier and label (equipment label).
⑤ Simultaneous insertion of OR operation and of a coil with the block above the coil at the current position.
⑥ Opens the I/O parameter dialogue (see Chapter 7.9.4).

Timer

Only one address from the T-memory area (PS:T000…T0FF, PN:T000..T1FF) can be assigned to a timer. The address must be entered hexadecimally as a bit address.

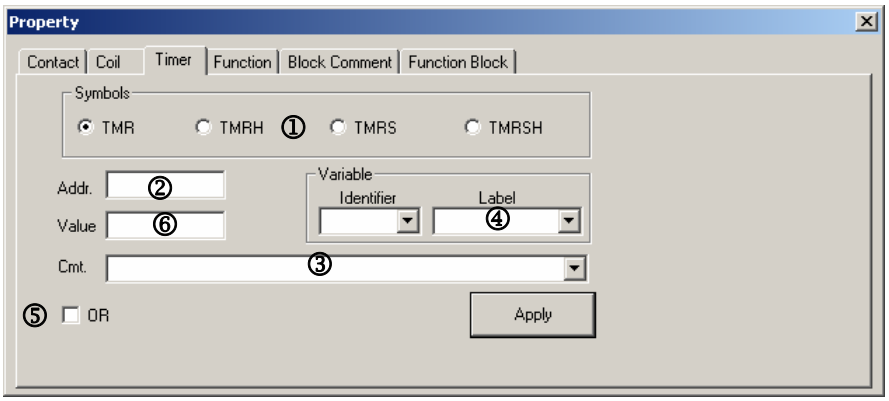


Figure 7-35 Property dialog for timers

① Type of timer.
② Entry field for address.
③ Entry field for comment (max. 100 characters).
④ Entry fields for identifier and label (equipment label).
⑤ Simultaneous insertion of an OR operation and of a timer.
⑥ Entry field for time value.

Counter (PN program only)

Only one address from the C memory area (000..1FF) can be assigned to a counter. The address must be entered as a bit address hexadecimally.

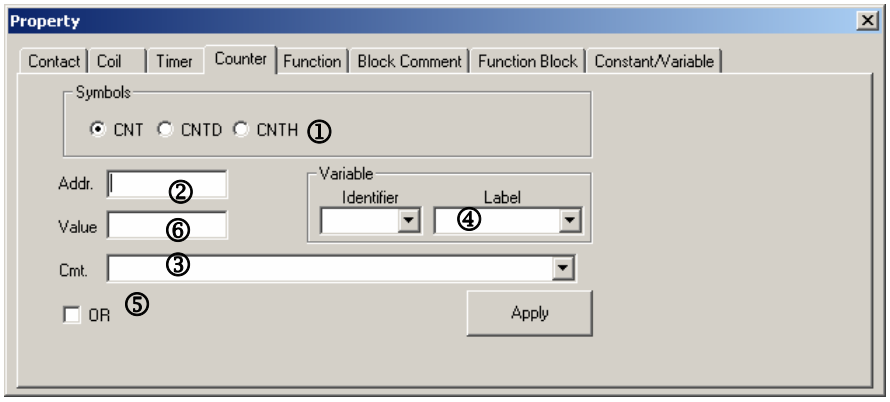


Figure 7-36 Counter property dialogue

① Type of counter.
② Entry field for address.
③ Entry field for comment (max. 100 characters).
④ Entry fields for identifier and label (equipment label).
⑤ Simultaneous insertion of an OR operation and of the counter with the block above the counter to the current position.
⑥ Entry field for time value.

Function

7.10.7 / 7-38

The selection possibilities are limited here to the required function via function name. For functional reasons, entries of a comment or activating the [OR] option do not have any effects here (see Chapter 7.10.7).

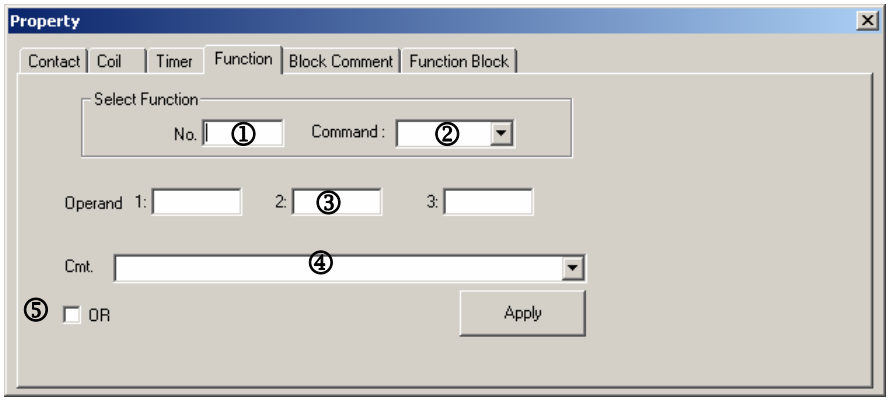


Figure 7-37 Function properties dialogue

① Display of the function number.
② Selection of the function type via the function name.
③ Operand
④ Entry field for comment (max. 100 characters).
⑤ Simultaneous entry of an OR operation and of a counter.

Block comment

Entry of the block comment.

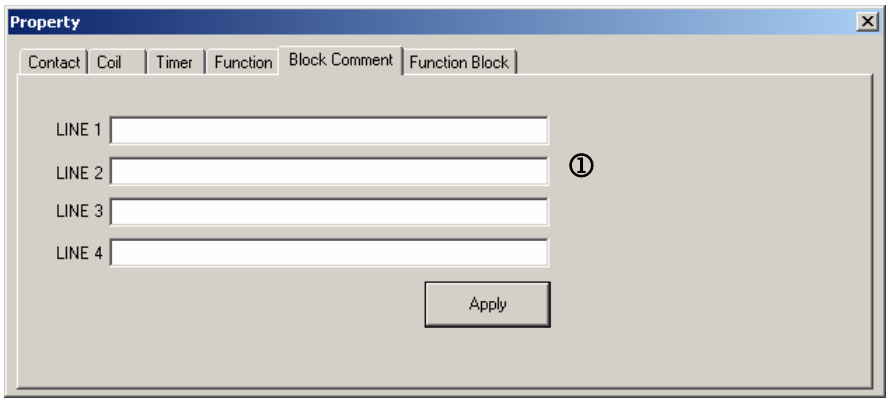


Figure 7-38 Property dialogue for block comments

① Possibility of entering four comments, each with up to 50 characters

Constants / variables (PN program only)

Definition of constants / variables in the PN program.

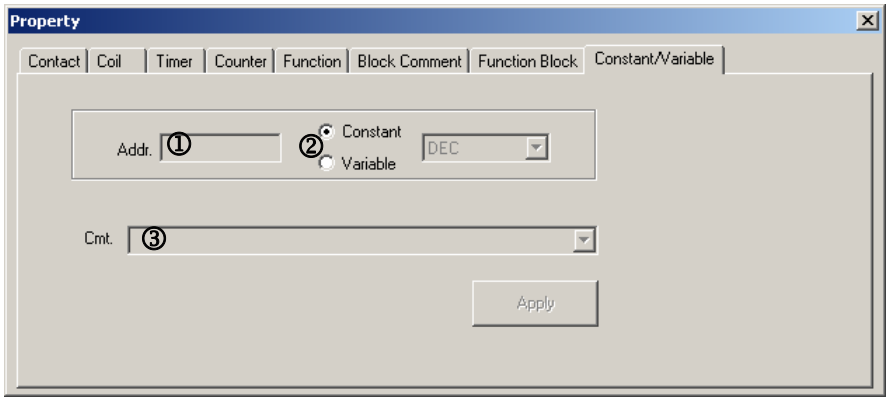


Figure 7-39 Property dialogue for constants / variables

① Determination of a hexadecimal constant / variable (will depend on ①)
② Selection of constant or variable. An element of the memory area must be stated in the case of a variable.
③ Entry field for comment (max. 100 characters).

7.11 Program examples

For better clarity, the functions ‘START’, ‘END’ and ‘PEND’ are not shown in the program examples. However, they are always part of the ladder diagram. In a few examples only excerpts from the overall programs are shown. The missing blocks are either responsible for a different functionality of the user program or have already been described in a different program example. The program examples refer to the corresponding circuitry examples in Chapter 4.3. The reference to the corresponding circuitry example is shown in the left margin.

4.3 / 4-11 

7.11.1 Manual resetting (Reset, acknowledgement)

If switching-off has been triggered by a safety function, it may be necessary, depending on the evaluation of the risk involved, to reset manually to cancel the shutdown of the outputs.

Manual resetting

- must be done through a separate input at the **SFL**.
- may not trigger any dangerous movement itself. For this purpose, it is necessary to provide the information on the manual resetting of the operational control.
- must be a manual and conscious action for example, the person must be able to see the danger area and he may actuate the button for manual resetting only after checking that there is no one in the dangerous area at that moment.
- may only occur through the trailing edge of a previously activated contact maker.

In the following you will find a program example to manually reset. If necessary, other details may be taken from the relevant B and C standards.

Example

Reset with edge detection through a physical input.

Description

Resetting only occurs after the button has been released.

Safety categorisation

May be used up to a maximum category 2 / PLc in compliance with DIN EN ISO 13849-1.

Program example:

Figure 7-40 Manual resetting

7.11.2 START level

The start is typically managed by the operational control. However, it is also possible to have the START signal managed by the **SFL**.

- The start or restart may only be possible if all safety devices are active and, if necessary, the manual resetting has been put into effect.
- The start signal may not be stored.

Program example:

Figure 7-41 Start level – Start button

7.11.3 EMERGENCY-OFF (EMERGENCY-STOP) circuitry

Ladder diagram

4.3.2 / 4-12

It starts with a monitored button (down edge).
The botton is pressed for 2 seconds and released.

Safety enable
Feed back signal (back check) must be and connection with a start button.

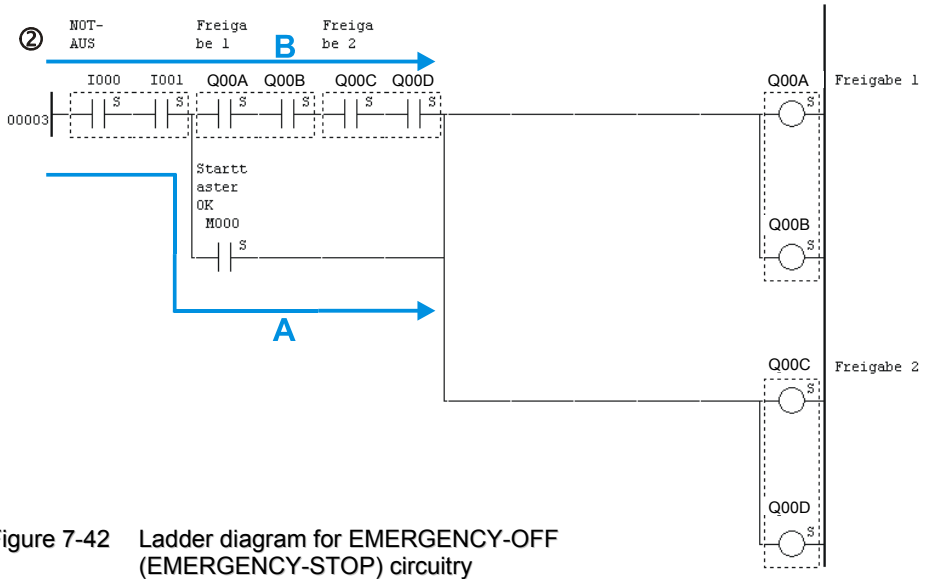


Figure 7-42 Ladder diagram for EMERGENCY-OFF (EMERGENCY-STOP) circuitry

Description

Start / Reset level ①

The start level has a ‘monitored start’. This means that a start is first produced after the release of the start button with trailing edge. This prevents unintentional restart in the event of an error (e.g. short circuit) in the start level.

There is an additional hardware connection in series from the feedback circuit (auxiliary contact) of the actuators to the start button. This measure prevents start-up if at least one actuator is operated.

Sensor / actuator level ②

The contacts Q00A to Q00D (= state of enables) are switched in series to the EMERGENCY-OFF (EMERGENCY STOP) sensor (I000/I001). If the start button is correctly actuated (M000 = high(1)) and the EMERGENCY-OFF (EMERGENCY-STOP) circuit is closed, the enables (Q00A to Q00D) are activated via a connecting branch A. The start condition is no longer given in the program run due to P000 = low(0). Connecting branch A is thus interrupted. The connection is now made via the closed branch B (as long as the EMERGENCY-OFF (EMERGENCY-STOP) sensor is closed). The EMERGENCY-OFF (EMERGENCY-STOP) ciruit is now open, and enables (Q00A to Q00D) are deactivated. This state remains in existence until the EMERGENCY-OFF (EMERGENCY-STOP) circuit is closed again and the start button is correctly confirmed. The above described process starts again from the beginning.

7.11.4 Guard monitoring

Ladder diagram

4.3.3 / 4-13

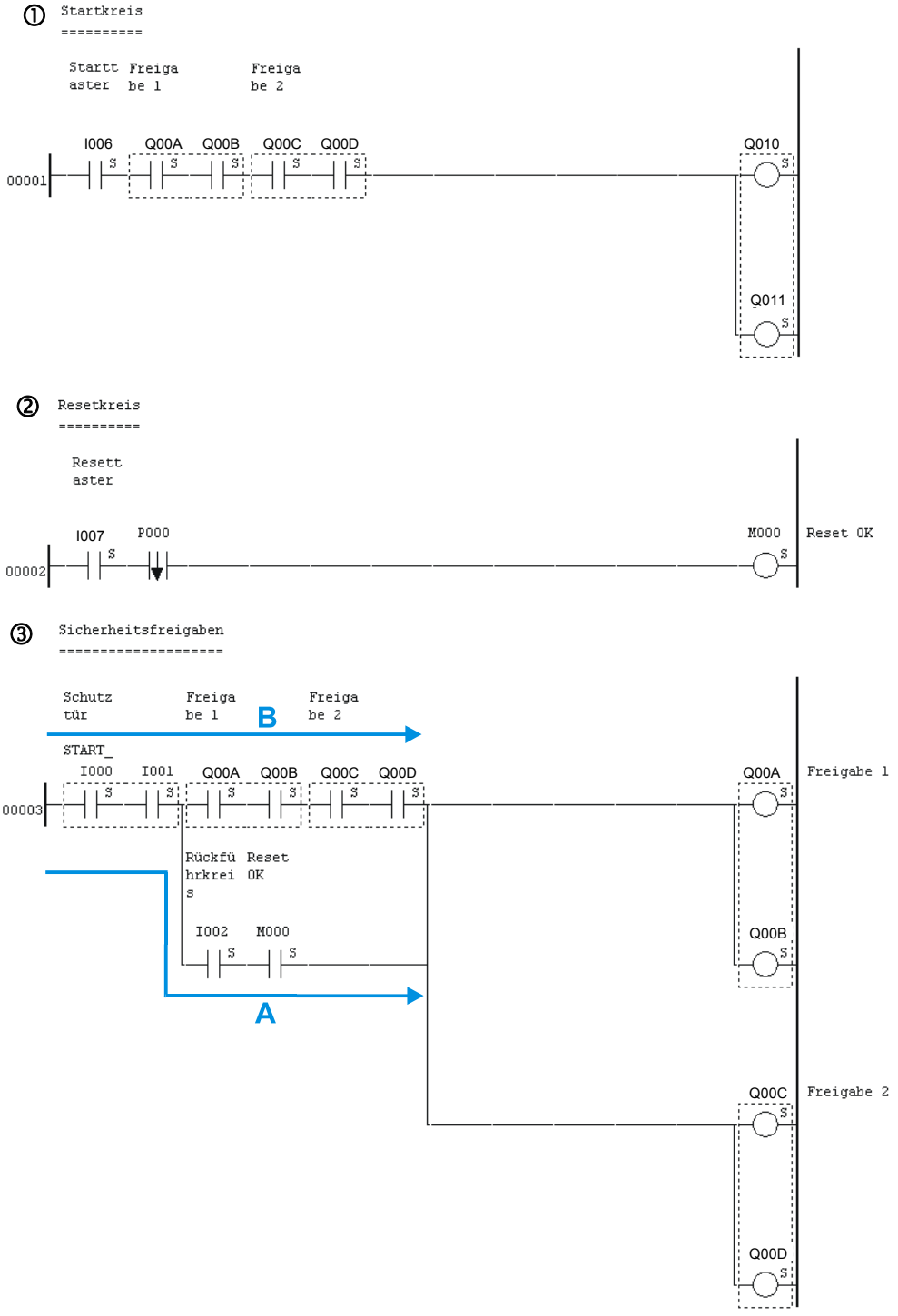


Figure 7-43 Ladder diagram - guard monitoring

Description

Start level ①

With the start button the standard requirement is communicated to the operational control (regulator enable) through the outputs (Q010/Q011) after manual resetting (Q00A to Q00D closed).

Reset level ②

The reset does not occur until the reset button has been released with the trailing edge. This prevents unintentional restart when an error occurs (e.g. short circuit) in the reset level.

Sensor / actuator level ③

The reset is realised here in series with the incorporated hardware feedback circuit (I002). If the guard (I000/I001) and feedback circuit are closed and the reset button (M000) is activated, the enables (Q00A to Q00D) are activated via a connecting branch A. As soon as an actuator has finished its mechanical switching procedure, the start condition no longer exists due to the open feedback circuit (I002). Connecting branch A is thus interrupted. The connection is now made via the closed branch B (as long as the guard is closed). If the guard is opened, the enables (Q00A to Q00D) are de-activated. This state remains in existence until the guard and the feedback circuit are closed again. The process described above starts again from the beginning.

7.11.5 Guard monitoring with locking

Door opener

Ladder diagram

4.3.4 / 4-14

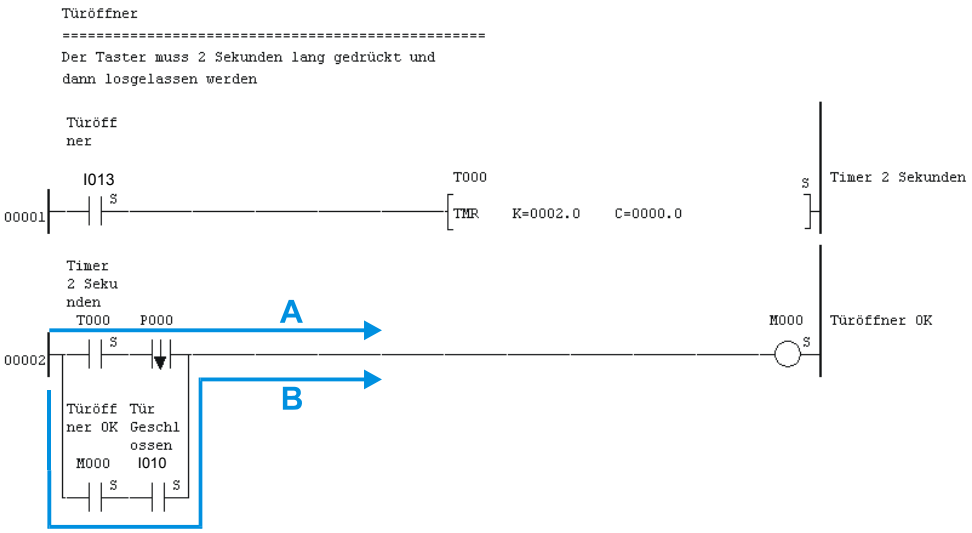


Figure 7-44 Ladder diagram – guard with locking (door opener)

Description

7.11.3 / 7-54

The interrogation of the door opener (I013) is realised here in connection with a timer (T000). The door opener must be held for at least 2 seconds and then released. The signal to open the door (M000) is activated through the connection branch A. Self-holding, which remains active until the guard is opened, is effected through connection branch B.

This type of self-hold must be used for all coils/timers set permanently with dynamic input signal because for safety reasons the ladder diagram symbol ‘coil self-holding’ (Flip-Flop) is not implemented.

Door interlocking

Ladder diagram

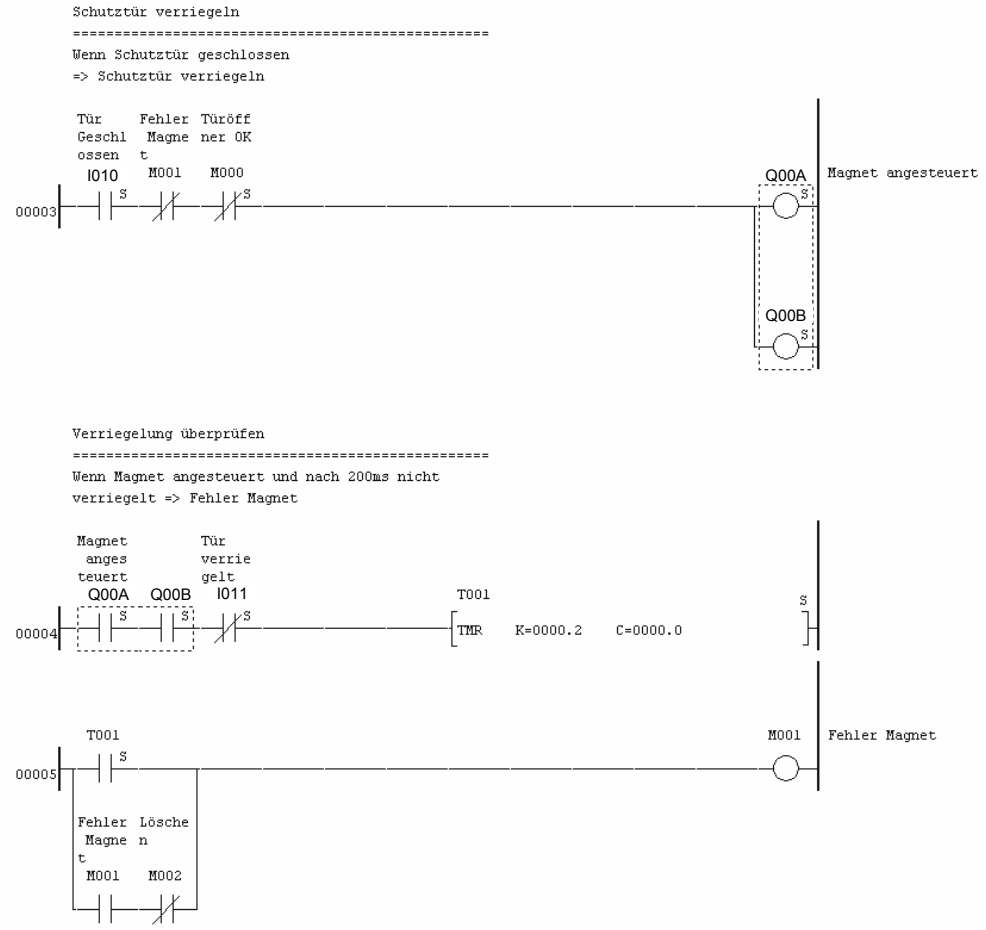


Figure 7-45 Ladder diagram – guard with locking (interlock)

Description

In the case of a closed guard (I010) the interlocking solenoid (Q00A/Q00B) is activated if no error of the solenoid (M001) has been detected beforehand and no request to open the guard has been recognised. After a waiting period of 200ms for the mechanical actuation realised by timer T0001, the guard must be interlocked (contact I011 = true(1)). If this is not the case, T001 becomes true(1) and thus also flag M001 which then goes into self-holding. The self-hold of M001 can be cancelled by flag M002.

Enable

Ladder diagram

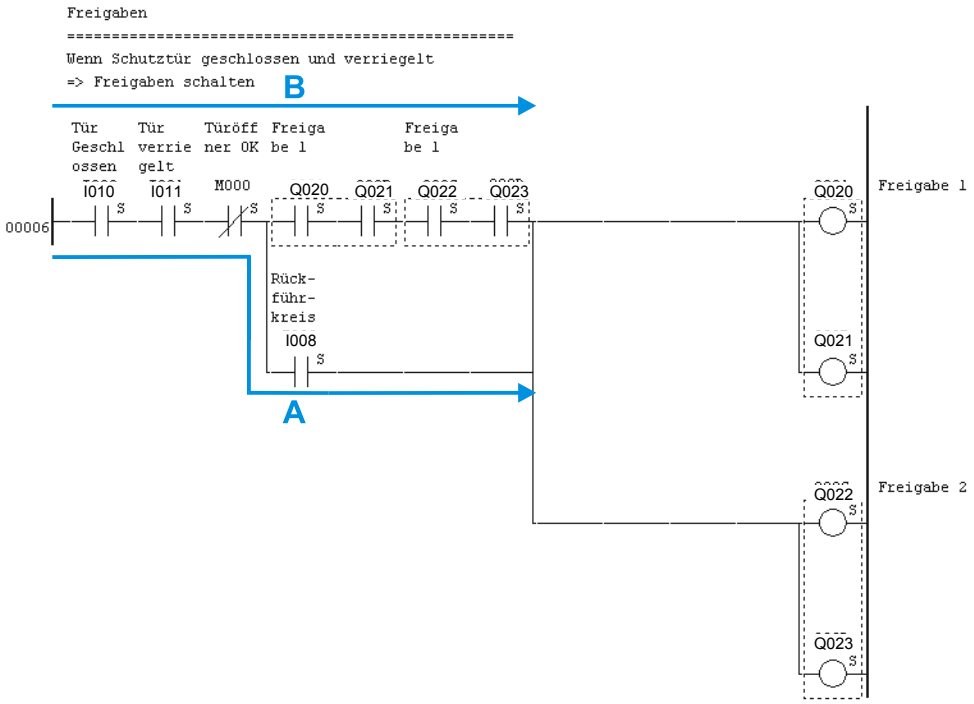


Figure 7-46 Ladder diagram – guard with locking (enable)

Description

In the case of a closed (I010) and locked (I011) guard and non-actuated door opener (M000) as well as closed feedback circuit (I008) the enables (Q020 to Q023) are activated via connection branch A. As soon as an actuator has ended its mechanical switching process, this connecting branch no longer exists due to the open feedback circuit (I008). Connecting branch A is thus interrupted. The connection is now made via the closed branch B. If the guard is now opened and a request to open the door (M000) is detected, connection branch B is interrupted and the enables (Q020 to Q023) are deactivated. The interlocking solenoid (Q00A/Q00B) is not triggered any longer either. This state remains in existence until the guard is closed again and locked and the feedback circuit is closed. The process described above starts again from the beginning.

7.11.6 Safety solenoid-operated switch

7.11.4 / 7-56 The ladder diagram and the description are provided in Chapter 7.11.4.

7.11.7 Current-sourcing semi-conductor

7.11.4 / 7-56 The ladder diagram and the description are similarly provided in Chapter 7.11.4 (but without start and reset circuit). Only the addresses for sensor and feedback circuit need to be adjusted.

7.11.8 Start-up testing

Ladder diagram

4.3.3 / 4-13

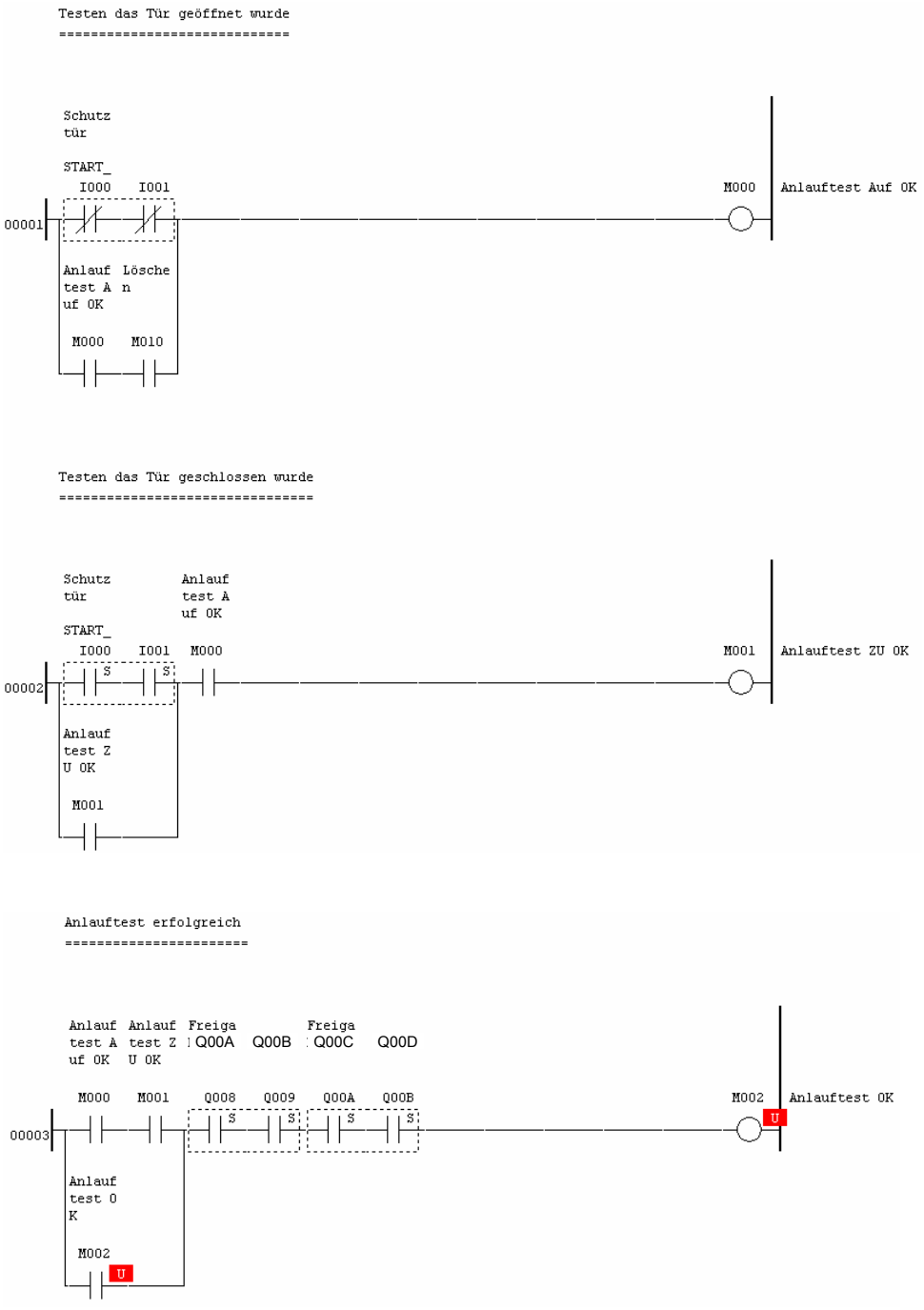


Figure 7-47 Ladder diagram – start-up testing

Description

The test is conducted in block 00001 as to whether both inputs of the door interlock have opened. If the test is positive, then and only then, the test is performed as to whether both contacts have closed again (block 00002). Only if both tests have been concluded positively in exactly this order is flag M002 set as start condition (block 00003). Flag M010 must be set. After the successful start-up (not shown here) flag M010 and thus the self-hold of M000 is deleted.

7.12 Library / Function block

It is possible to use the library function of **SFL-SOFT** without Sentinel hardware key (Dongle).

7.12.1 Description

Library

A library contains one or more function blocks. In order to be able to use a library's function blocks, the library must first have been integrated into the project. Only one library may be integrated into a project. The maximum size of a library for the PS program amounts to 16k WORD, for the PN program to 32k WORD.

Function block

Function blocks (FB) are program sections which, like a subprogram, can be called up by the user program. They offer the possibility of first using created program sections in any user programs. An FB must possess at least one input and one output. An FB is created by means of the ladder diagram programming language in the same way as the user program.

7.12.2 Types of function blocks (FB)

This chapter describes the function blocks which are available with **SFL-SOFT**.

PS program

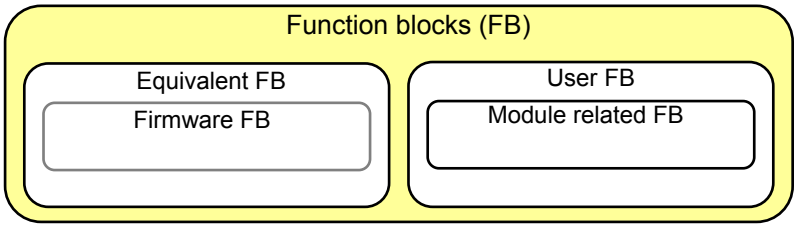


Figure 7-50 FB types for use in the PS program

PN program

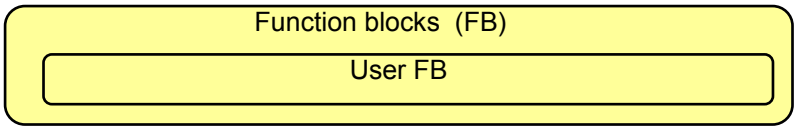


Figure 7-51 FB types for use in the PN program

Function blocks

Firmware FB

FB's for Mode 1 which are realised in the firmware of the CPU module. **SFL-SOFT** has no access to these FB's.

Equivalent FB

Ladder diagrams of the firmware FB's. These ladder diagrams are stored in the firmware of the CPU module and can be uploaded by **SFL-SOFT** when the CPU module is in Mode 1.

User FB

FB which has been generated by the user with **SFL-SOFT**.

The symbolic illustration in the project window is made with the colour violet.

Module related FB

FB which has been generated by the user with **SFL-SOFT** and which is to be used with a special module. This FB can only be integrated in the PS program if the corresponding module appears at least once in the configuration. See Chapter 7.12.4.

7.12.4 / 7-66

The symbolic illustration in the project window is made with the colour blue.

7.12.3 Creating a library

To create a new library, please proceed as follows:

1. Start **SFL-SOFT** and then close the project window.
2. Select [Library] - [New] - [PS Library (S)] in the main menu. The following dialogue then appears.

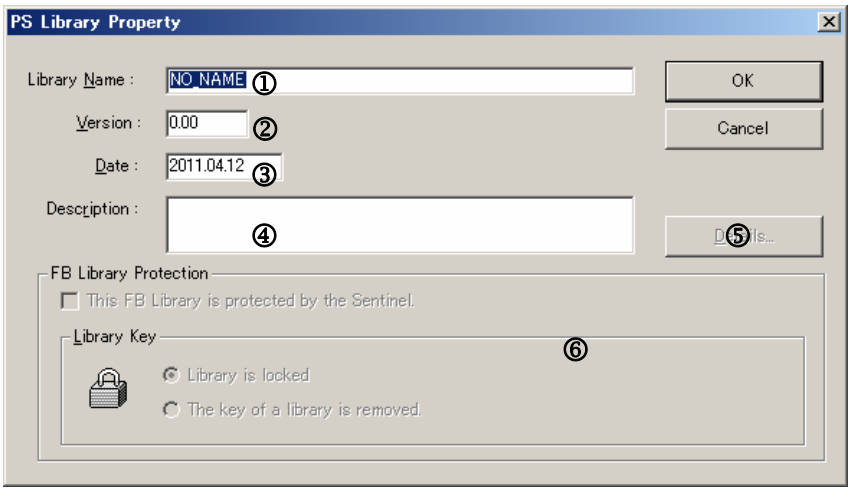


Figure 7-52 Property dialogue FB library

① Name of the library (maximum 50 alphanumeric characters incl. '_').
② Version number (0.00…99.99).
③ Date of creation (yyyy.mm.dd).
④ Description (maximum 100 characters).
⑤ Not used.
⑥ Not used.

After confirming the entered data with the [OK] switch, a new folder is automatically created in the project window (Rider Library). After creation of a New Folder (right mouse button, [New Folder]) it is possible to add FB (right mouse button, [Add FB]). After having added the first FB, by clicking the right mouse button, you can reach a pop-up menu with the entries shown below.

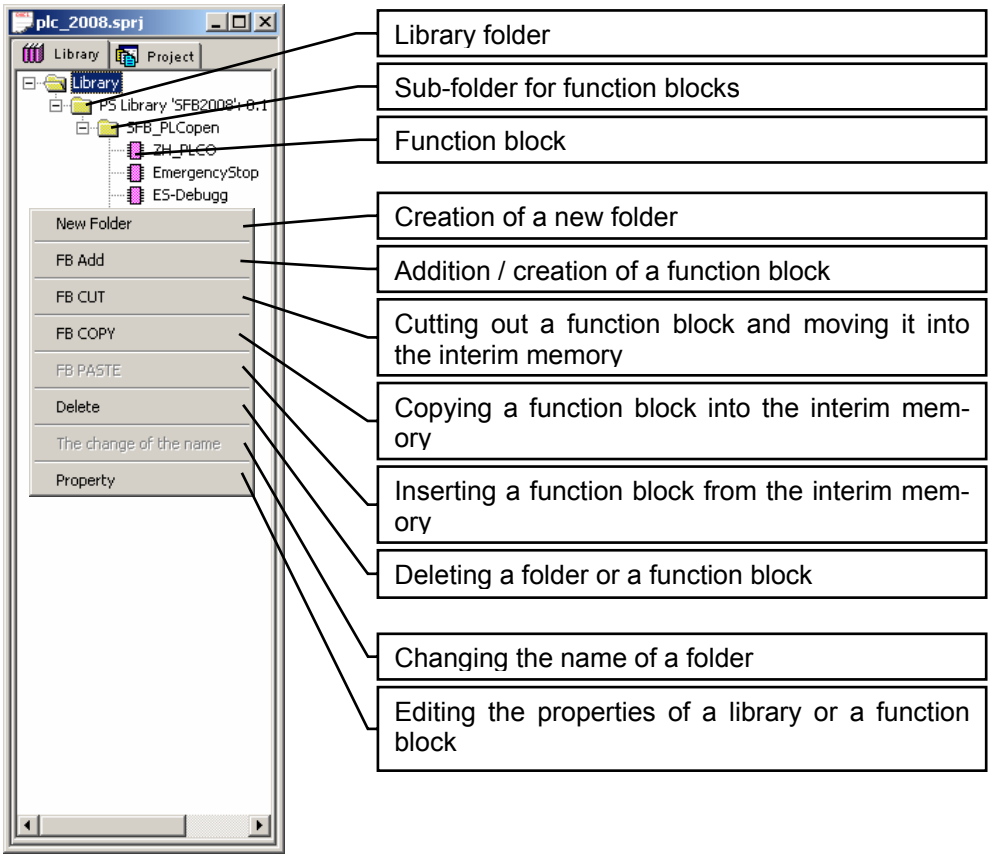


Figure 7-53 Project window tab library with pop-up menu

7.12.4 Creating a function block

Function blocks can only be created in sub-folders. For this purpose select the entry [New Folder] (see Figure 7-53) under the Library rider from the pop-up menu in the project window.

In this sub-folder you can now add or create a function block by means of the entry [Add FB] in the pop-up menu. During or after creation of a function block the function block can be protected with a password. The protection can only be released by knowing this password.

Then make the settings for the function block to be created.

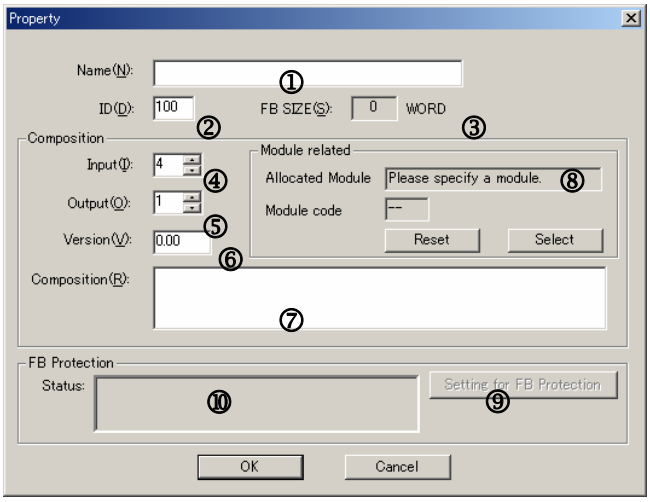


Figure 7-54 Property dialogue function block

① Name of the function block (maximum 36 alphanumeric characters without blanks).
② Clear ID number for each function block (1…1023).
③ Shows the size of the function block.
④ Number of inputs (1…20).
⑤ Number of outputs (1…20).
⑥ Version number (0.00…99.99).
⑦ Description (maximum of 100 characters).
⑧ Module assignment. Indication of the assigned module if the FB is a module related FB. See Chapter 7.12.2.
⑨ FB protection status is displayed
⑩ Setting for FB Protection function.

The total number of inputs and outputs of a function block must be greater than two.

The function block is created as soon as the entered data has been confirmed with the [OK] switch. Once the function block has been created, you can reach the operating mode by opening the created function block.

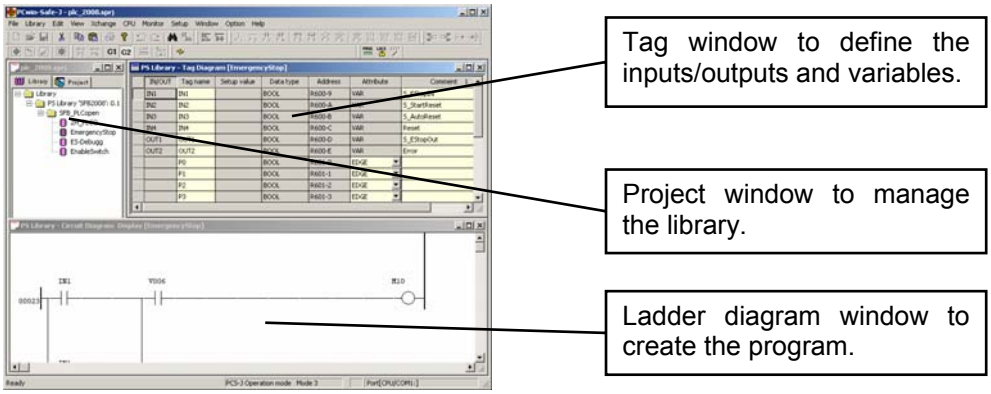


Figure 7-55 Operating mode function blocks

Creating a module-related FB

1. Selecting [Select] in the property dialogue of the function block causes a window to appear with detailed settings (see Figure 7-54).

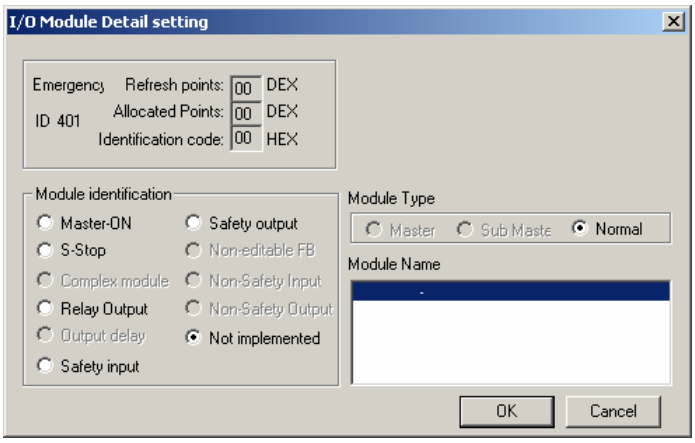


Figure 7-56 Property dialogue of a module-related FB (1/3)

2. Selection of the module

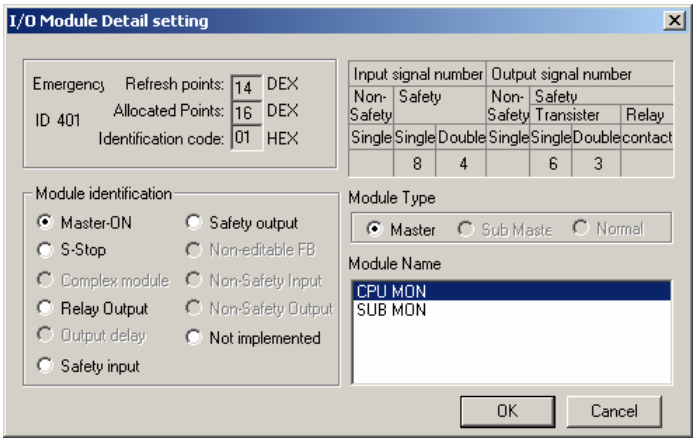


Figure 7-57 Property dialogue of a module-related FB (2/3)

3. Actuating [OK] selects the module without it being possible to use the FB.

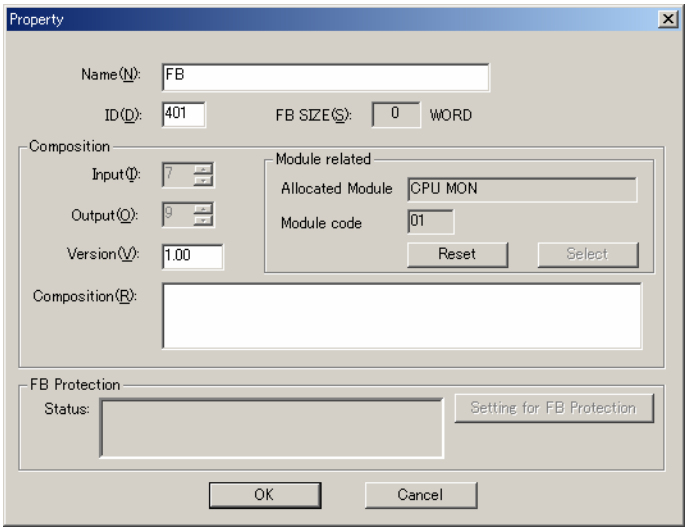


Figure 7-58 Property dialogue of a module-related FB (3/3)

Selecting [Reset] cancels the assignment.

Setting for FB Protection

1. Selecting [Setting for FB Protection] in the property dialogue of the function block causes a window to appear with detailed settings.

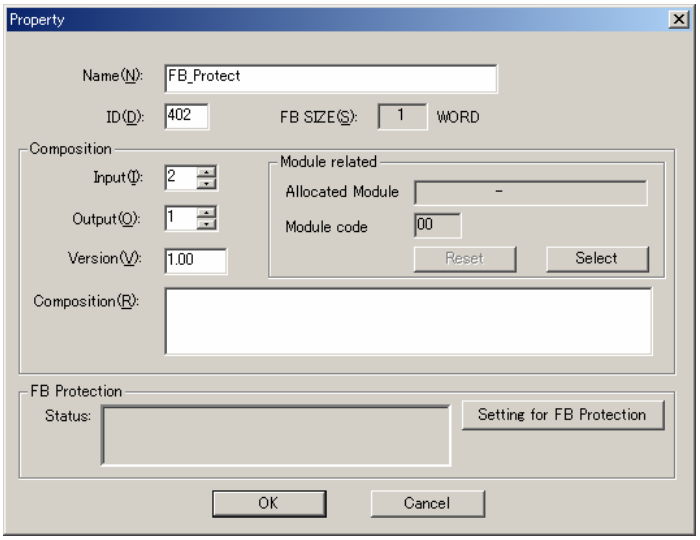


Figure 7-59 Property dialogue of FB Protection (1/5)

2. FB protection setting person's name is inputted.

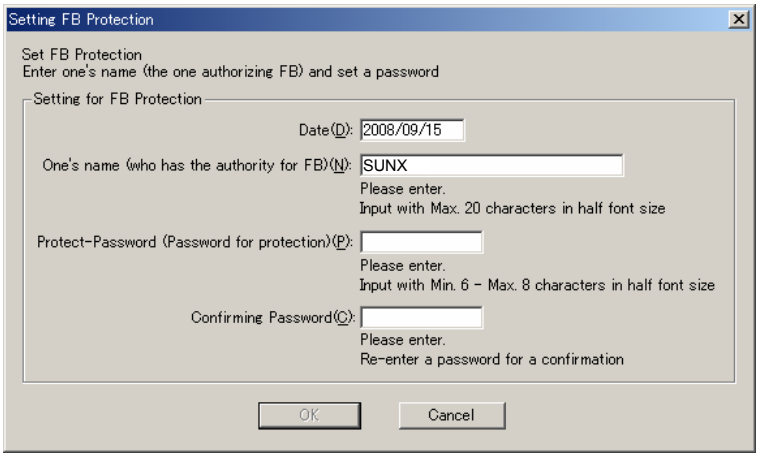


Figure 7-60 Property dialogue of FB Protection (2/5)

3. FB protection password is inputted.

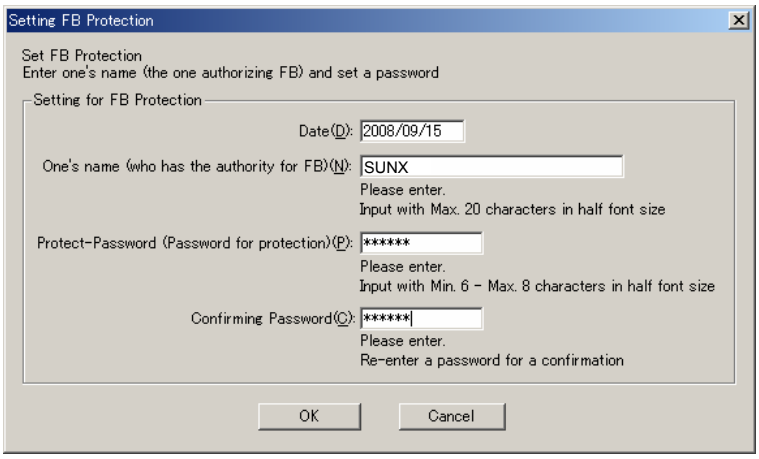


Figure 7-61 Property dialogue of FB Protection (3/5)

4. FB protection status is displayed.

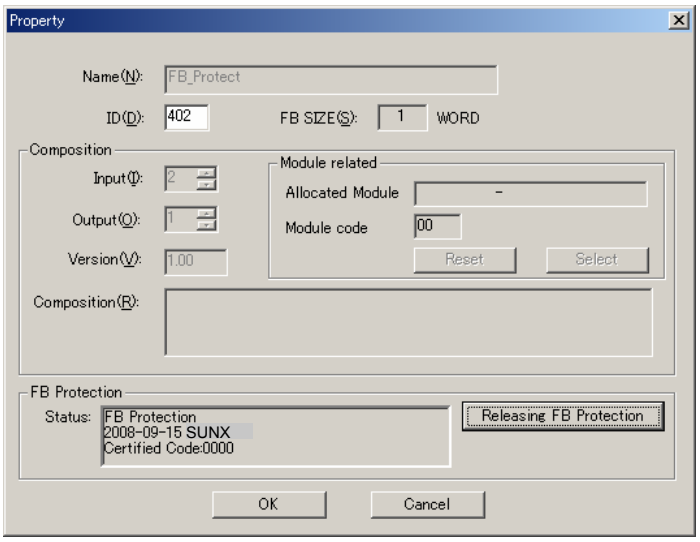


Figure 7-62 Property dialogue of FB Protection (4/5)

5. FB icon under FB protection is displayed.

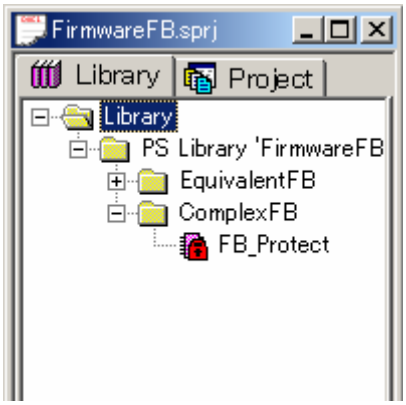


Figure 7-63 Property dialogue of FB Protection (5/5)

Releasing for FB Protection

1. Selecting [Releasing FB Protection] in the property dialogue of the function block causes a window to appear with detailed settings.

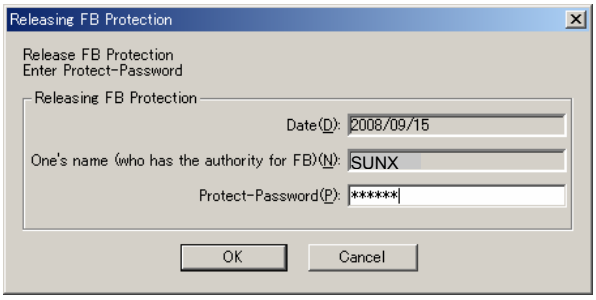


Figure 7-64 Property dialogue of FB Protection (1/1)

Setting for Invisible FB

1. Selecting [Setting for FB Protection] in the property dialogue of the function block causes a window to appear with detailed settings.

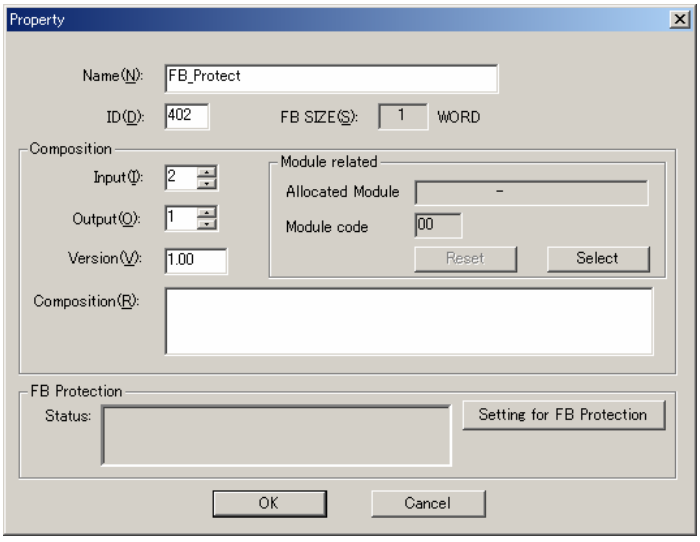


Figure 7-65 Property dialogue of FB Protection (1/5)

2. Invisible FB setting person's name+%UNV is inputted.
   Ex) If person's name is SUNX, please input "SUNX%UNV"

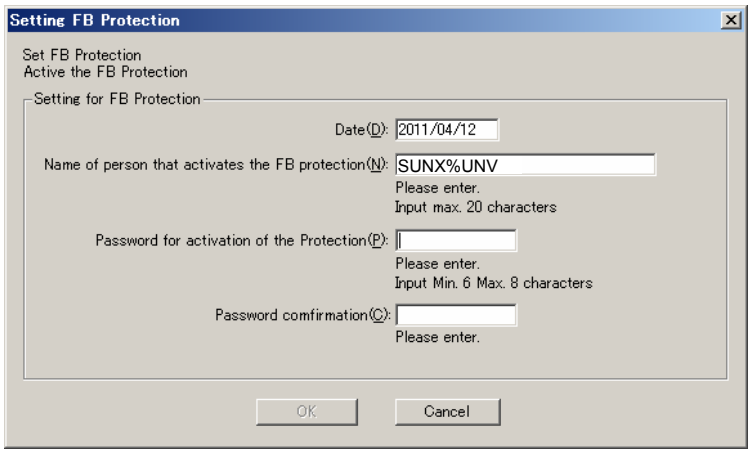


Figure 7-66 Property dialogue of FB Protection (2/5)

3. FB protection password is inputted.

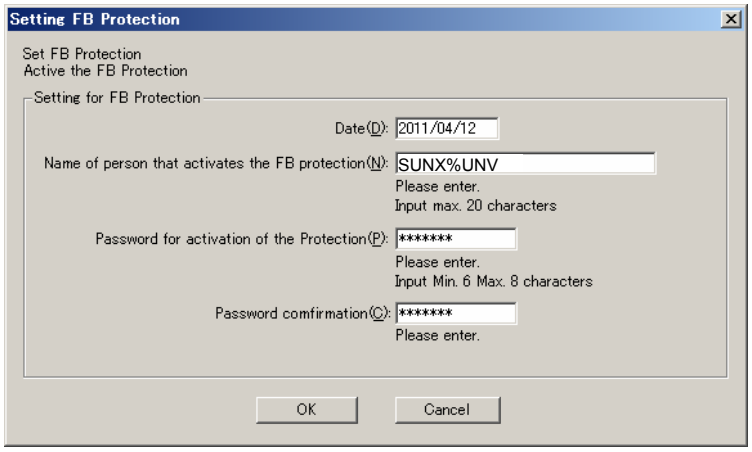


Figure 7-67 Property dialogue of FB Protection (3/5)

4. FB protection status is displayed.

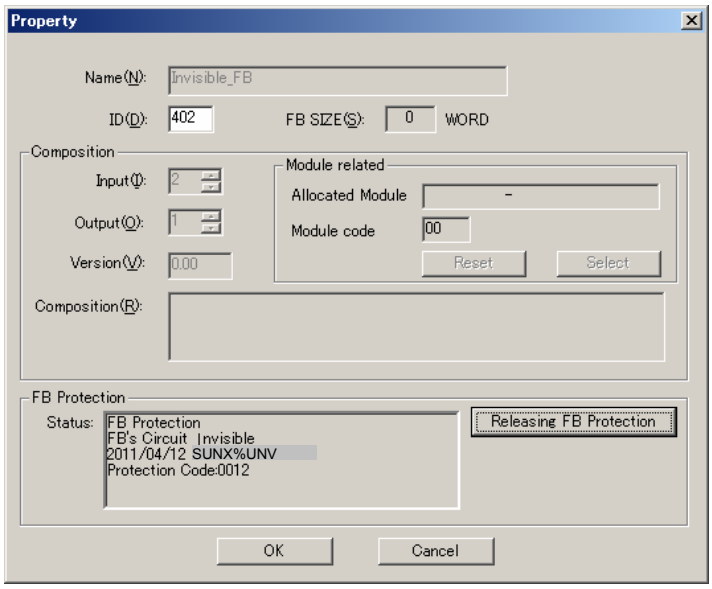


Figure 7-68 Property dialogue of FB Protection (4/5)

5. FB icon under Invisible FB is displayed.

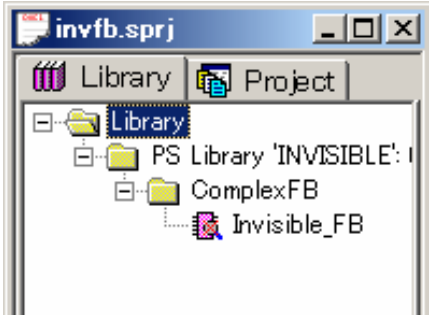


Figure 7-69 Property dialogue of FB Protection (5/5)

Releasing for Invisible FB

1. Selecting [Releasing FB Protection] in the property dialogue of the function block causes a window to appear with detailed settings.

Figure 7-70 Property dialogue of FB Protection (1/1)

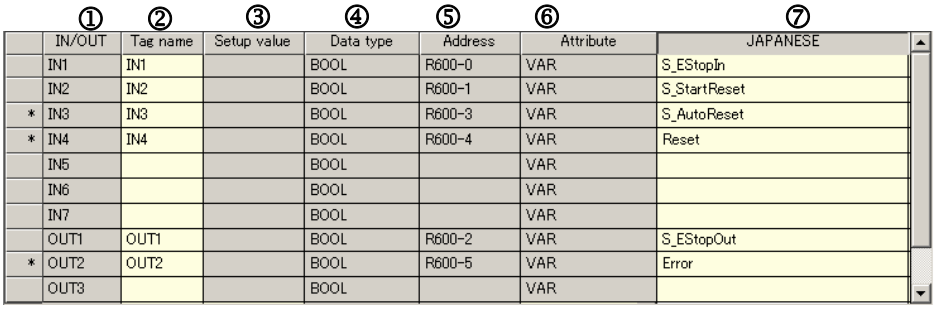
Tag window

Description

Unlike the case when creating the user program, in function blocks you do not work with direct addresses but with so-called tags. The addresses are assigned automatically in a memory area which is not directly accessible to the programmer (Prefix R, area R300 .. R6FF). The entries in the table are marked by different colours, whereby:

Yellow Cells may be edited by the programmer.

Grey Cells are filled in automatically by **SFL-SOFT** and cannot be edited by the programmer.



	① IN/OUT	② Tag name	③ Setup value	④ Data type	⑤ Address	⑥ Attribute	⑦ JAPANESE
	IN1	IN1		BOOL	R600-0	VAR	S_EStopIn
	IN2	IN2		BOOL	R600-1	VAR	S_StartReset
*	IN3	IN3		BOOL	R600-3	VAR	S_AutoReset
*	IN4	IN4		BOOL	R600-4	VAR	Reset
	IN5			BOOL		VAR	
	IN6			BOOL		VAR	
	IN7			BOOL		VAR	
	OUT1	OUT1		BOOL	R600-2	VAR	S_EStopOut
*	OUT2	OUT2		BOOL	R600-5	VAR	Error
	OUT3			BOOL		VAR	

Figure 7-71 Tag window

① IN/OUT
These cells mark the previously defined inputs/outputs.

② Tag name
Clear reference to an address/timer/edge detection contact. Under this name the address/timer/edge detection contact is activated during programming. A tag name consists of a maximum of 9 alphanumeric characters incl. underlining. No differentiation is made between lower case and upper case letters.

③ Setup value
7.10.5 / 7-36
This column can only be edited for the timer attribute. The set time (see Chapter 7.10.5) of the timer can be entered here. A tag name can be entered here too. A modification in this column is only accepted in the ladder diagram after the activation of the switch [Assign] in the property dialogue.

④ Data type
The data type belonging to the tag name.

PS library: only the data types BOOL (e.g. M123, D0123-F) and WORD (only when a timer is used) are available here. The data type WORD is automatically set when a timer is used; a manual selection of the data type WORD is not possible.

PN library: the data types BOOL, WORD and BYTE are available here.

⑤ Address
The address is automatically filled in with the exception of the ‘EXTERN’ attribute.

⑥ **Attribute**
Identifies the type of the tag name.

PS library

TIMER ⇨ Timer
EDGE ⇨ Edge detection
VAR ⇨ Variable inside the function block.
EXTERN ⇨ Address outside the function block.

Attribute	Address area	Assignment	Addressing	
			BOOL	WORD
TIMER	Timer (T000-00F)	automatically	✓	-
EDGE	Variable from the R600-R6FF sector	automatically	✓	-
VAR	Variable from the R600-R6FF sector	automatically	✓	-
CURRENT	Instantaneous value timer	automatically	-	✓
EXTERN	Any address from the V/S memory area	manual	✓	-

Table 7-17 Addressing types, attribute types function block (PS Library)

PN library

TIMER ⇨ Interval Timer / Timer
COUNTER ⇨ Counter
EDGE ⇨ Edge detection
VAR ⇨ Variable inside the function block.
EXTERN ⇨ Address outside the function block.

Attribut	Address area	Assignment	Addressing		
			BOO	BYTE	WORD
TIMER	Timer (T000-00F)	automatically	✓	-	-
COUNTER	Counter (C000-00F)	automatically	✓	-	-
EDGE	Variable from the R600-R6FF	automatically	✓	-	-
VAR	Variable from the R600-R6FF	automatically	✓	✓	✓
CURRENT	Instantaneous value timer Timer (N000-N00F)	automatically	-	-	✓
EXTERN	Any address from the V/S memory area	manuell	✓	✓	✓

Table 7-18 Addressing types, attribute types function block (PN Library)

When using FBs, the timers 000 to 00F in the user program are not available. They are reserved for use in FBs. If an FB is used multiply with timers, the **SFL** ensures that there is no mutual influence.

⑦ **Comment**
Possibility of entering a comment text (max. 100 characters).

The comment language displayed for FB can be changed by language setting. Although language setting can be set as [Library] - [Language Setup] and one of languages must be English.

Adding lines

1. Bring the mouse pointer to the first column (in front of IN/OUT) to the line above which you wish to insert a new line.
2. Press the left mouse button. An insertion character (‘>’) is shown in the first column.
3. Now press the ‘INS’ button to insert a line.

You can insert additional inputs/outputs by executing the above procedure above an IN/OUT line.

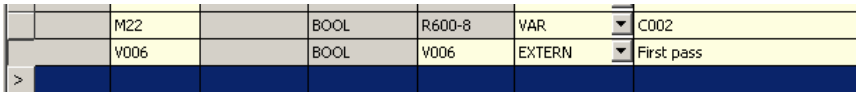


Figure 7-72 Insertion / deletion of a line in the tag table

Deleting lines

1. Bring the mouse pointer to the first column (in front of IN/OUT) to the line you wish to delete.
2. Press the left mouse button. An insertion character (‘>’) is shown in the first column.
3. Now press the ‘DEL’ button to delete the line.

You can delete unnecessary inputs/outputs by executing the above procedure on an IN/OUT line.

Ladder diagram

7.10 / 7-31

A ladder diagram is created in the same way as the creation of a user program described in Chapter 7.10. An exception, however, is addressing, which occurs here symbolically by means of the tag names. Furthermore, to prevent recursive program calls, it is not possible to use any function blocks.

In addition to entering the tag name by means of the keyboard, it is also possible to enter by means of ‘Drag and Drop’. If you press the left mouse button over a highlighted line, you can transfer the contents of the cell into the tag edit field of the property dialogue by dragging the mouse while keeping the left mouse button pressed.

Logical Check (PS Library only)

7.13.2 / 7-84

A logical check of a function block is requested, as in the user program, before saving the library. Further details can be found in Chapter 7.13.2.

7.12.5 Altering / revising

Version management

Each library and each function block has a version number. To make version management easier, **SFL-SOFT** reminds you before every saving process (insofar as alterations have been made) to update the version number (if desired).

Figure 7-73 Dialogue reminder to alter version

① Select [yes] if you have already updated the version number or if you do not wish to alter the version number.
② Select [No] if you have not yet updated the version number and now wish to do so. The property dialogue in the library (Figure 7-52) or function block (Figure 7-54) will then open.

SFL-SOFT does not have any automatic version management. The programmer must decide himself if the alterations he has made necessitate an updating of the version number.

Copying / Moving

Function blocks can be copied / moved through the Windows interim memory. As in a library the name and ID number of a function block must be unequivocal, proceed as follows when copying:

1. Copy a function block (Pop-up Menu [FB copy] in the project window rider library) into the interim memory.
2. Open the property dialogue (Pop-up menu [Properties] in the project window rider library) and change the name and ID number of the function block.
3. Insert the function block from the interim memory (Pop-up-Menu [FB insert] in the project window rider library) into any other folder you wish.

In order to copy/move function blocks into another library, start for each library its own instance of **SFL-SOFT**.

7.12.6 Reading / comparing

Reading

Reading from the file

Use the [Library] - [Open] - [PS library] or [Library] - [Open] - [PN library] from the main menu and then select the library you require from the list.

Reading from the CPU

Use the [Library] - [Read CPU] - [PS library] or [Library] - [Read CPU] - [PN library] from the main menu and wait until the readout is finished.

Comparing

7.14.3 / 7-94

Libraries can be compared with each other like user programs. To do so, select [Edit] - [Block Compare Program] in the editing mode from the main menu. Details can be found in Chapter 7.14.3.

Additionally, you can compare a library with the one saved in the **SFL**. To do so, select [Library] - [Compare with the CPU] - [PS Library] or [Library] - [Compare with the CPU] - [PN Library].

The comparison can only be made if you have selected the option [All programs + Parameter + Library + Comment] when transferring the program (see Chapter 7.13).

7.12.7 Program example

7.11.3 / 7-54 The program example shows the conversion from the sample program from Chapter 7.11.3 into a function block.

Tag window

	IN/OUT	Tagname	Einstellwert	Datentyp	Adresse	Attribute	Germany
*	IN1	start		BOOL	R600-0	VAR	Starttaste (abfallende Flanke)
*	IN2	eingang1		BOOL	R600-1	VAR	Eingang für 2-kanaligen NOT-AUS Sensor
*	OUT1	freigabe1		BOOL	R600-2	VAR	Erste Freigabe
*	OUT2	freigabe2		BOOL	R600-3	VAR	Zweite Freigabe
*		timer	20	BOOL	T000	TIMER	Zeitgeber 2 Sekunden
				WORD	N000	CURRENT	Aktueller Zählerstand
*		flanke		BOOL	R601-0	EDGE	Flankenerkennung (abfallende Flanke)
*		merker0		BOOL	R600-4	VAR	Starttaster OK

Figure 7-74 Tag window sample program function block

Ladder diagram

7.10 / 7-31 The ladder diagram corresponds in its functioning to that from Chapter 7.10. The inputs/outputs are now implemented here as variables by means of tag names.

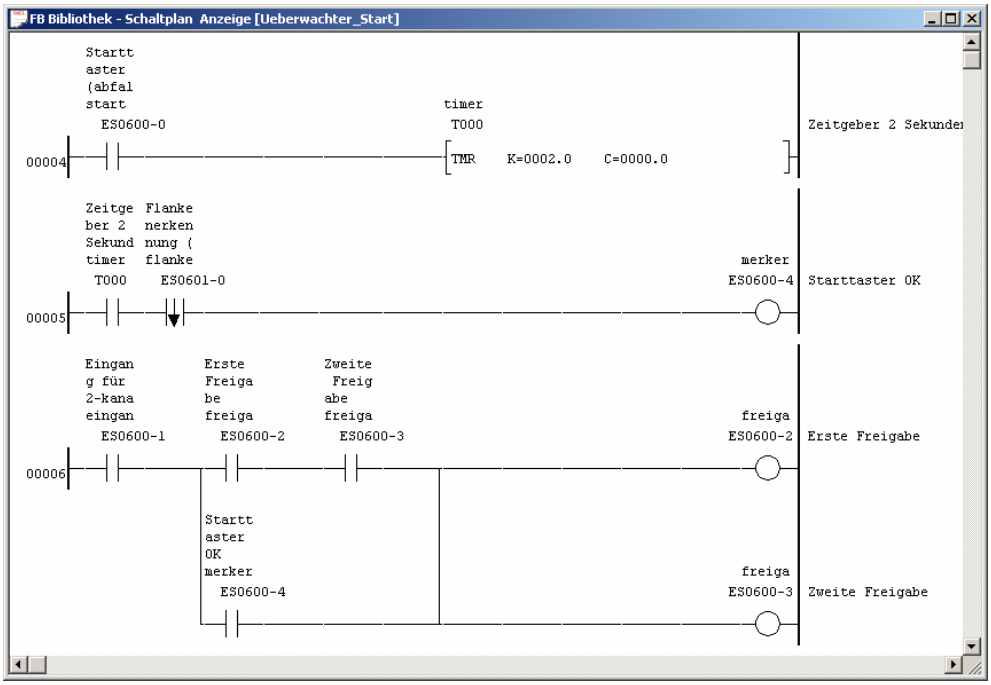


Figure 7-75 Ladder diagram sample program function block

7.12.8 Working with libraries / function blocks

Integrating a library

In order to be able to use function blocks in a user program, you must first integrate the library into the project. For this purpose, the path for the PS and PN library must be indicated under [Option] - [Configuration] in the tab [Library].

Select then in the main menu [File] the entry [Link Library].

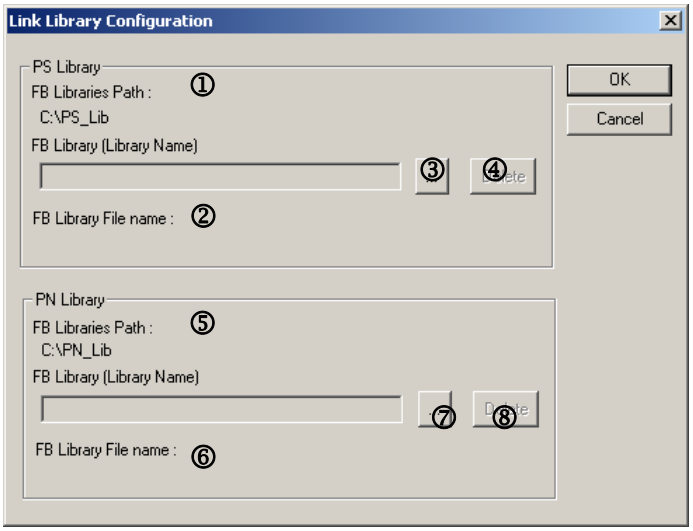


Figure 7-76 Dialogue integrate library

7.17.2 / 7-102

① Set library path (see Chapter 7.17.2).
② Name of the integrated PS library.
③ Selection of a PS library.
④ Separate integrated PS library from the project.

7.17.2 / 7-102

⑤ Set library PN path (see Chapter 7.17.2).
⑥ Name of the integrated PN library.
⑦ Selection of a PN library.
⑧ Separate integrated PN library from the project.

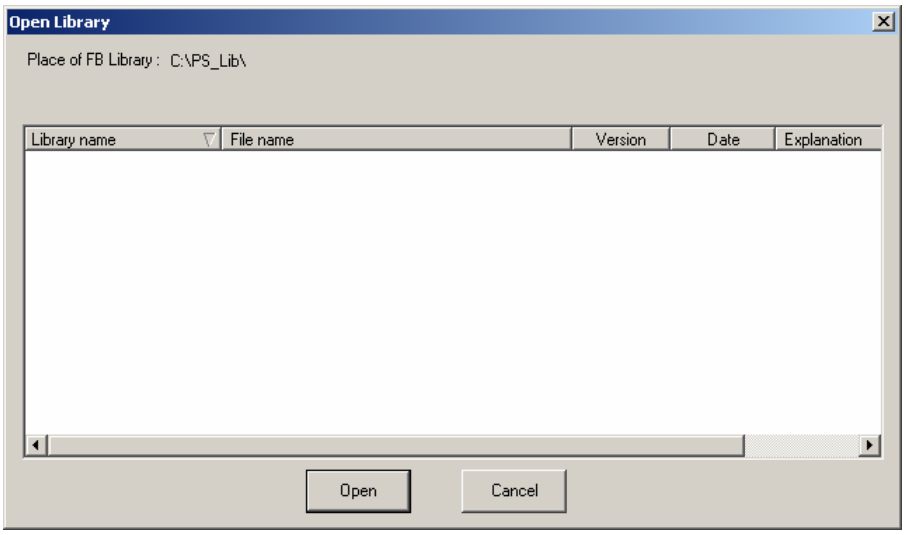


Figure 7-77 Dialogue open library

Once you have selected and integrated a library, **SFL-SOFT** confirms the successful integration of the library and inserts the function call for the library in the ladder diagram before the commencement of the program.

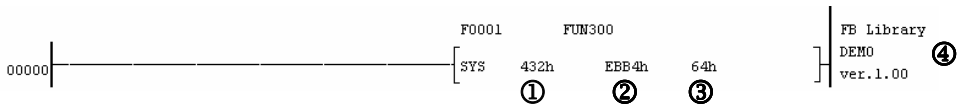


Figure 7-78 Ladder diagram symbol for integrated library

① PS library system function 432h (integrate library).
PN library system function 432h (integrate library).
② ID number 1 of the library.
③ ID number 2 of the library.
④ Name and version number of the library.

Insertion of a function block

A function block can be inserted into the diagram ladder by means of a tool button or the function key [F8].

Selection of a function block

A function block is selected through the property dialogue. The possibilities are limited here to the selection of a function block and the entry of a comment.

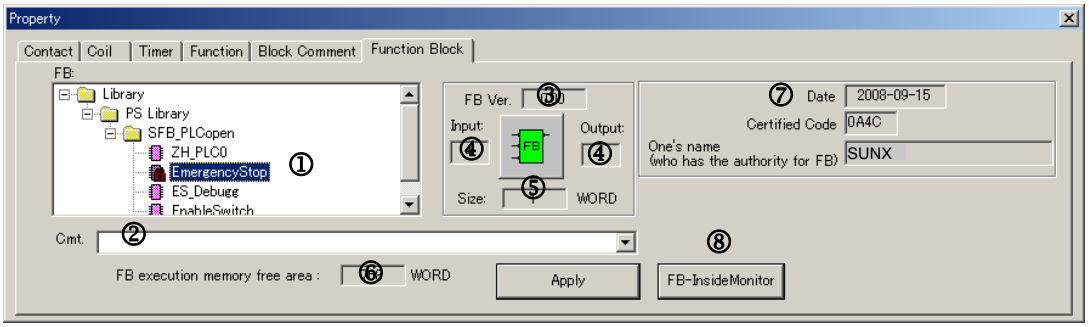


Figure 7-79 Dialog Eigenschaften Funktionsblock

① Selection of the function block.
② Entry field for comment (max. 100 characters).
③ Version number of the function block.
④ Number of the inputs/outputs
⑤ Storage size in WORD.
⑥ Remaining free memory for the use of function blocks within a user program. This number is reduced after every insertion (use) of a function block by its memory capacity.
⑦ Status of FB protection
⑧ A screen is changed to the FB-Inside monitor

Defragment FB execution memory

Due to the deletion of a function block from the ladder diagram, memory gaps can emerge in the FB execution memory. This can have as a consequence that no further function blocks can be inserted into the ladder diagram, although there is still free memory available. The free memory blocks are not large enough then to include the structure of the function block. With the menu item [Xchange] - [Defragment FB execution memory], you can reorganise (defragment) the FB execution memory.

7.13 Checking

7.13.1 Checking LD program

If the ladder diagram is opened you can check the ladder diagram by selecting [Edit] - [LD Program Check] from the main menu. The check is made automatically before every manual (tool button  or menu [Xchange] - [LD Edited Blocks] or automatic (save/transfer) conversion. The errors are subdivided into five groups according to different criteria.

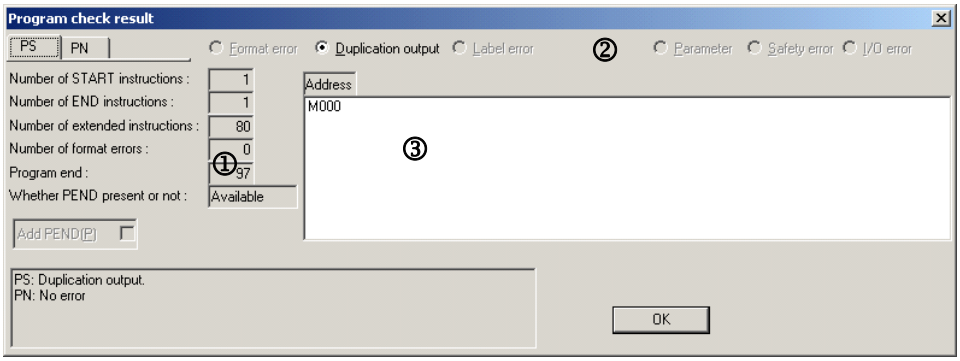


Figure 7-83 Dialogue LD Program check

① Additional information.
② Selection of the error group (only available if error exists).
③ Error messages within selected group. Double clicking on the error message will bring you automatically to the defective block.

Error groups

Format errors

The errors shown below permit a display of the ladder diagram despite existing errors.

Error code	Description
04	False address area for contact for edge detection.
05	Double use of a ‘P’ address
06	Error in the conversion of ‘single’ to ‘double’ contact/coil
0C	Undefined command in program

Table 7-19 Error messages – format error group

Format errors (continued)

The errors shown below do not permit a display of the ladder diagram. Instead of the corresponding block only the error message with the error code is displayed in the ladder diagram.

Error code	Description
01	Display limit exceeded (max. 22 lines per block)
02	Stack overflow with STR, STR NOT statement
03	OR statement without STR statement
04	AND statement without STR statement
05	Stack overflow with FPS statement
06	FRD statement without FPS statement
07	FPP statement without FPS statement
08	No value at stack (S0), AND STR not possible
09	No value at stack (S0), OR STR not possible
0A	Timer statement stack error
0B	Program limit exceeded (block with more than 256 steps)
0E	Bit sample error in timer
0F	No value at stack (S0), FPP not possible
10	Error in stack processing
11	OR statement in timer block.
12	OR,OR NOT,OR STR statements directly after FPS statement
13	STR,STR NOT,OR,OR NOT,OR STR,AND STR statements after OUT statement
14	No OUT statement exists in block

Table 7-20 Error messages – format error group (LD display not possible)

Duplicated output

Error code	Error message
-	Duplicated output address (coil/coil).
-	Duplicated output address (coil/function).
-	Duplicated function number

Table 7-21 Error messages – duplicated output group

Label

Error code	Error message
-	START statement multiply present
-	END statement multiply present
-	START statement present without END statement
-	END statement present without START statement
-	No START statement present
-	No END statement present
-	Neither START nor END statement present

Table 7-22 Error messages – label group

Parameter

Error code	Error message
-	Coil with address of an output module not used for 2 channels
-	The number of assigned channels exceeds the maximum
-	The number of assigned inputs exceeds the maximum
-	The number of assigned outputs exceeds the maximum

Table 7-23 Error messages – parameters group

Safety

Error code	Error message
-	Station number used not present.

Table 7-24 Error messages – safety group

I/O

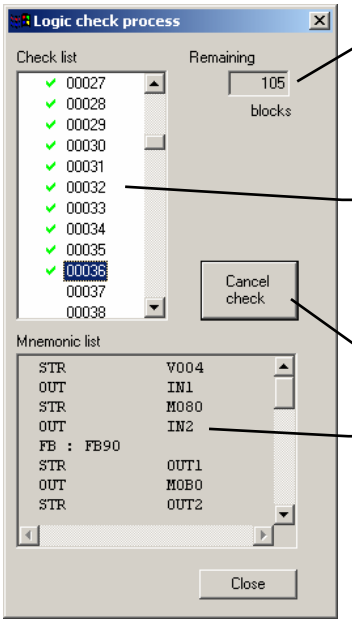
Error code	Error message
-	Adresse eines nicht installierten Moduls wurde verwendet.
-	I(X) wird für eine Ausgangs Spule verwendet

Table 7-25 Error messages – I/O group

7.13.2 Logic check

The request for a logic check is made before every saving / transfer of the PS program (if modifications have been made). Whilst the check during saving can be postponed to a later date, the check before every transfer of the program (if modifications have been made) to the **SFL** is absolutely essential.

It serves to verify the ladder diagram by a different mode of presentation. For this purpose the ladder diagram is translated into a statement list. The programmer must now confirm that every block of the ladder diagram agrees with the corresponding statement list.



Shows the number of remaining blocks to be checked.

Blocks of the ladder diagram (those ticked have already been confirmed). Double clicking on a block number with the left mouse button moves you to the appropriate block in the ladder diagram (and vice versa).

Confirmation / revocation of the checking of a block.

The selected block as statement list.

Figure 7-84 Logic check dialogue

Mode of operation of the translation process

In order to perform a check of the block of the ladder diagram shown as a statement list, it is necessary to understand the principal procedure of **SFL-SOFT** during translation.

For the conversion of a ladder diagram into a statement list it is necessary to imagine a virtual 1-register machine with a register width of 1 bit. In addition to the working register A, this machine has an interim memory (stack S0…S23) to save 24 1-bit values. The working register always contains the result of the last operation (Current Result (CR) according to IEC 61131). The interim memory is organised as a stack. The last stored value is always located in S0. Every statement from table 7.10.4 can be specified as an instruction sequence for this virtual machine. Timer, functions and edge detecting contacts are viewed as function call.

Working register	Interim memory (stack)										
A	S0	S1	S2	S3	S4	S5…S18	S19	S20	S21	S22	S23

Table 7-26 Memory of the virtual machine

Translation tables

Our assumed virtual machine has only a very limited instruction set. In order to distinguish between the instructions in the statement list, the instructions of the virtual machine are shown by an operand in parentheses. The 1 bit operand is shown as 'v' for value.

Command	Function
load(w)	Loads w into A
push	Copies A to stack and increases stack address
pop	Loads value of stack into A and lowers stack address
read	Loads value of stack into A without altering address
out(w)	Assigns w the value of A
not	Inverts the value of A
and(w)	Performs AND logic operation with A and w and saves result in A
nand(w)	Performs NAND logic operation with A and w and saves result in A
or(w)	Performs OR logic operation with A and w and saves result in A
nor(w)	Performs NOR logic operation with A and w and saves result in A
xor(w)	Performs XOR logic operation with A and w and saves result in A
nts(w)	A = 1, with change from 0 to 1 of w; otherwise A=0
pts(w)	A = 1, with change from 1 to 0 of w; otherwise A=0
andpop	Performs AND logic operation of A and S0, saves result in A and lowers stack address
orpop	Performs OR logic operation of A and S0, saves result in A and lowers stack address

Table 7-27 Instructions of virtual machine

Every instruction in Table 7-2 can now be specified by an equivalent instruction sequence of the virtual machine.

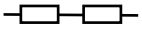
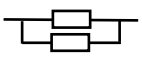
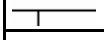
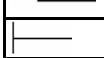
Ladder diagram	Instruction list	Virtual machine
	STR v	push; load(v)
	STR NOT v	push; load(v); not
	AND v	and(v)
	AND NOT v	nand(v)
	OR v	or(v)
	OR NOT v	nor(v)
	AND STR	andpop
	OR STR	orpop
	OUT v	out(v)
	PTS v	pts(v)
	NTS v	nts(v)
	FPS	push
	FRD	read
	FPP	pop
	FST	load(1)
	NOT	not
		no operation

Table 7-28 Translation table of statement list / virtual machine

Translation example

Using a block ladder diagram the following is an example of a translation of the ladder diagram to instruction list. This example merely serves to describe the translation process and is not conceived as an example of use.

Ladder diagram

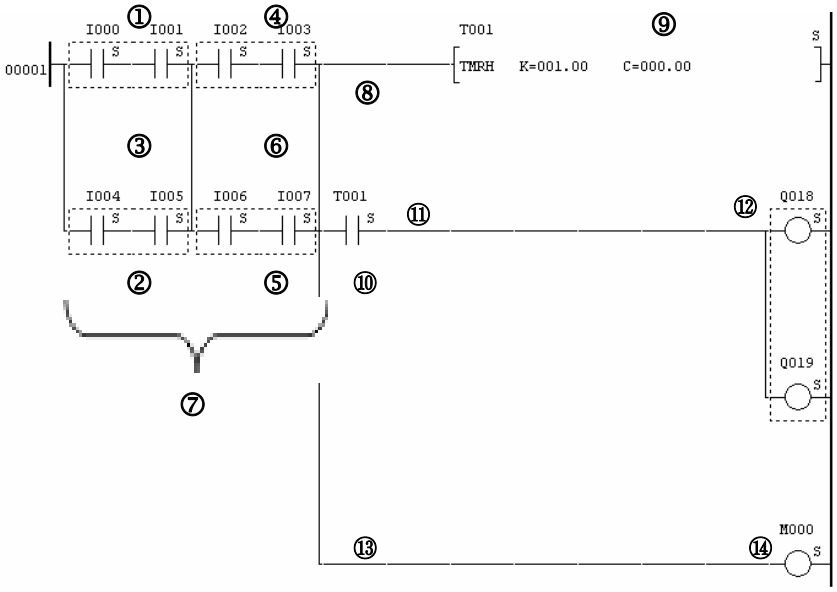


Figure 7-85 Ladder diagram – example of translation process LD -> SDL

Statement list

The statement list is now created step by step from the ladder diagram. The translation process is worked through from top to bottom and from left to right according to the priorities of Booles algebra. This always produces a clear statement list for every ladder diagram block. The following initial states are assumed:

A = 0; I000 to I007 = 1; T001 = 1

The registers (A, S0…S3) show states after execution of the instruction in the 'virtual' column.

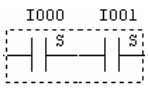
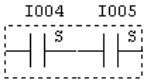
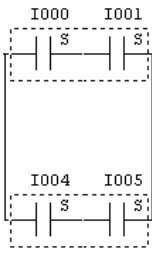
Step	LD	STL	virtual	A	S0	S1	S2	S3
⓪				0	-	-	-	-
①		STR I000	push	0	0	-	-	-
			load(I000)	I000	0	-	-	-
		AND I001	and(I001)	1	0	-	-	-
②		STR I004	push	1	1	0	-	-
			load(I004)	I004	1	0	-	-
		AND I005	and(I005)	1	1	0	-	-
③		OR STR	orpop	1	0	-	-	-

Table 7-29 Example of translation LD -> STL (1/2)

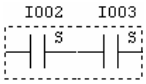
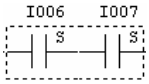
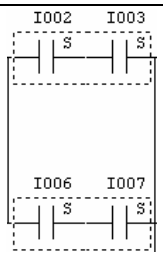
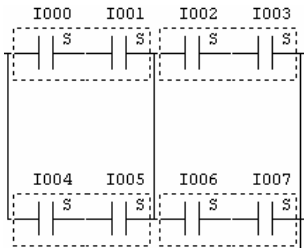
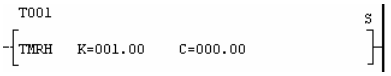
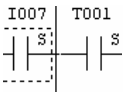
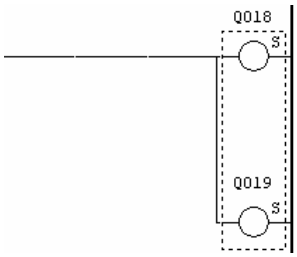
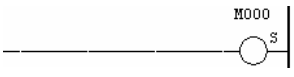
Step	LD	STL	virtual	A	S0	S1	S2	S3
④		STR I002	Push	1	1	0	-	-
			load(I002)	I002	1	0	-	-
		AND I003	and(I003)	1	1	0	-	-
⑤		STR I006	Push	1	1	1	0	-
			load(I006)	I000	1	1	0	-
		AND I007	and(I007)	1	1	1	0	-
⑥		OR STR	orpop	1	1	0	-	-
⑦		AND STR	andpop	1	0	-	-	-
⑧		FPS	push	1	1	0	-	-
⑨		TMRH T001 K=001.00	T001 = 1, wenn Eingang 1,0 Sekunden 1					
⑩		FRD	read	1	1	0	-	-
⑪		AND T001	and(T001)	1	1	0	-	-
⑫		OUT Q018	out(Q018)	1	1	0	-	-
		OUT Q019	out(Q019)					
⑬		FPP	Pop	1	0	-	-	-
⑭		OUT M000	out(M000)	1	0	-	-	-

Table 7-30 Example of translation LS -> STL (2/2)

Logic check of a function block

The logic check of a function block is limited to the logic check of the elements of the LD at its inputs and outputs and the transer to/from the function block. The logic check of the function block is carried out after its setting / modification.

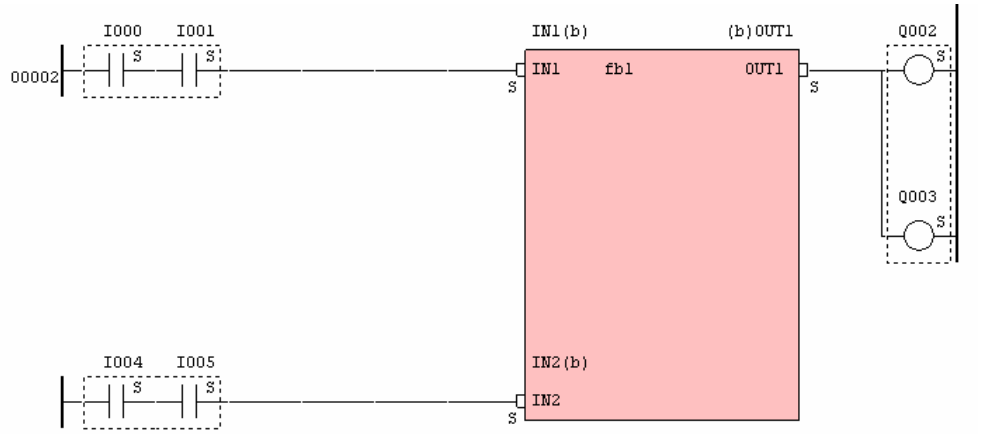
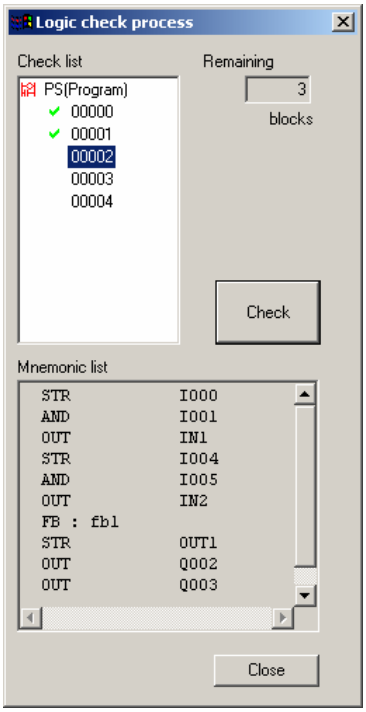


Figure 7-86 Presentation of a function block

The inputs of the function block are called IN1, IN2. The output of the function block is called OUT1. The function block itself bears the designation fb1. This function block is presented in the dialogue logic check as a statement list as described in the following.



The result of rounding off I000/I001 is transferred to IN1 of the function block:

```
STR   I000
AND   I001
OUT   IN1       ; Transfer to the function block
```

The result of rounding off I002/I003 is transferred to IN2 of the function block:

```
STR   I004
AND   I005
OUT   IN2       ; Transfer to the function block
```

The call up of the function block then follows. To check, its name is displayed too.

```
FB : fb1
```

After the call up, the output of the function block OUT1 is read out and edited further.

```
STR   OUT1      ; reading back from the function
OUT   Q002      ; block
OUT   Q003
```

Figure 7-87 Dialogue logic check with function block

7.13.3 Program size and execution time

The PS program may only have a max. size (32000 bytes) and a run time (1.2ms). The PN program may only have a max. size (31488 bytes) and a run time (1.2ms). You can perform a check by selecting [Xchange] - [Program Size Check] in the main menu. The check is performed automatically in the background before transfer to **SFL**.

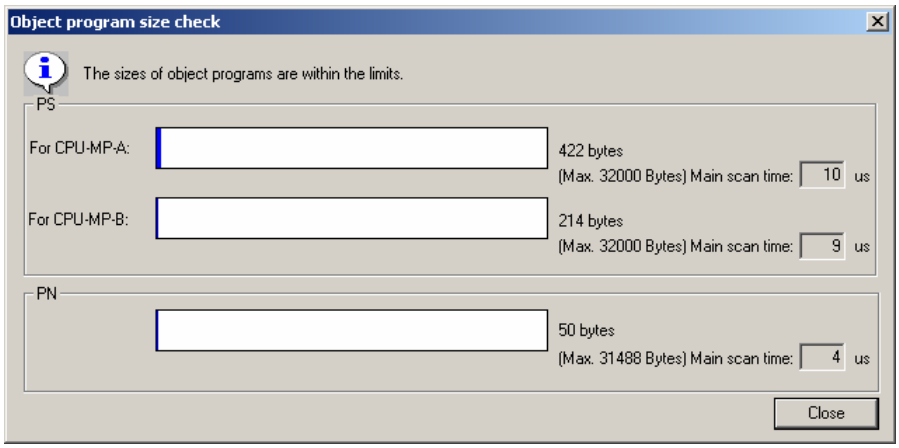


Figure 7-88 Program size check dialogue

7.14 Changing / revising

You have the possibility of performing different changes for the entire user program within the framework of the editing options.

7.14.1 Changing contacts

I/O addresses

This dialogue can be reached via the [Edit] - [Contact Change] - [I/O Address] entry in the main menu.

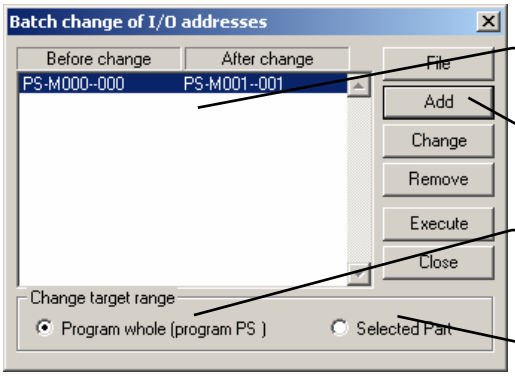


Figure 7-89 Change of I/O addresses dialogue

[File]

Loads a statement list from a CSV-file (comma separated value). The file must have the ending ‘csv’. You can create such a file using Excel or a text editor. If you use Excel to create the file, make sure that a comma is set as separating symbol.

Header

The text ‘#CHANGEALL’ must be entered in the first line of the file.

I/O addresses

The area with the addresses to be modified is introduced with ‘BEGIN_IOBIT’ and ended with ‘#END’. In between are the new address and the address to be modified separated by a comma ‘M000, M010’. A comment on the documentation can be attached similarly separated by a comma.

Function operand

The area with the function operands to be modified is introduced with ‘#BEGIN_FUNCOP’ and ended with ‘#END’. In between are the function operands to be modified and the new function operands separated by a comma ‘D0000L, D0001L’. A comment on the documentation can be attached similarly separated by a comma.

Example

```
#CHANGEALL
#BEGIN_IOBIT
M000, M010, changes address M000 in M010
M001, M011, changes address M001 in M011
#END
#BEGIN_FUNCOP
D0000, D0001, changes address D0000 in D0001
#END
```

[Add]

Adds a new entry of the addresses to be altered to the change list. The addresses must be entered without area specifier hexadecimally as bit address. In the case of addresses from the WORD memory area (D, N, S) a bit number must be specified by '-*'.

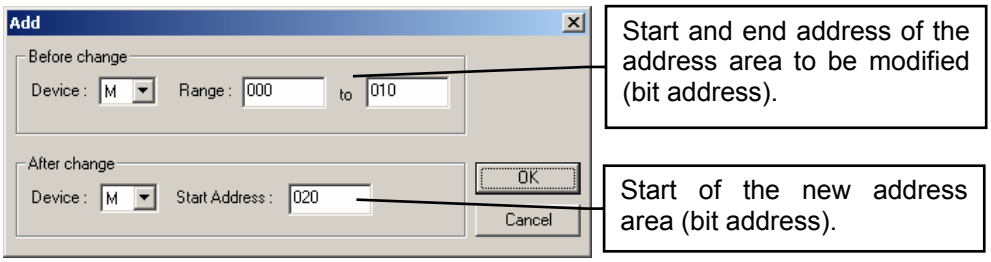


Figure 7-90 Sub dialogue [Change] [Batch Change of the I/O addresses]

[Remove]

Removes the selected entry from the change list.

[Execute]

Performs the changes for all selected entries of the change list.

[Close]

Closes the dialogue.

Function Instruction Operand (PN program only)

You can reach this dialogue in the main menu under the entry [Edit] - [Contact Change] - [Function Instruction Operand]. The procedure is analagous to that to modify I/O addresses. Only the type of addressing is as byte or word address.

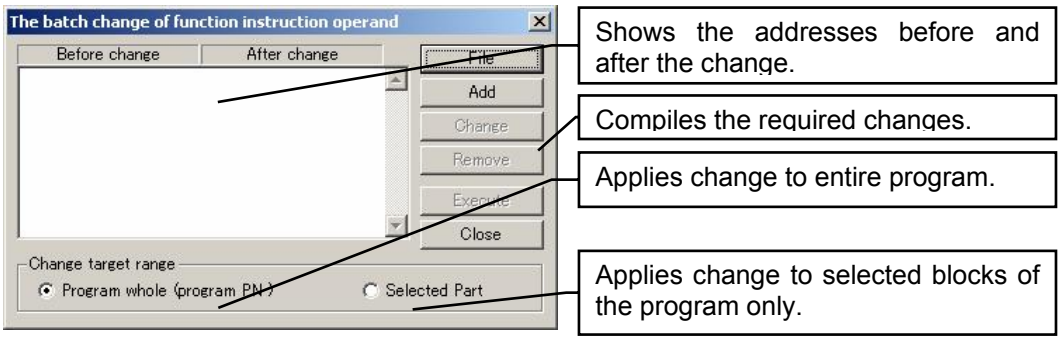


Figure 7-91 Change function operand dialogue

NC/NO contact

This dialogue can be reached via [Edit] - [Contact Change] - [AB Contact] in the main menu. You can change an NC contact into an NO contact by entering a bit address with area specifier and vice versa. This function can only be used for individual contacts.

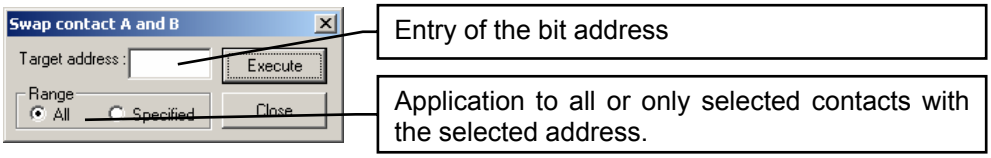


Figure 7-92 Swap contact A and B dialogue

7.14.2 Changing numbering

If the numerical order has been altered by various processes (delete/insert) it is possible to restore ascending numbering through entry in the main menu [Edit] - [Renumbering] - [Function Instruction Serial No.] and [Edit] - [Renumbering] - [Edge Detection Address]. Please bear in mind that these functions cannot be reversed.

7.14.3 Comparing programs

It is possible to compare two programs by selecting the menu point [Edit] - [Block Compare Program]. The comparison is always made block by block. The result of the comparison is shown by a tabular comparison of the individual blocks. The differences are marked by different colour symbols. The opened project is always the starting point. Double clicking on an entry or pressing the 'Block display' button moves you directly to the respective block in the ladder diagram.

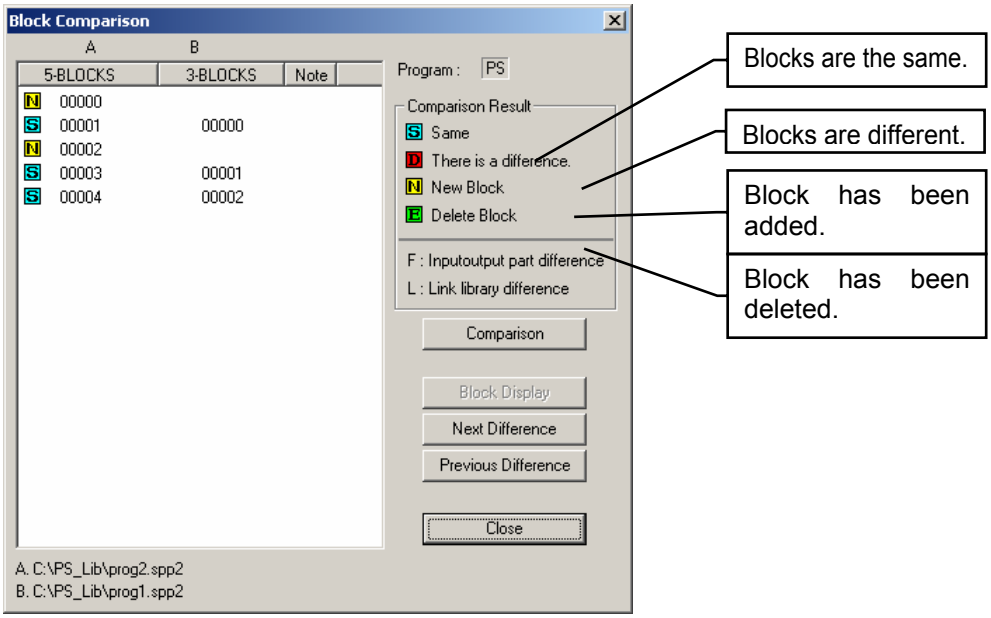


Figure 7-93 Program block comparison dialogue

7.15 Monitoring

For troubleshooting purposes and to study sequences of a user program you can monitor the current state of the inputs, outputs and registers during operation.

7.15.1 Ladder diagram

With an open ladder diagram window you can start or stop the graphic observation in the main menu under [Monitor] - [Start CPU Monitor].

The inputs and outputs are shown in different colours depending on their logical state. The colours are freely adjustable in the dialogue [Adjust] (which can be reached by the main menu [Options] - [Adjust]) under the [Colours] rider.

In addition to the states of the contacts, the time which has currently passed is specified under the function 'timer'.

Depending on requirements the interrogation/updating time of the monitoring can be set in the dialogue [Configuration] (which can be reached by the main menu [Options] - [Configuration]) under the instructions rider in a range of 100ms to 5000ms.

7.15.2 Memory area

The memory area is monitored in its own dialogue which you can reach in the main menu under [Monitor] - [Register+I/O Address Monitor] or alternatively in the project window under the entry [DataFiles] - [Register]. You can compile the memory areas to be monitored randomly and save them for later use. An entry in this list will always consist of a WORD (16 bit) and will contain a maximum of 16 entries (256 bits).

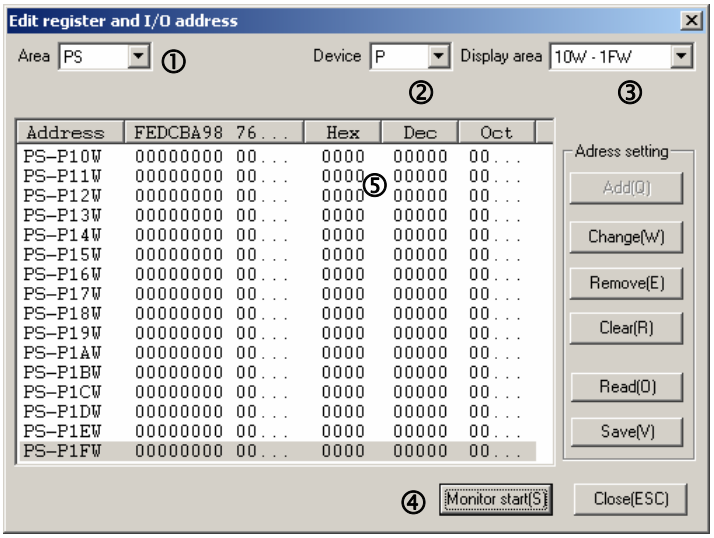


Figure 7-94 Monitor register dialogue

① Selection of the memory area
PS = Memory of the PS program
PN = Memory of the PN program
Common = Common area of the PS/PN memory
② Selection of the memory type
③ Selection of the address area within the memory type
④ Start of monitoring
⑤ List of the addresses to be monitored

Compilation

If you select a display area directly ③ a maximum of 16 entries is automatically compiled. You can alter this compilation randomly and save it on the fixed disk for later use.

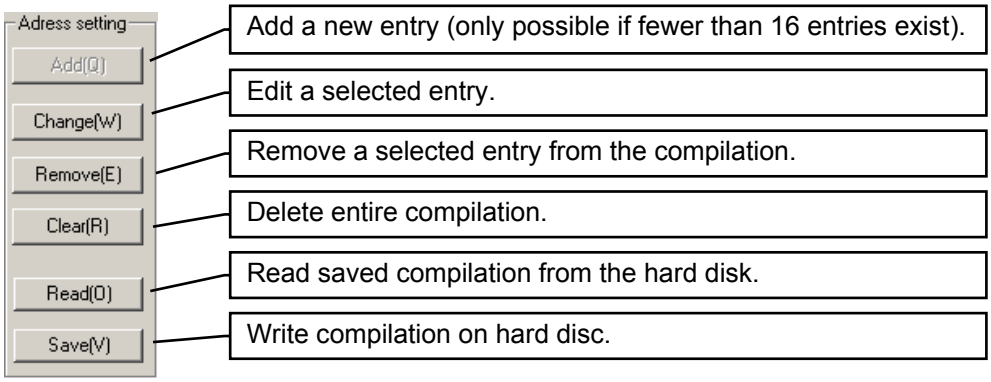


Figure 7-95 Address selection in monitor register dialogue

Adding/Changing

This dialogue can be reached via the „Add“ button (fewer than 16 entries) or via the „Change“ button (if an entry is selected). You have the possibility here to add one or several entries to the compilation.

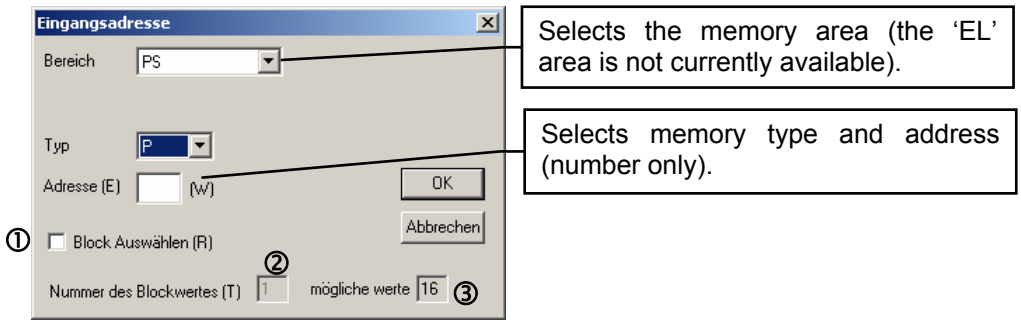


Figure 7-96 Add/change entry in the monitor (register) dialogue

If you select the box „Block Auswählen“ ① as many WORD blocks as specified in ② will be added automatically starting from the entered address. The number of WORD blocks which can be automatically added is restricted to a maximum of 16 and is displayed in ③.

7.16 Documenting

Good documentation is a part of every program. **SFL-SOFT** provides you with two different types of documentation. A simple one to be printed out as required and a detailed one for the finished program to be added to the plant documentation. Both types have the same content and differ only in their visual presentation (formatting).

A print-out in simple presentation can be selected via the menu [File] - [Print]. Select the option [File] - [Drawing Style Print], option for a print-out of a more detailed formatting.

It is essential to print out the program completely for plant documentation to correctly complete program creation. The programmer must enter his name on the cover sheet of the documentation and confirm the correctness of the documentation/of the user program by his signature.

The documentation is subdivided into several areas which can be individually selected and printed out. Before printing out the final documentation you must enter the additional data necessary for documentation.

7.16.1 Setting format

The defaults for the cover sheet and the requisite additional data for documentation are entered here. You can reach this input mask in the main menu under [File] [Drawing Style Print] [Form setup]. In addition every dialogue has a button [Format] to select the print options by which you can similarly reach this input mask.

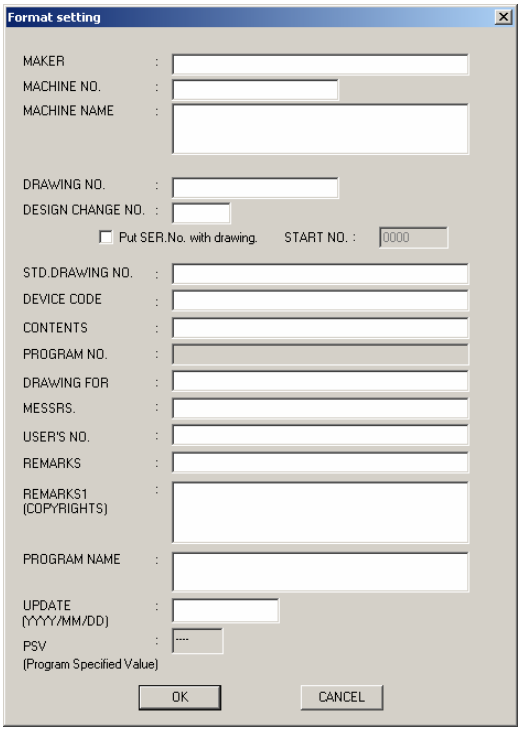


Figure 7-97 Format setting dialogue

In order to insert a line break in the case of multi-line entry fields press the 'RETURN/ENTER' key while simultaneously pressing the 'SHIFT' key .

The date is entered as follows:

YYYY/MM/DD

whereby:

DD = day

MM = month

YYYY = year

7.16.2 Selecting printer

Before starting a print-out you should select a printer. You can reach the appropriate dialogue in the main menu under [File] [Print setup...]. A selection dialogue usual in Windows will appear. Every dialogue also possesses a button to select the print options [Printer setting] with which you can similarly obtain the printer selection dialogue.

7.16.3 Ladder diagram

Ladder diagrams of selected blocks can be printed out by selecting in the main menu [File] [Print...] [Circuit Diagram List (Ladder)...].

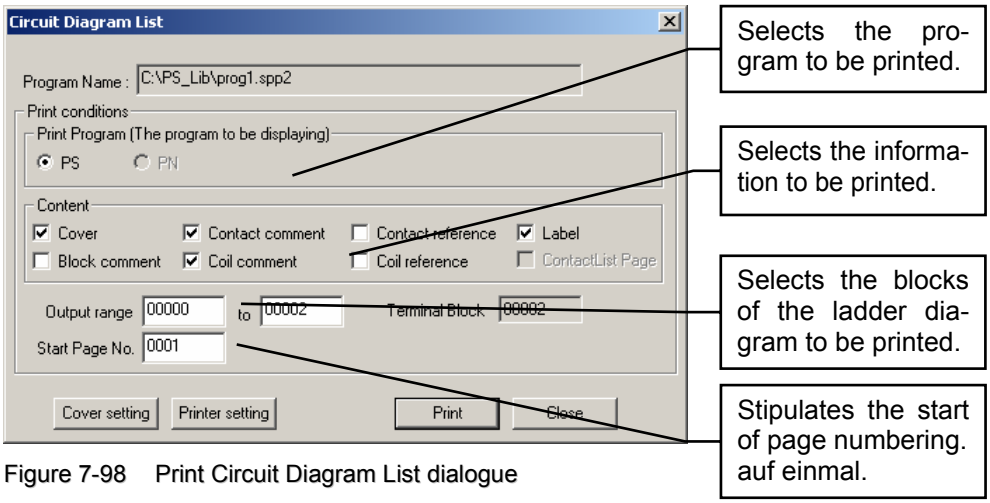


Figure 7-98 Print Circuit Diagram List dialogue

7.16.4 Comment

Comments of selected memory areas can be printed out by selecting in the main menu [File] [Print...] [Comment List...].

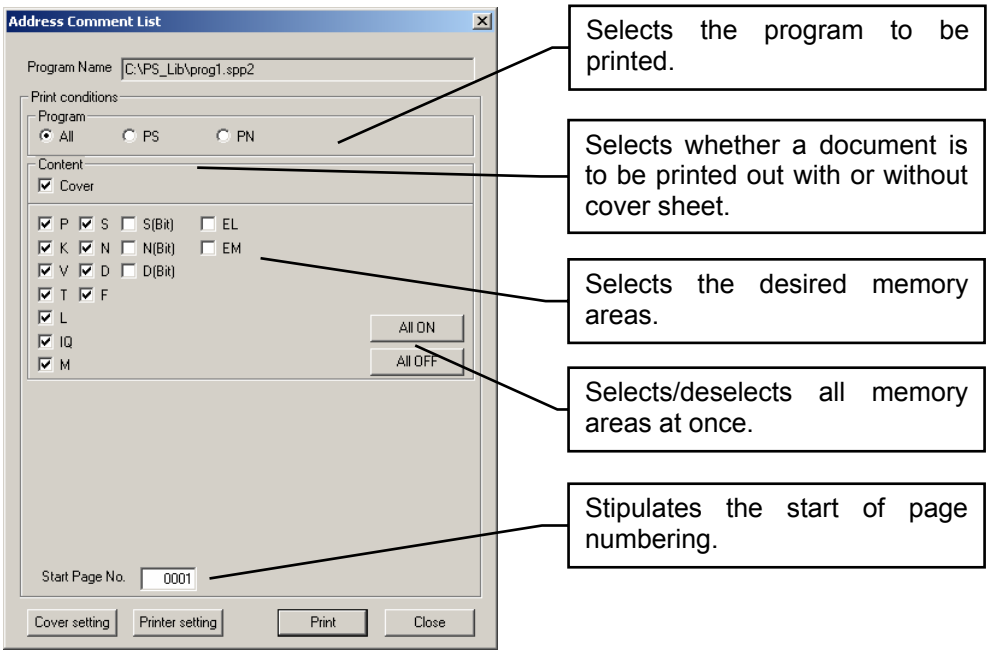


Figure 7-99 Print Comment List dialogue

7.16.5 Contact table

Contact tables of selected memory areas can be printed out by selecting in the main menu [File] [Print...] [Contact Table List...].

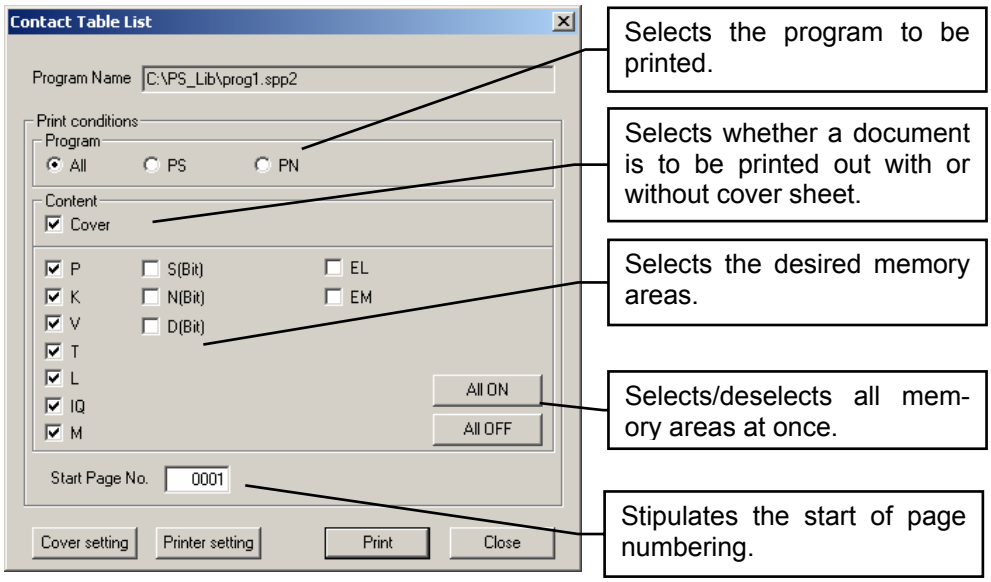


Figure 7-100 Print Contact Table List dialogue

7.16.6 Use status

Used addresses of selected memory areas can be printed out by selecting in the main menu [File] [Print...] [Using Status List...].

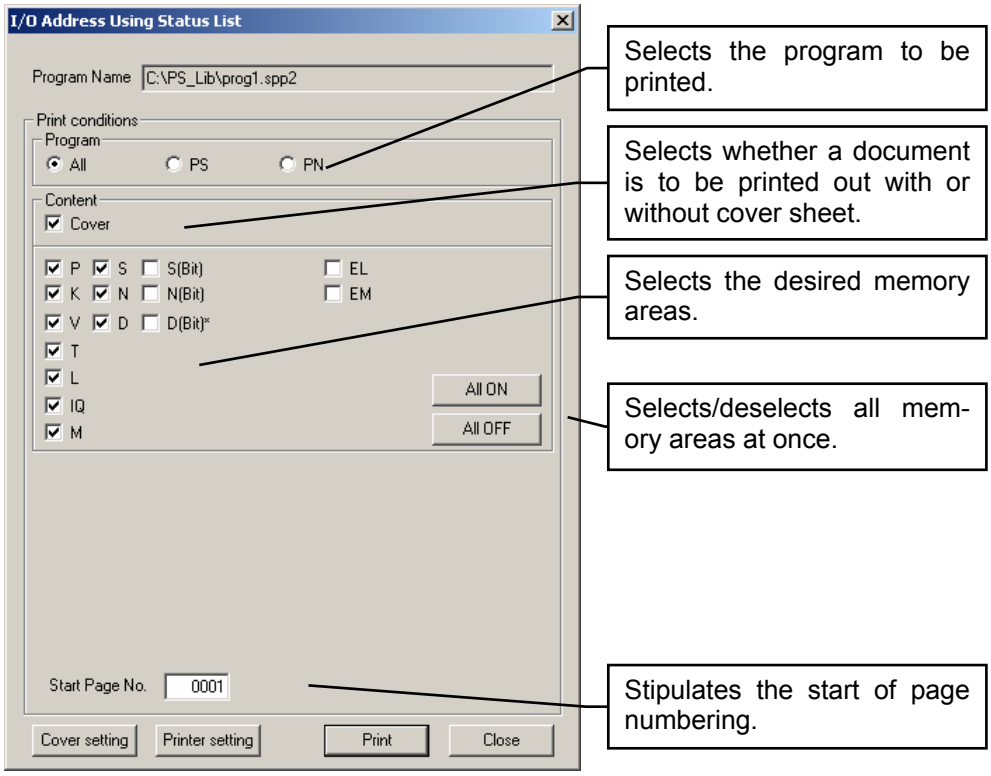


Figure 7-101 Print Usin Status List dialogue

7.16.7 Parameters

A list of the parameter settings (hardware configuration) can be printed out by selecting in the main menu [File] - [Print...] - [Parameter List...].

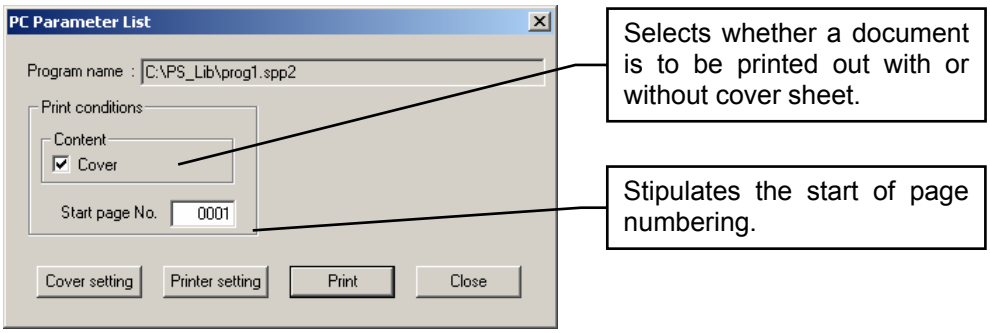


Figure 7-102 Print Parameter List dialogue

7.17 Program options

7.17.1 Presentation

The menu [Option] - [Customize] provides you with presentation options.

Font

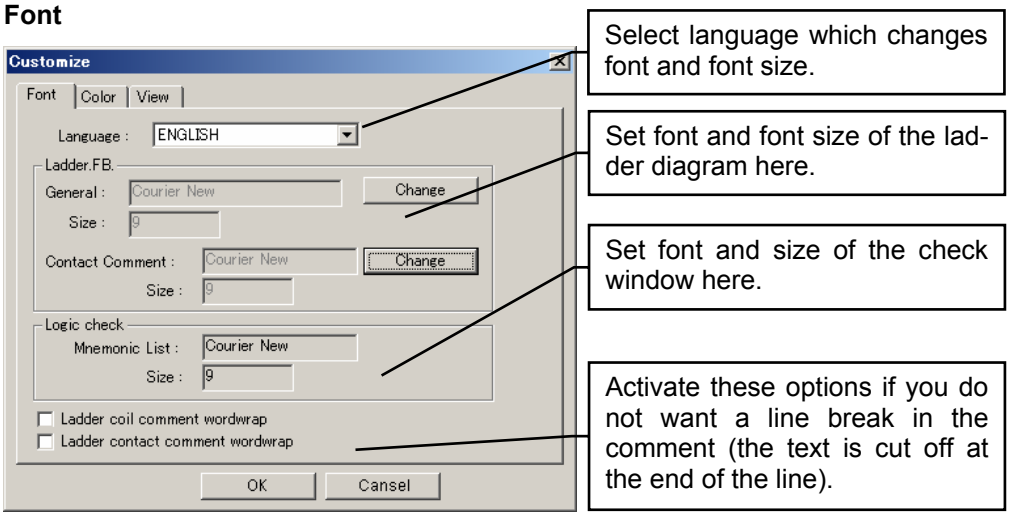


Figure 7-103 Font selection dialogue

Colour

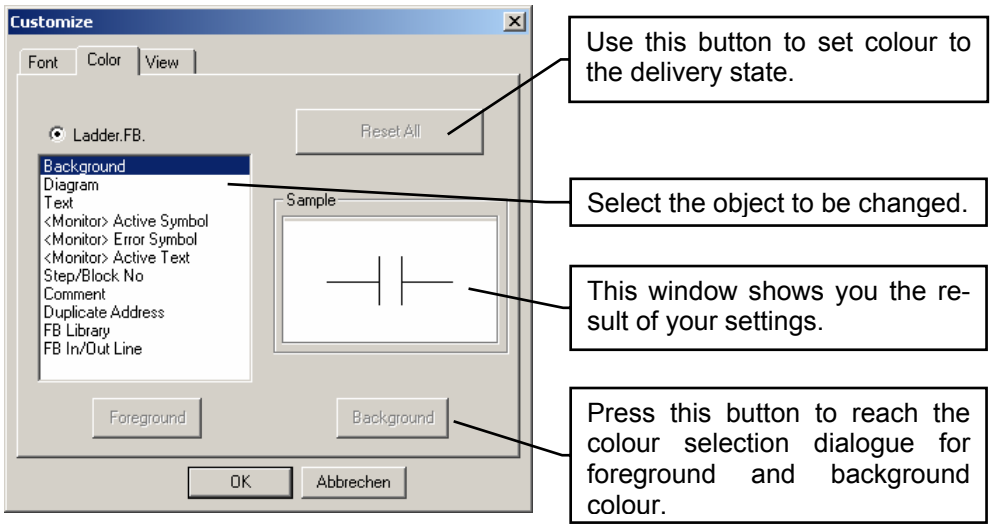


Figure 7-104 Colour settings dialogue

View

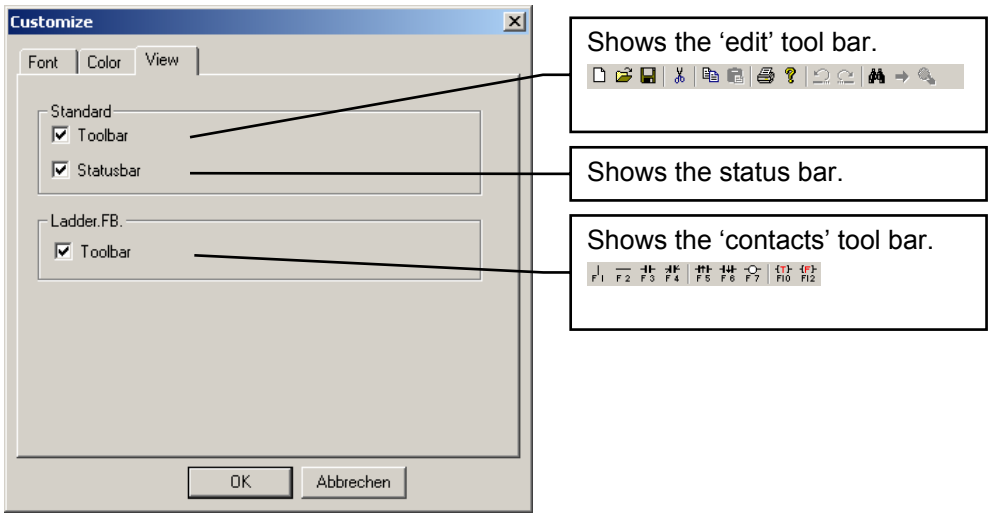


Figure 7-105 View settings dialogue

7.17.2 Configuration

The menu [Option] - [Configuration] provides options to edit and create a program.

File

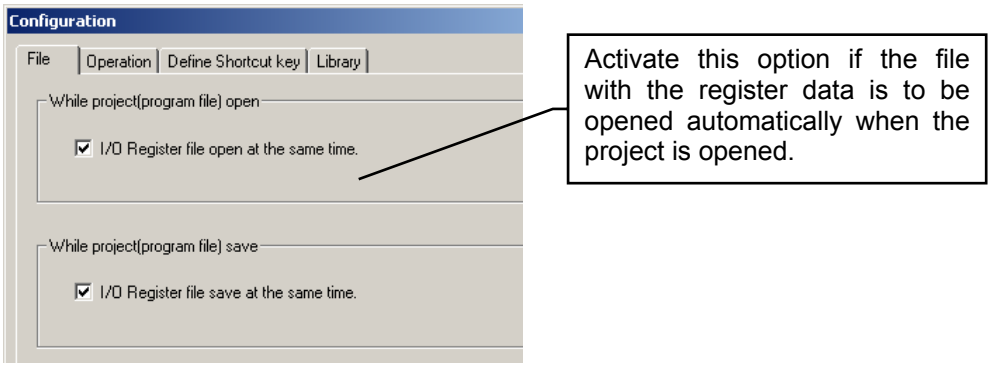


Figure 7-106 File configuration dialogue

Operations

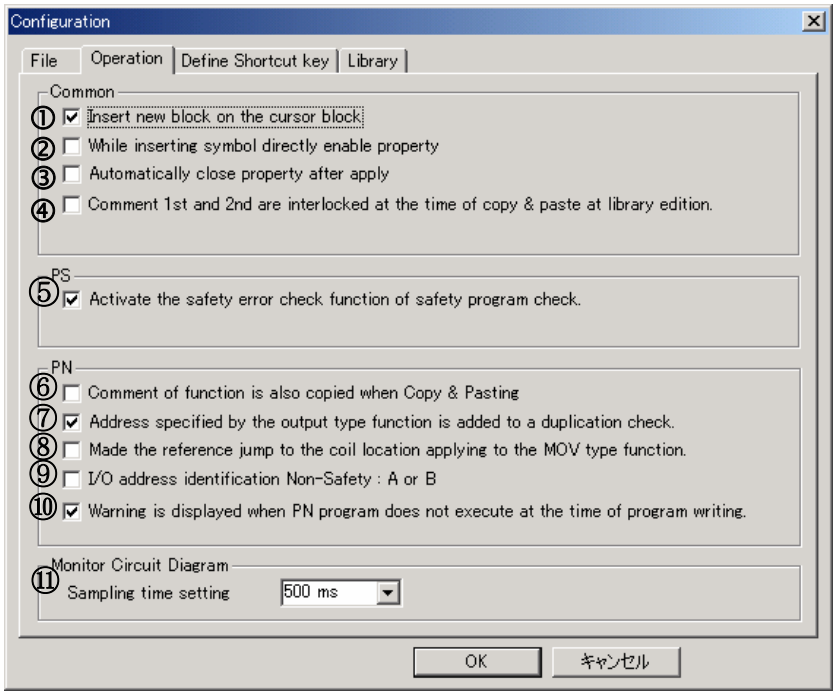


Figure 7-107 Operation configuration dialogue

① Activated: A new block is inserted at the cursor position and the existing block is pushed downwards.
Deactivated: A new block is created beneath the block at the cursor position.

② Activated: The dialogue is automatically opened upon assignment of a contact from 'contacts' in the tool bar and remains open until it is manually closed.
Deactivated: The dialogue is not automatically opened when a contact from the 'contacts' in the tool bar is assigned.

③ Activated: The property dialogue is automatically closed after assignment.
Deactivated: The property dialogue remains open until it is manually closed.

④ Activated: When editing libraries, comment 1 and comment 2 are copied if 'copy and paste' is used.
Deactivated: When editing libraries, comment 1 and comment 2 are not copied if 'copy and paste' is used.

⑤ Activated: A Logic check is done when LD Program Check is performed.
Deactivated: No Logic check is done when LD Program Check is performed.

⑥ Activated: Function comments are copied if 'copy and paste' is used.
Deactivated: Function comments are not copied if 'copy and paste' is used.

⑦ Activated: A check if the same address is used by multible functions for result is done.
Deactivated: No check if the same address is used by multible functions for result is done.

⑧Activated: Perform a reference jump to the coil in MOV function types.
Deactivated: Don't Perform a reference jump to the coil in MOV function types.

⑨Activated: The prefix for I/O memory area is ‘A/B’.
Deactivated: The prefix for I/O memory area is ‘X/Y’.

⑩ Activated: When writing the program and a PN program will not be running, the warning message will be displayed.
Deactivated: The warning message will not be displayed.

⑪The sampling time for the CPU/register monitoring can be set in a range of 100ms to 5000ms.

Defining short-cut keys

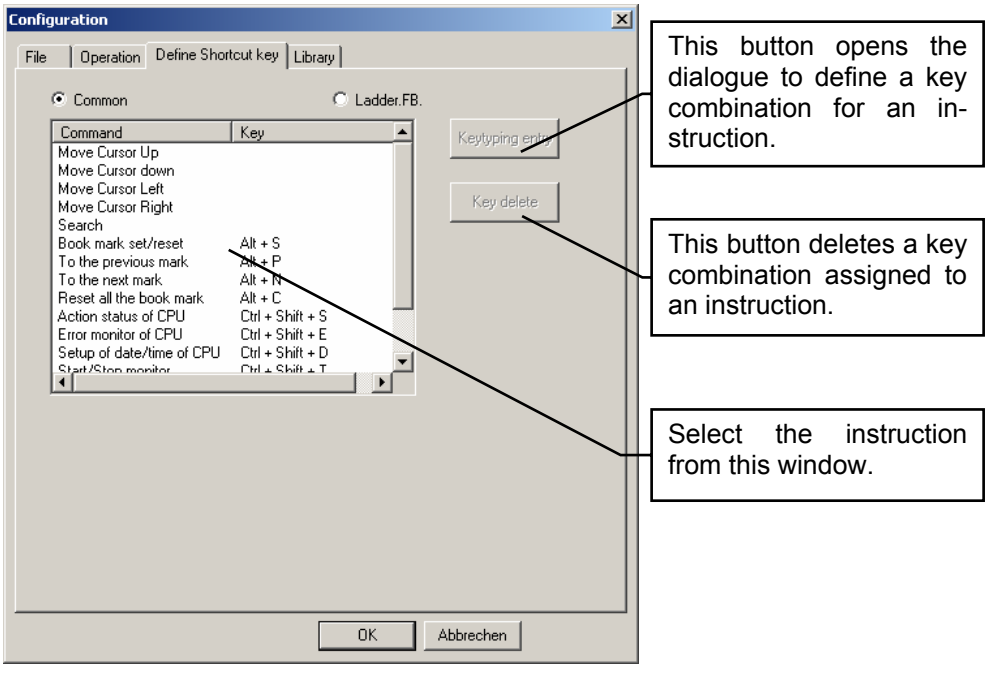


Figure 7-108 Definition of short cut keys dialogue

Definition of keys

In order to define a new key combination, first select the instruction using the mouse and then press the ‘Keytyping entry’ button. You can then press any key/key combination in the following dialogue and assign the previously selected instruction by way of ‘OK’.

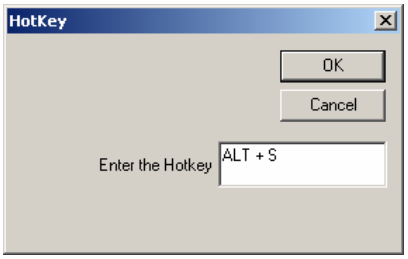


Figure 7-109 Entering a short-cut key

Deleting key

In order to delete an assigned key combination, first select the instruction using the mouse and then press the ‘delete key’ button.

Library

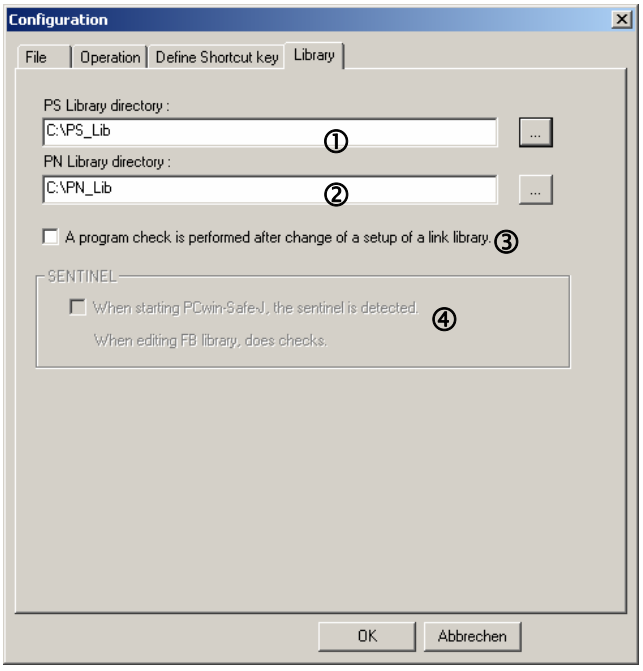


Figure 7-110 Configuration library dialogue

① Select PS library directory.
② Select PN library directory.
③ Activate this option to have **SFL-SOFT** perform a check on the program when library is linked.
④ Activate the hardware dongle enquiry (without function).

*1 : ① and ② can not be set under the same directly.

7.18 CPU initialisation

The CPU module can be set to the delivery state again using the **SFL-SOFT** main menu dialogue [CPU] - [CPU initialization]. The password is identical with the password which has been used to activate the edit mode.

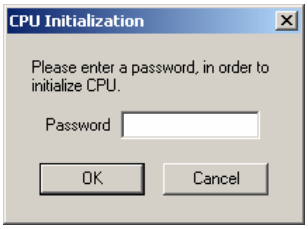


Figure 7-111 CPU initialisation

An initialisation can be carried out if the switch of the CPU module has been set to ‘W.E.’ and the ‘R.ST’ button has been pressed.

8 Operation and maintenance

This chapter explains first start-up, maintenance and troubleshooting of the **SFL**.

8.1 Delivery status

The CPU module is delivered in the following state:

- PIN Code „0000“
- No PS/PN program
- No parameters

The user must change the PIN code (“0001” .. “9999”). The new PIN code is stored in the flash of the CPU module.

8.2 Start-up

Before operating the **SFL** controller for the first time it is advisable to incorporate a shut-down device (EMERGENCY-OFF [EMERGENCY-STOP] or similar) into the power supply.

If you have not already done so, first assemble the back-up battery as described in Chapter 4.2.3 before commissioning.

Switching on

If you have installed a booster module, the following must be observed when switching on.

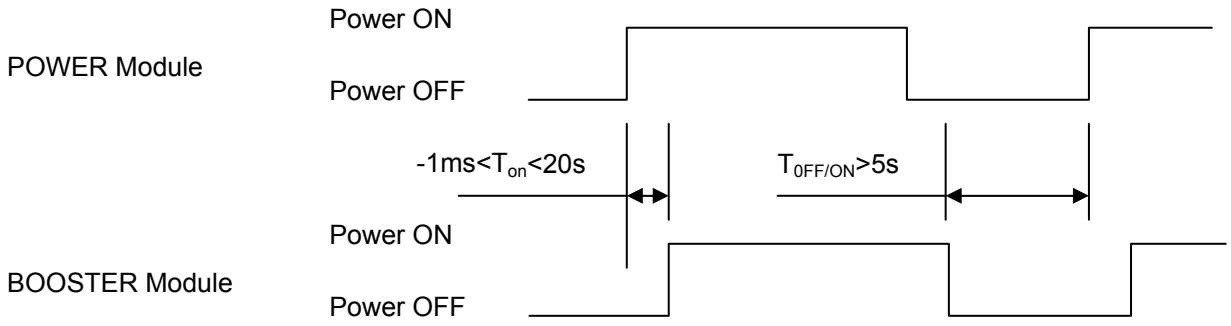
If the power supplies of the booster and power on module cannot be switched on at the same time, the following order must be observed:

1. Booster module
2. Power module

Take care about power supplies ON timing as follows.

Ton : Timing, after turn-ON power supply of POWER Module to turn-ON BOOSTER Module power supply

TOFF/ON : Timing, after turn-OFF BOOSTER Module power supply to turn-ON POWER Module power supply



If such timing is not observed, an error may be produced during system initialization and USB communication may become difficult.

8.3 Disturbances/error messages

8.3.1 Error categories

The error messages are subdivided into 3 categories. The category of error can be recognised by the states of the special flags V01 to V03.

Alarms

Alarms are signalised by a set flag V03. They are shown in the display of the CPU module by the prefix “AL” before the error code. An alarm warning does not lead to a functional impairment of the **SFL**. Safe operation is guaranteed even with a permanent alarm. An alarm can only be recognised by the messages of the display or by query of flag V03.

Slight errors

Slight errors are signalised by a set flag V02. They are shown in the display of the CPU module by the prefix "ER" before the error code. Their occurrence usually indicates incorrect settings, errors in the user program or errors in the cabling (e.g. cable break or short circuit to ground). The **SFL** moves to the safe state (all safe outputs are shut down). The user program continues to be executed (interrogation of the inputs, monitoring by **SFL-SOFT**) without the possibility to enable the Outputs. Leave the safe state is only possible by Power OFF/ ON or be activating the Restart Button.

Serious errors

Serious errors are signalised by a set flag V01. They are shown in the display of the CPU module by the prefix "ER" before the error code. These errors are usually hardware errors. If an error of this kind occurs, the user program is immediately interrupted. The **SFL** controller moves to the safe state. Leave the safe state is only possible by Power OFF/ ON or be activating the Restart Button.

8.3.2 Error memory

When an error occurs, the error code, further additional information as well as the time of the error are stored in a special error memory. This memory can be read out using the **SFL-SOFT** programming software.

Ring buffer

The error memory is designed as a ring buffer. The last 8 errors are saved. If all 8 entries are used and if a further error then occurs, the oldest entry is overwritten. Existing entries are not deleted after the error has been eliminated. It is not possible to manually delete the error messages.

If there is no back-up battery, all errors are deleted following a Power OFF.

Error entry

An error entry takes up 10 memory spaces (20 bytes). The entries are displayed hexadecimally in BCD code. A byte serves to represent a decimal number (0...9). The year figure specifies the decade. The day of the week is represented by the figures 0…6, Sunday to Saturday. Depending on error code you will find additional information on the errors which have occurred in S201 and S202. The **SFL-SOFT** programming software provides the possibility to display error messages in plain text with information on their elimination in addition to viewing as a memory excerpt.

Adress	
S200	Entry 1
S20A	Entry 2
S214	Entry 3
S21E	Entry 4
S228	Entry 5
S232	Entry 6
S23C	Entry 7
S246	Entry 8

Adress	Highbyte	Lowbyte
S200	Error code	
S201	Additional information 2	Additional information 1
S202	Additional information 4	Additional information 3
S203	Time occurred (seconds) = 0000 to 0600	
S204	Time occurred (minutes) = 0000 to 0600	
S205	Time occurred (hour) = 0000 to 0203	
S206	Time occurred (day) = 0001 to 0301	
S207	Time occurred (month)= 0001 to 0102	
S208	Time occurred (year) = 0000 to 0909	
S209		Time occurred (day of the week))

Table 8-1 Structure of the error memory

Example

Address FEDCBA98 76543210 HB LB

0200h 10100000 01001000 A0 48 => Processor A, error 048
0201h 00000001 00000000 01 00 => add. info 1=0 , add. info 2=1
0202h 10100000 00100010 00 22 => add. info 3=22, add. info 4=0
0203h 00000000 00000100 00 04 => second = 4
0204h 00000011 00000110 03 06 => minute = 36
0205h 00000001 00000010 01 02 => hour = 12
0206h 00000001 00000000 01 00 => day = 10
0207h 00000001 00000000 01 00 => month = 10
0208h 00000000 00000011 00 03 => year = 03
0209h 00000000 00000011 00 03 => day of the week = 3 = Wednesday

On Wednesday 10.10.2003, at 12:36 and 4 seconds, processor A in Slot 1 detected an error in the input comparison at Address 0.

8.3.3 Error messages/codes

Overview

The following table provides an overview of the error codes and their meaning. Every error message can be generated both by processor A and by processor B. The prefix A/B indicates the processor from which the error message originates. An error message can be queried within the user program by means of special flags. A set flag shows an existing error. The categorisation of the error (how serious) is shown by the flag specified in the “stage” column. More information on error categorisation is provided in Chapter 8.3.1.

8.3.1 / 8-3

Error code	Special-flag	Stage	Meaning
A/B 013	V0C1	V001	Error 24V supply voltage
A/B 021	V0C2	V001	CRC error program memory
A/B 022	V0F0	V003	Voltage of the back-up battery too low.
A/B 023	V0C2	V001	CRC error program parameter.
A/B 031	V0C9	V001	Execution time of the user program too long.
A/B 032	V0C0	V001	Read error in the RAM memory.
A/B 035	V0C0	V001	System error in the CPU module.
A/B 042	V0C5	V001	Parameter error Input/Output module.
A/B 043	V0E0	V002	Hardware error Input/Output module.
A/B 048	V0C8	V001	Configuration error Input/Output module
A/B 071	V0C9	V001	Faulty statement in the user program
A/B 0A3	V0F5	V003	Real time clock error.
A/B 0AB	V0CA	V001	Write error program memory (flash).
A/B 0AC	V0C0	V001	Error in circuitry of battery monitoring.
A/B 0AD	V0C2	V001	Data error
A/B 0AE	V0CB	V003	Data error not detected
A/B 0AF	V0F5	V003	RTC not set
A/B 0B0	V0C0	V001	Communication error between Processor A and B.
A/B 0B1	V0C3	V001	Transmission error back plane BUS.
A/B 0B5	V0C3	V001	Error back plane BUS.
A/B 0FE			Error 24V DC supply
A/B 0FF			Although the error signal from an I/O module was detected, the I/O module which becomes error does not exist.

Table 8-2 Brief description of error codes

Reference

Error	Info	Type	Cod	Description
A/B 013	Error 24V supply voltage			
	1	Error	00	Error in the power module
			01	Error in the booster module
	2	Error	00	Power on : booster module with fewer than 10 mod-
			01	Power on : no booster module with more than 9
			02	Operation : booster module with fewer than 10
			03	Operation : no booster module with over 9 modules
A/B 021	A CRC error has occurred in the internal program memory.			
A/B 022	Voltage of the back-up battery is too low.			
A/B 023	CRC error program parameter.			
A/B 031	The maximum execution time of the user program has been exceeded.			
A/B 035	A serious error has occurred in the **SFL–CPU**.			
	1	Error	00	Hardware error or program error
			01	Error in time slice or event processing.
			02	Error in data comparison between MP-A and MP-B
A/B 042	A parameter error has occurred in an Input/Output module.			
	1	Station	→	Station number (00h)
	2	Slot	→	Slot number (00h…0Fh; FFh = indeterminate)
	3	Error	01	Faulty assignment of an Input/Output module
A/B 043	An error was detected during the self-diagnosis of an Input/Output module.			
	1	Station	→	Station number (00h)
	2	Slot	→	Slot number (00h…0Fh; FFh = indeterminate)
	3	Error	11	Error during testing of the RAM
			12	Error during testing of the ROM
			13	Error during testing of the processor
			14	Error during testing of the 24V DC supply
			15	Error during testing of the shut-down relay
			16	Error management LOW
			17	Error FME pin
			21	Input: error during test pulses
			22	Input: input level different
			31	Output: error during test pulses
			32	Output: error during reading back
			33	Output: error relay contact
			34	Output: error output circuit check
			41	Communication MP-A and MP.B: Error connection set-up
			42	Communication MP-A and MP.B: Error during transmission
			43	Communication MP-A and MP.B: Error telegram format
			44	Communication MP-A and MP.B: Error watchdog
			45	Communication MP-A and MP.B: Comparison error of input data
			46	Communication MP-A and MP.B: Comparison error of the output data
			51	Communication back plane BUS: Error connection set-up
			52	Communication back plane BUS Error during transmission
			53	Communication back plane BUS Error telegram format
			54	Communication back plane BUS Error watchdog
			61	Parameter: format error of the Input/Output pa-
			62	Parameter: data error of the Input/Output parameter
			71	EEPROM: read error
			72	EEPROM: write error
	4	Chan-	→	Faulty Input/Output channel (00h…1Fh)

Table 8-3 Reference error messages 1/2

Reference (continued 1)

Error code	Info	Type	Code	Description
A/B 048	The module assembly and the Palametrierung do not agree.			
	1	Station	→	Station number (00h)
	2	Slot	→	Slot number (00h…0Fh; FFh = indeterminate)
	3	Data	→	Parameterised module ID
	4	Data	→	Read out module ID
A/B 071	The result of an operation is outside the permitted value range.			
	1	PC	L	Program counter low byte
	2		H	Program counter high byte
	3	Number	L	Instruction number low byte
	4		H	Instruction number high byte
A/B 0A3	An error has occurred in the real time clock.			
	1	Error	01	Write / read error
A/B 0AB	A write error has occurred in the flash ROM memory.			
	1	Error	01	Internal flash ROM
			02	External flash ROM
	2	Error	11	Error during deletion
			12	Error during writing
			13	Error during comparison
A/B	An error has occurred in the battery monitoring circuit.			
A/B	Battery back-up data could not be restored .			
A/B 0AE	Checking for A/B OAD errors by **SFL-SOFT** not yet performed.			
A/B 0AF	RTC not set			
A/B 0B0	A communication error has occurred between Processor A and Processor B.			
	1	Error	01	Time out error
			10	Error in serial communication
			11	Reception error
			12	Transmission error
			20	Invalid function call
			30	Faulty data received
			32	Transmission error
			FF	Other error
A/B 0B1	A communication error has occurred on the back plane BUS			
	1	Error	01	Time out error
			10	Error in serial communication
			11	Reception error
			12	Transmission error
			20	Invalid function call
			30	Faulty data received
			32	Read-back error
			80	Intial error/Error signal error
			FF	Other error
	2	Station	→	Station number (00)
	3	Slot	→	Slot number (00h…0Fh; FFh = indeterminate)
			7F	Simultaneously 2 or several slots
A/B 0B5	An error has occurred in the system initialisation.			
	1	Error	01	Initialisation of a module not completed.
			02	A module has produced an error message.
A/B 0FE	Error in the 24VDC supply			
A/B 0FF	Although the error signal from an I/O module was detected, the I/O module which becomes error does not exist.			

Table 8-4 Reference error messages 2/2

8.4 Maintenance

Assuming correct assembly and circuitry the maintenance work is restricted to the changing of the back-up battery (approx. every 5 years). The battery must be changed if the error code 022 or the special flag VF0 has not been deleted after more than a 4-hour charging period. If the error message continues after replacing the battery, an internal error of the CPU module exists. In this case please consult your supplier.

8.4.1 Changing the battery

4.2.3 / 4-9  See Chapter 4.2.3.

8.4.2 Defective fuses

If the internal fuse of a module is destroyed due to faulty circuitry or a component defect, do not attempt under any circumstances to replace it yourself. Please consult your supplier to repair the module.

8.4.3 Failure as a result of common cause

A failure as a result of common cause is to be given special consideration during installation, service and maintenance work. This is a failure which may be the result of one or several events causing the simultaneous failure of both channels of the **SFL** and thus may lead to a system failure. The channel separation must be maintained during all work to and with the **SFL**.

8.5 Service

In the case of defective modules or other problems please consult one of the following addresses:

Asia

Panasonic Industrial Devices SUNX Co., Ltd.

Overseas Sales Dept. (Head Office)

2431-1 Ushiyama-cho, Kasugai-shi, Aichi, 486-0901, Japan

Phone: +81-568-33-7861 Fax: +81-568-33-8591

Europe

Europe Headquarter: Panasonic Electric Works Europe AG

Rudolf-Diesel-Ring 2, D-83607 Holzkirchen, Germany

Phone: +49-8024-648-0

North America

US Headquarter: Panasonic Electric Works Corporation of America

629 Central Avenue New Providence, New Jersey 07974 USA

Phone: +1-908-464-3550

9 Annex

9.1 Technical data

9.1.1 General data (valid for all modules)

Position	Description				
Supply voltage	24V DC ±10%				
Power input	Current: max. 2A				
Wattage	Max. 48W				
Ambient temperature	0 to 55°C				
Air humidity	30 to 85% RH (non-condensing)				
Air pressure	86kPa bis 106kPa				
Ambient atmosphere	No aggressive gasses permitted				
Degree of pollution	2 to DIN EN 50178				
Rated isolation voltage	Protective class I (<50 V * √2 = 71 V)				
Area of use	II : Average environmental and operational influences are to be expected, the assembly side (e.g. electrical installation room) protects the installation from strong influences (e.g. vibration-dampened side of information of the switch cabinet).				
Installation site	Metal switch cabinet protective class at least IP54				
Proof test	20 years				
Mode of operation	High demand rate or continuous demand limited by the system reaction time (high demand or continuous mode)				
Oscillations	DIN EN 60068-2-6	Frequency	Accel.	Amplitude	Runs
		10…57Hz	-	0,35 mm	20 cycles (1 octave/min)
		57…150Hz	5,0 g	-	
Shock impact	DIN EN 60068-2-29. (10g 1000 ±10 times, X,Y,Z Direction)				
Voltage interruption	Maximum 10ms at intervals of minimum 1 second				

Table 9-1 General technical data for all modules

9.1.2 Electromagnetic compatibility (valid for all modules)

The functional safety is guaranteed even in the case of higher interference levels (typical factor 2).

Position	Norm	Specification
Discharge of static electricity	IEC 6100-4-2	Contact discharge : ±6kV Air discharge : ±8kV
Electromagnetic HF field	IEC 6100-4-3	Housing : 80 ... 2000MHz / 10V/m
Fast transient	IEC 6100-4-4	Signal connections : ±1kV Direct current inputs : ±1kV Function ground : ±1kV
Surge voltages	IEC 6100-4-5	Signal connections : ±1kV Direct current inputs : ±1kV
High frequency	IEC 6100-4-6	Signal connections : Direct current inputs : Function ground : 0.15 ... 80MHz / 10 V

Table 9-2 Electromagnetic compatibility

9.1.3 SFL-CPU-MON-V2／SFL-CPU-OP-MON-V2

Position	Description
Supply voltage	24V DC ±10 %
Power input	2.5A with fully equipped station (180mA CPU modul only)
Fusing	internal fuse 3.2A
Output voltage	24V DC / max. 1A (in case of error!)
Dimensions / weight	45 × 100 × 80 (W/H/D) / 230g

Position	Description
Program system	Stored user program
Program execution	Cyclical calculation
Input/Output control	Register display
Execution time	15ms
Basic commands	PS:16 PN:16
Timer commands	PS:4 PN:4
Function commands	PS:7 PN:54
Program memory	12K WORD (internal memory: 64kB)
Memory type	CMOS-RAM, FLASH-ROM
Back-up battery	Rechargeable (Lithium Battery: life cycle max. 5 years)
External I/O channels	PS:256 PN:256
Internal I/O channels	PS:1024 PN:1024
Hold flags	PS:768 PN:768
Link flags	PS:2048 PN:2048
Edge-detection flags	PS:512 PN:512
Data register	PS:512 Bytes PN:1024 Bytes
Value register	PS:512 Bytes PN:2048 Bytes
Special register	PS:2048 Bytes PN:2048 Bytes
Comment memory	64k bytes
Communication(PC3)	SN-I/F
Display	4 digit 7 segment display

Table 9-3 **SFL-CPU**

9.1.4 SFL reaction time

Output SFL	System reaction time
Mode 1	
Semi-conductor	52.6ms
Relay	67.6ms
Mode 3	
Semi-conductor	22.6ms
Relay	37.9ms
Transmission of EL data	47,8ms

Table 9-4 System reaction timet

9.2 Considered standards

standard	Designation
DIN EN 61508; 1-7	Functional safety of safety-related electrical, electronic, programmable, electronic systems
DIN EN 954-1:1997-03	Safety of machines - safety-related parts of control systems Part 1: General design principles; German version EN 954-1:1996
DIN EN ISO 13849-1:2007-07	Safety of machines - safety related parts of control systems Part 1: General design principles (ISO 13849-1:2000); German version EN ISO 13849-1:2006
DIN EN ISO 13849-2:2003-12	Safety machines - safety related parts of control systems Part 2: Validation (ISO 13849-2:2003); German version EN ISO 13849-2:2004
DIN EN 60204-1:2007-06	Safety of machines – electrical equipment of machines Part 1: General requirements (IEC 60204-1:2005, modified); German version EN 60204-1:2006
DIN EN 61000-2-5:1995	Electromagnetic compatibility

Table 9-5 Considered standards

9.3 Declarations of conformity

Itemized Essentials of EC Declaration of Conformity

Manufacturer's Name: Panasonic Industrial Devices SUNX Co., Ltd.
Manufacturer's Address: 2431-1, Ushiyama-cho, Kasugai, Aichi 486-0901, Japan

EC Representative's Name: Panasonic Marketing Europe GmbH Panasonic Testing Center
EC Representative's Address: Winsbergring 15, 22525 Hamburg,Germany

Product: Safety Multi-application Controller
Model Name: SFL Series
Trade Name: Panasonic

Application of Council Directive:

- 2006/42/EC Machinery Directive
- 2004/108/EC EMC Directive

Tested according to:

- EN ISO 13849-1: 2008
- EN ISO 13849-2: 2008
- EN 61131-2: 2007

Type Examination: Certified by BGIA Institut für Arbeitsschutz der Deutschen Gestzlichen Unfallversicherung
Alte Heerstraße 111, 53757 Sankt Augustin Germany

Please contact .....

Panasonic Industrial Devices SUNX Co., Ltd.
■ Overseas Sales Division (Head Office): 2431-1 Ushiyama-cho, Kasugai-shi, Aichi, 486-0901, Japan
■ Telephone: +81-568-33-7861 ■ Facsimile: +81-568-33-8591
panasonic.net/id/pidsx/global

About our sale network, please visit our website.

March, 2013 PRINTED IN JAPAN
ME-SFL-06