Programmable Valve Terminal Type 03
Electronics Manual
Control Block SF 3

Festo PLC integrated
Field bus protocol: Festo field bus

Manual 0503d 165 486 GB
Advantages of programmable valve terminals

Many control tasks in the field of pneumatics can be automated without a control cabinet (e.g. cylinders, sensors, analogue I/Os). A valve terminal with integrated programmable control (control block SF 3) and with the command set and function range of a powerful mini control system can be programmed simply with Festo programming software FST 200.

There are two designs of control block
- SF 3-02 for valve terminal type 02
- SF 3-03 for valve terminals types 03/04-B/05

Advantages of programmable valve terminals with control block SF 3
- Built-in PLC with field bus module
- Protection class IP65 – no control cabinet required
- Simple equipment layout with independent control on site
- Low wiring requirements
- Up to 128 local inputs and 128 local outputs
- Operating mode options: standalone, master or slave
- Inspected unit
- Electrical inputs, e.g. for sensors
- Electrical outputs, e.g. for electrical actuators
- Dependent on type of valve terminal: Modules with analogue I/Os, AS-i Master or CP module also available
Chapter summary

Chapter 1  User instructions and system summary
contain information irrespective of the type of valve
terminal and the selected node.

Chapter 2  System description of valve terminal
contains the system description of the valve
terminal ordered.
Part 2a (option): Valve terminal, type 02
Part 2b (option): Valve terminal, type 03

Chapter 3  System description of control block SF 3
contains all information specific to PLCs
(programmable logic controllers) for standalone
operation.

Chapter 4  System description of field bus module
contains additional information about the master
and slave operating modes for the field bus.

Chapter 5  Description of analogue modules
(only types 03...05)
contains all information about actuation of
analogue modules.

Chapter 6  Description of AS-i master (only types 03...05)
contains all information about actuation of an
AS-i bus system.

Chapter 7  Description of CP interface (only types 03...05)
contains information about the actuation of a
CP system.

Appendix A  Technical Appendix, examples of circuitry

Appendix B  Summary of operands, function modules (CFM)
and error messages

Appendix C  Command interpreter

Appendix D  Glossary / Index
### Amendments to manual version 9610a  
(as from software status 0198, V1.6)

<table>
<thead>
<tr>
<th>Modification</th>
<th>Function/effect</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional remanent data words supported by function modules 5 and 6</td>
<td>As from hardware status 10.97, the data remanent values (saved and protected against loss due to power failure) have been extended by 512 words (16-bit values). These remanent data words are freely available and are supported by function modules 5 (Read remanent data) and 6 (Write remanent data). The former remanent operands are kept (downwards compatible).</td>
<td>3-24, 3-43, B-18, B-19</td>
</tr>
<tr>
<td>Extension of function module 35 (error interrogation of all AS-i slaves)</td>
<td>If any parameter is transferred when function module 35 is accessed, the address of the first faulty AS-i slave will be shown as a decimal value in return parameter P4. This value is zero if the registered AS-i slave is functioning correctly. If access is made without a transfer parameter, the existing functioning method will be kept.</td>
<td>6-96, B-37</td>
</tr>
<tr>
<td>AS-i fault when starting</td>
<td>The starting reaction of the SF 3 control block to an AS-i fault has been modified. The factory default setting is now the soft reaction to faults, i.e. further treatment of the fault is possible in the user program. This applies both to AS-i faults triggered by Power Failure on the AS-i master, as well as AS-i faults triggered by a faulty/failed AS-i slave. The hard reaction to faults (Stop all programs and reset local outputs) can be set, as before, with function module 37</td>
<td>6-80, B-40</td>
</tr>
<tr>
<td>Extension to function module 44 (status interrogation of a field bus slave)</td>
<td>If no parameter is transferred when function module 44 is accessed, the address of the first faulty field bus slave, as well as that of the first failed field bus slave are shown as decimal values in the return parameters. These values are zero if the registered field bus slaves are functioning correctly. If access is made with the address of a field bus slave as transfer parameter, the existing functioning method will be kept.</td>
<td>4-34, B-55</td>
</tr>
<tr>
<td>Modification</td>
<td>Function/effect</td>
<td>Page</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------</td>
<td>------</td>
</tr>
<tr>
<td>New function modules 50 and 51 for identifying a Festo field bus slave</td>
<td>If a registered slave is to be clearly identified as Festo field bus master by the SF 3 control block, the information, stating which slave is involved, can be read out as clear text and stored as a string in the field for special operands FU48 ... FU4095 with the aid of function module 50 (Read information of a field bus slave). The address of the special operand will be specified as transfer parameter when the function module is accessed. In order to repeatedly read out the clear text message of a field bus slave, it is necessary in the case of most Festo field bus slaves to reset this first with function module 51 (Reset field bus slave).</td>
<td>B-61, B-63</td>
</tr>
<tr>
<td>New function modules 10 and 11 for interrupt-controlled counters/timers</td>
<td>Function modules 10 and 11 enable you to register fast counting procedures or to control time-dependent events when processing user programs, irrespective of the cycle time. Parametrizing is undertaken with function module 10 (Define/read out interrupt-controlled counters/timers); the interrupt-controlled counters/timers are started or stopped with function module 11 (Block/release interrupt-controlled counters/timers).</td>
<td>3-46, B-20, B-26</td>
</tr>
</tbody>
</table>
Programmable valve terminal
with
control block SF 3

Chapter 1:
User instructions and system summary
Contents

1.1 GENERAL SAFETY INSTRUCTIONS
- Intended use ........................................ 1-3
- Target groups ...................................... 1-4
- Danger categories ................................. 1-5
- Pictograms ......................................... 1-6
- Instructions for this manual .................... 1-7
- Service ............................................. 1-8

1.2 SYSTEM SUMMARY
- System structure .................................... 1-11
- Valve terminals in standalone operating mode ........................................... 1-12
- Valve terminal in master operating mode ................................................. 1-14
- Valve terminal in slave operating mode .................................................... 1-16
- Operating equipment with ABG (DCU) .................................................... 1-18
- Emulator as commissioning support ......................................................... 1-19
- Programming software FST 200 ............................................................... 1-20
- Valve terminals with analogue modules .................................................... 1-22
- Valve terminals with AS-i master ............................................................. 1-24
- Valve terminal with CP interface ............................................................ 1-26

1.3 SYSTEM LIMITS AND PLANNING ASPECTS
- System limits ........................................ 1-29
- Planning aspects of valve terminals types 03...05 .................................. 1-31
1.1 GENERAL SAFETY INSTRUCTIONS
1.1 GENERAL SAFETY INSTRUCTIONS

Intended use . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-3
Target groups . . . . . . . . . . . . . . . . . . . . . . . . . . 1-4
Danger categories . . . . . . . . . . . . . . . . . . . . . . . . 1-5
Pictograms . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-6
Instructions for this manual . . . . . . . . . . . . . . . . . 1-7
Service . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1-8
1.1 GENERAL SAFETY INSTRUCTIONS

Intended use

The valve terminal documented in this manual is only intended for the following use:

• Control of pneumatic and electric actuators (valves und output modules).
• Interrogation of electric sensor signals through the input module.

Only use the valve terminal as follows:

• in keeping with intended use
• in faultless technical condition
• without modifications.

When connecting standard trade components, such as sensors and actuators, given limits for pressures, temperatures, electric data, torques etc. should be observed.

Note trade association regulations, technical monitoring body, VDE conditions or relevant national conditions.
Target groups

This manual is for the exclusive use of trained specialists in control and automation (technology), who have experience of installation, commissioning, programming and diagnosis of programmable logic controllers (PLC) and field bus systems.
Danger categories

This manual contains instructions about possible dangers which may occur if the product is not used correctly.

A distinction is made between the following instructions:

- **WARNING**
  - This means that injury to human beings as well as material damage can occur if these instructions are not observed.

- **CAUTION**
  - This means that material damage can occur if these instructions are not observed.

- **PLEASE NOTE**
  - This means that this instruction must also be observed.
Pictograms

Pictograms and (graphical) symbols supplement the danger instructions and draw attention to the nature and consequences of the dangers. The following pictograms are used:

Uncontrolled movements of loose conduit (tubing).

Uncontrolled movement of the connected actuators.

High electric voltage or undefined switching states of the electronic components which affect the connected circuits.

Electrostatically vulnerable components. These will be destroyed if their contact surfaces are touched.

If a programmable valve terminal with analogue modules or AS-i master is used, refer to important instructions in chapters 5 und 6.

Valve terminals are very heavy. Ensure they are mounted properly. Wear protective shoes.
Instructions for this manual

This manual uses the following product-specific abbreviations.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Terminal or valve terminal</td>
<td>Programmable valve terminal with control block SF 3</td>
</tr>
<tr>
<td>Node</td>
<td>Control block SF 3</td>
</tr>
<tr>
<td>sub-base</td>
<td>Pneumatic sub-base for valves</td>
</tr>
<tr>
<td>S sub-base</td>
<td>For two single valves type 03 (MIDI/MAXI)</td>
</tr>
<tr>
<td>D sub-base</td>
<td>For double pilot valves or mid position valve type 03 (MIDI/MAXI)</td>
</tr>
<tr>
<td>ISO sub-base</td>
<td>Sub-base for 4,8 or 12 valve type 05 (ISO 5599/I, size 1 or 2) or sub-base with intermediate solenoid plate MUH for one valve type 04-B; (ISO 5599/II, size 1, 2 or 3)</td>
</tr>
<tr>
<td>I/O</td>
<td>Input</td>
</tr>
<tr>
<td>I/O</td>
<td>Output</td>
</tr>
<tr>
<td>I/O</td>
<td>Input and/or output</td>
</tr>
<tr>
<td>AS-i I/O</td>
<td>Input and/or output on AS-i bus system</td>
</tr>
<tr>
<td>P module</td>
<td>Pneumatic module, general</td>
</tr>
<tr>
<td>I/O module</td>
<td>Electric module with digital inputs or outputs</td>
</tr>
<tr>
<td>Analogue module</td>
<td>Electric module with analogue inputs or outputs</td>
</tr>
<tr>
<td>AS-i master</td>
<td>Electric module with AS-i master modules for up to 31 AS-i slaves</td>
</tr>
<tr>
<td>CP interface</td>
<td>Electric module with four connections for up to eight CP modules</td>
</tr>
<tr>
<td>PLC</td>
<td>Programmable logic controller; abbrev.: Controller (or PLC)</td>
</tr>
</tbody>
</table>

Fig. 1/1: Index of abbreviations

Technical terms are explained in the glossary in Appendix D.
Programmable valve terminals consist of different components:

- Node with control block SF 3
- In the case of type 02 valve terminals: sub-base corresponds to the number of valves
- In the case of type 03, type 04-B and type 05 valve terminals:
  - pneumatic and electrical modules

**PLEASE NOTE**

- Depending on your order, this manual may contain Chapters 2a or 2b.
- These chapters refer to the following types:  
  Chapter 2a - type 02  
  Chapter 2b - type 03  
- For most drawings in this manual, one standard type 03 valve terminal is used with four pneumatic sub-bases and I/O modules.

![Fig. 1/2: Standard equipment for the drawing](image)

**Service**

In the event of technical problems, please contact your local Festo Service.
1.2 SYSTEM SUMMARY
1.2 SYSTEM SUMMARY

System structure .......................... 1-11
Valve terminals in standalone
operating mode ............................ 1-12
Valve terminal in master operating mode 1-14
Valve terminal in slave operating mode 1-16
Operating equipment with ABG (DCU) 1-18
Emulator as commissioning support 1-19
Programming software FST 200 1-20
Ladder diagram (LDR) ...................... 1-20
Statement list (STL) ....................... 1-20
Convert ..................................... 1-21
Equipment for program generation 1-21
Valve terminals with analogue modules 1-22
Valve terminals with AS-i master 1-24
Valve terminal with CP interface 1-26
1.2 SYSTEM SUMMARY

System structure

The integrated control is a powerful mini control system with field bus module. There are the following options for operating these units:

- Pushbutton panel (START/STOP)
- Keyboard with text display (ABGx)
- PC with FST software

For programming you require a PC and the FST 200 Festo programming software. The PC must be connected to the diagnostic interface of the valve terminal.

The following programming languages can be used:

- LDR (ladder diagram)
- STL (statement list)

The following diagram provides a system summary:

Fig. 1/3: System structure with SF 3
Valve terminals in standalone operating mode

Valve terminal with SF 3 for controlling a standalone machine.

Small standalone machines or sections of the system can be controlled independently using a valve terminal with SF 3. Similarly, standalone subsystems with separate functionality can be implemented as part of a larger unit.

The control program can be implemented in a very simple manner using FST 200 programming software. This provides a flexible solution to control problems.

Using programmable valve terminals offers the following advantages:
- No need for time-consuming wiring and installation.
- Less fitting space due to smaller and more compact design.
- Pre-assembled and inspected pneumatic and electrical modules.
- Simple installation when connecting up pneumatic and electrical actuators and sensors with additional modules.
- Maintenance-free operation due to built-in EEPROM.
Fig. 1/4: System summary: Valve terminals in standalone operating mode
Valve terminal in master operating mode

Valve terminal with SF 3 and a field bus extension to control equipment.

In addition to valves and local inputs and outputs, other field bus stations can be connected to programmable valve terminals with an integrated field bus module. This makes it possible to solve automation tasks by using a large number of pneumatic components, electric sensors and other actuators. Similarly, standalone subsystems with separate functionality can be implemented as part of a larger unit.

The use of programmable valve terminals with field buses offers the following advantages:

- No restriction on option of using SF 3 in stand-alone operating mode.
- Field bus is built-in as standard.
- Simple to install and flexible to extend with up to 31 field bus stations.
- Simple configuration of field bus station with a configuration program integrated in the FST 200.
Fig. 1/5: System summary: Valve terminal in master operating mode

- Display + Operation
  - Master valve terminal with SF 3

- Programming + Configuration

- Slave (passive) valve terminal type 02
- Slave (passive) valve terminal type 03
- Slave (passive) valve terminal type 05
- Slave (passive) field bus I/O FB202

Max. 31 field bus slaves
Valve terminal in slave operating mode

A valve terminal with SF 3 connected to a field bus as a slave controls the function units of the equipment itself and communicates via the field bus with a higher-order master.

When the programmable valve terminal is used as a slave, the mechanical structure of a machine or a unit can be represented by distribution of the PLCs (programmable logic controllers). All stand-alone modules or function units then have their own control program with which subranges can be controlled.

The use of programmable valve terminals as slaves offers the following advantages:
- No restriction on option of using SF 3 in standalone operating mode.
- Modular construction of the unit/machine is possible.
- Function modules on unit or machine can be linked together individually.
- Convenient partial commissioning possible.
- High availability of the system due to standalone subranges.
- Local display and operation possible.
Fig. 1/6: System summary: Valve terminal in slave operating mode
Operating equipment with ABG (DCU)

In addition to the actual control unit and the actuators and sensors, machines and systems also need equipment in order that they can be operated and so that operating states can be displayed. The following variations are possible for programmable valve terminals with SF 3:

- Conventional operation by switch or pushbutton and display from indicator light
- Display and control units (DCU)

Display and control units have the following advantages:

- Display of system status with message texts
- Cost-effective control with function keys
- Simple project engineering of ABG/DCU using the display editor integrated in the FST 200.
- Commissioning and testing ABG/DCUs using software emulation

The conventional control level is connected to the digital I/O modules. The display and control units are connected to the valve terminal using the DIAG-interface.
Emulator as commissioning support

During the commissioning process, the PC is connected to the programmable valve terminal by the DIAG interface in order to load the programs and to track the program flow.

Operation using the DCUs is then interrupted. To ensure that the entries required by the DCU can still be made despite this interruption, programs have been devised which emulate the operating and display units. Depending on the program used, the corresponding emulator is loaded in the PC’s memory as a background program. When keystroke combination ALT F10 is entered, the emulator is called up and the DCU appears on the screen.

![Fig. 1/8: Emulator for DCU (example)](image-url)
Programming software FST 200

The software package FST 200 offers a system-oriented programming interface for the programmable valve terminal. This contains all the software modules required for programming, testing and commissioning the SF 3 and SB 202 / SF 202.

Programming of the control block can be illustrated in the following ways:

Ladder diagram (LDR)

```
I2.3  I4.3  O1.6
I3.2
```

Statement list (STL)

```
STEP mark
IF  I 2.3
  AND  I 4.3
  ORI 3.2
THEN SET  O 1.6
```
With FST 200 programming software, it is possible to handle all aspects of project engineering of valve terminals with built-in Festo PLC. This reduces the time required to become familiar with the system. Project data can be transferred and there is a saving in engineering costs.

The programming software makes the following items possible:

- Changing over software from FST 202 to FST 203
- Programming in ladder diagram (LDR) and statement list (STL)
- Projection for display and control unit *)
- Emulator for the display and control unit *)
- Configuration of field bus module
- Configuration and commissioning of terminal with AS-i Master

*) Description in FST 200 manual (V3.2) - part 2

Convert

You can access the software Convert under Access program in the software package FST 200. With Convert you can convert old SB 202/SF 202 projects for the SF 3 (converting the operands). Instructions on the Functioning Method can be found in Online help (key F9).

Equipment for program generation

The following equipment can be used for program generation for the SF 3 control block:

- PC/laptop with programming software FST 200
Valve terminals with analogue modules

In many automation tasks, in addition to digital inputs and outputs, analogue signals are also required. For these problems, special analogue modules can be provided for the programmable valve terminals with SF 3 (type 03...05). These enable the terminals to process analogue input signals such as setpoint specification and actual value responses as well as analogue outputs for the actuation of final control elements.

These analogue modules exist in the following versions:

- **Universal**
  - (optional with current or voltage interface)
  - current-loop interface 4...20 mA, angular frequency 116 Hz
  - voltage-loop interface 0...10 V, angular frequency 116 Hz

- **Proportional**
  - (adapted for actuation of the proportional valve; 4...20 mA, angular frequency 100 Hz)

Using a programmable valve terminal with analogue processing offers the following advantages:

- preliminary processing of analogue signals directly on the work process
- easy installation for connection of proportional valves
- short leads, therefore less interference
Fig. 1/9: System summary: Valve terminals with analogue modules
Valve terminals with AS-i master

In many machines or systems, the pneumatic final control elements are remotely distributed or arranged in small groups. Using the actuator sensor interface, these digital final control elements can be easily connected in the programmable valve terminal with SF 3 (type 03...05).

Using programmable valve terminals with AS-i master has the following advantages:
- Options for SF 3 in standalone operating mode are unaffected.
- Easy-to-install connection of pneumatic final control elements and sensors in distributed systems.
- Scope for flexible extension.
- Pneumatic installation adapts itself to mechanical structure of machine or system.
- Tube lengths are shortened.
- Simple configuration of AS-i network with addressing device and FST 200 programming software (AS-i bus configurator).
Fig. 1/10: System summary: Valve terminals with AS-i master
Valve terminal with CP interface

The Festo concept for CP valve terminals gives you the scope, by virtue of a modular structure, to integrate valve terminals and I/O modules in your machines and systems in an optimum manner.

Advantages:
- Independent CP system with up to 64I and 64O
- Compact CP valve terminals
- Assembly close to the cylinders
- Short air supply lines
- Short pressurising and venting times
- Small valves possible (cost-optimised)
- Remote CP I/O modules
1.3 SYSTEM LIMITS AND PLANNING ASPECTS
CONTENTS

1.3 SYSTEM LIMITS AND PLANNING ASPECTS

System limits . . . . . . . . . . . . . . . . . . . . . . . . . . 1-29
Planning aspects of valve terminals types 03...05 . . . . . . . . 1-31
Planning aspect 1
Common voltage supply for all outputs . . . 1-32
Planning aspect 2
Separate voltage supply for single high-current output module . . . . . . . . . 1-34
Planning aspect 3
Possible combinations of I/O modules . . . 1-36
1.3 System limits and planning aspects

1.3 SYSTEM LIMITS AND PLANNING ASPECTS

System limits

In theory, a fully-fitted complete system with programmable valve terminal can take the following form:

- SF 3 as master with up to 128 local inputs and 128 local outputs.
- 31 field bus stations as intelligent slaves or remotely distributed equipment (e.g. field bus valve terminal) in each case with max. 128 inputs and 128 outputs (max. 1048 I/Os on field bus).
- AS-i master with 31 AS-i slaves (max. 128 AS-i inputs and 128 AS-i outputs).
- Analogue and digital input and output modules (max. total of 12 modules in any combination).
- Display and control unit (DCU) on SF 3.
In practice, the above number of devices and/or inputs and outputs is restricted by the size of the user memory (128 kBytes) and the cycle time (1 ms/1 k instruction).

The number of controllable inputs and outputs is always dependent on the complexity of the control problem and the use of special peripheral equipment such as display and control units which occupy additional user memory.

Guideline value for a rapid "average" application: approx. 300 I/Os (including DCU).

**PLEASE NOTE**
The number of controllable inputs and outputs is always dependent on the complexity of the control problem and the use of special peripheral equipment. For any larger applications, the memory requirement should be estimated on an individual basis.
Planning aspects of valve terminals types 03...05

This chapter contains some advice on the following planning aspects for modular valve terminals:

**PLEASE NOTE**
Check your EMERGENCY STOP circuitry to ascertain which measures are necessary in order to bring the machine/system into a safe state in the event of an emergency stop (e.g. switching off the operating voltage for the valves and the output modules, switching off the compressed air).

- **Planning aspect 1**
  Common voltage supply to all outputs; i.e. the EMERGENCY STOP function for all outputs is implemented via pin 2 of the node/adapter block (valve and electric modules).

- **Planning aspect 2**
  Separate voltage supplies for individual high-current output modules; i.e. the auxiliary power supply in combination with the high-current outputs enables them to operate independently of the EMERGENCY STOP function.

- **Planning aspect 3**
  Possible combinations of I/O modules.
  Instructions for planning the sequence for assembly of I/O modules and for combining these modules on a valve terminal.
Planning aspect 1
Common voltage supply for all outputs

This involves all components of the valve terminal being supplied with a 24 V power supply via pins 1 and 2 on the node/adapter blocks.

- Pin 1: 24 V (± 25 %), max. 2.2 A operating voltage for the internal electronics of the node and all I/O modules. 24 V DC power supply to all inputs/sensor (PNP and NPN).
- Pin 2: 24 V (± 10 %), max. 10 A operating voltage for the valve and electrical outputs. Please note that when the valves are switched off (e.g. during EMERGENCY STOP) all the electrical outputs are also switched off.
Advantages:
- Easy installation – simply involves connecting up a power supply unit.
- All outputs on the valve terminal are switched off at the hardware level when EMERGENCY STOP is activated (failsafe).

Disadvantages:
- It is not possible to implement a differentiated set of EMERGENCY STOP characteristics with which certain electric outputs remain active.

Fig. 1/12: Common voltage supply to all outputs (example)
Planning aspect 2
Separate voltage supply for single high-current output modules

This involves fitting at least one module for a 24 V auxiliary power supply to the left side of the node. This module provides electrical insulation of the electric I/O side. The high-current output modules are fitted to the left side of the auxiliary power supply unit and are only supplied from their 24 V power source. A mixture of negatively and positively switching high-current output modules can be fitted.

Voltage supply via node:
• Pin 1: 24 V (± 25 %), max. 2.2 A operating voltage for the internal electronics of the node and all I/O modules. 24 V DC power supply to all inputs/sensors (PNP and NPN).
• Pin 2: 24 V (± 10 %), max. 10 A operating voltage for the valves and only for the electric outputs (PNP; 0.5 A). Note that, when the valves are switched off (e.g. during EMERGENCY STOP), only these electric outputs (PNP; 0.5 A) are actually switched off.

Voltage supply from auxiliary power supply:
• Terminal 2: 24 V (± 25 %), max. 25 A operating voltage for all high-current outputs (PNP or NPN, 2 A) mounted to the left of the relevant auxiliary power supply (power supply ends with last high-current output module).

Note:
Due to the auxiliary power supply, the operating voltage of the high-current outputs is entirely separate from pin 2 on the node.
"Normal" output modules (PNP; 0.5 A) mounted to the left of the last high-voltage output modules are still supplied with power from pin 2 of the node.
Advantages:
• Additional 25 A per auxiliary power supply are available for loads with high current consumption (e.g. hydraulic valves).
• Modules with four high-current outputs (HC-OUTPUT, each with optional 2 A PNP or NPN): these can obtain their current from the auxiliary power supply on the right side.
• Electric high-current outputs to the left of the auxiliary power supply can remain active when the EMERGENCY STOP is active.
• Several auxiliary power supplies per terminal are possible.

Disadvantages:
• An auxiliary power supply unit occupies the space of one I/O module (max. 12 modules).
If the high-current outputs to the left of the auxiliary power supply are also switched off during EMERGENCY STOP, it may be necessary to provide additional appropriate installations.

Fig. 1/13: Separate voltage supply to all outputs (example)
Planning aspect 3
Possible combinations of I/O modules

A wide range of universal and special I/O modules are available for modular valve terminals and these can be combined in virtually any sequence. Take due account during the planning stage, or when converting these terminals, of permitted combinations. This point always applies: maximum of 12 electric modules per terminal.

The following applies to individual electric modules:

• digital I/O modules (4I, 8I and 4O PNP or NPN) in any combination and at every position (5).
• analogue I/O modules (PROP; UNIVERSAL) in any combination and at every position (4).
• Auxiliary power supply units: always fitted to every position (3).
• Modules with high-voltage outputs (HC-OUTPUT, PNP or NPN) only to the left of an auxiliary supply, but in any combination (2).
• The CP interface must always be fitted on the right side (6).
• The AS-i master must always be fitted on the left side (1).
Additional information on the individual modules will be found in:

- Chapter 2 for the I/O module
- "Supplementary description of the I/O modules" insert in Chapter 2 for the additional power supply and the high current output modules
- Chapter 5 for the analogue I/O modules
- Chapter 6 for the AS-i master
- Chapter 7 for the CP interface

Fig. 1/14: Possible combinations of electrical I/O modules (example)
Programmable valve terminals
with
Control block SF 3

Part 2b: System description
of type 03 valve terminal
Contents

2.1 COMPONENTS .......................... 2-3
  Structure of type 03 valve terminal  ......... 2-3

2.2 ASSEMBLY ............................. 2-9
  Mounting the components .................. 2-9
  Input/output modules .................... 2-10
  End plates ............................. 2-12
  Top-hat rail clamping unit ............... 2-14
  Mounting the valve terminal .............. 2-15

2.3 ELECTRICAL CONNECTIONS ............. 2-21
  Connecting up the operating voltages ...... 2-22
  Calculating the current consumption ...... 2-24
  Earthing ................................ 2-27
  Connecting the diagnosis interface ...... 2-28
  Connecting the field bus interface ...... 2-30
  Connecting the inputs and outputs ...... 2-35
  DUO cable ................................ 2-36
  Designation of inputs and outputs ...... 2-37

2.4 ADDRESSING ............................. 2-41
  General remarks .......................... 2-41
  Establishing configuration data .......... 2-41
  Calculating number of local inputs/outputs 2-42
  Address assignment of valve terminal .......... 2-43
  Address assignment after extension/conversion 2-47

2.5 TECHNICAL DATA ....................... 2-51
2.1 COMPONENTS
2.1 COMPONENTS

Structure of type 03 valve terminal ........... 2-3
2.1 COMPONENTS

Structure of type 03 valve terminal

The type 03 valve terminal consists of individual modules. Different functions, connection, display and operating elements are assigned to each of the modules. The following illustration provides a summary:

<table>
<thead>
<tr>
<th>Digit</th>
<th>Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Node</td>
</tr>
</tbody>
</table>
| 2     | Electrical modules (input/output modules), equipped with  
        • digital inputs (modules with 4 or 8 inputs) PNP/NPN  
        • digital outputs (modules with 4 outputs) PNP  
        • supplementary modules |
| 3     | Left end plate, with bore for additional protective earth conductor connection |
| 4     | Pneum. MIDI, MAXI modules (connection blocks) equipped with S-valves:  
        • 5/2-solenoid valves  
        • 5/2-double pilot valves  
        • 5/3-mid-position valves (exhausted, pressioned, blocked)  
        • Blanking plates S = auxiliary pilot air |
| 5     | Pneumatic modules for auxiliary pressure supply |
| 6     | Right end plate  
        • with and without connections  
        • with and without regulator for limitation of pilot pressure |

*Fig. 2/1: Modules for type 03 valve terminal*
On the electrical modules, you will find the following connection and display elements:

<table>
<thead>
<tr>
<th>Digit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Output socket for an electrical output (PNP)</td>
</tr>
<tr>
<td>2</td>
<td>Yellow LED (status display on each output)</td>
</tr>
<tr>
<td>3</td>
<td>Red LED (error display on each output)</td>
</tr>
<tr>
<td>4</td>
<td>Input socket for <strong>one</strong> electrical output (PNP or NPN)</td>
</tr>
<tr>
<td>5</td>
<td>Green LED (status display on each input)</td>
</tr>
<tr>
<td>6</td>
<td>Input socket for <strong>two</strong> electrical inputs (PNP or NPN)</td>
</tr>
<tr>
<td>7</td>
<td>Two green LEDs (status display, one LED on each input)</td>
</tr>
<tr>
<td>8</td>
<td>Node with LEDs, field bus and diagnosis interface, more detailed</td>
</tr>
<tr>
<td></td>
<td>description in chapter on &quot;Electrical Connections&quot;</td>
</tr>
<tr>
<td>9</td>
<td>Right end plate</td>
</tr>
<tr>
<td>10</td>
<td>Fuse for inputs/sensors</td>
</tr>
<tr>
<td>11</td>
<td>Operating voltage connection</td>
</tr>
<tr>
<td>12</td>
<td>Supplementary modules</td>
</tr>
<tr>
<td></td>
<td>• analogue I/Os</td>
</tr>
<tr>
<td></td>
<td>• AS-i master</td>
</tr>
<tr>
<td></td>
<td>• auxiliary 24 V/25 A power supply</td>
</tr>
<tr>
<td></td>
<td>• high current outputs (PNP or NPN)</td>
</tr>
<tr>
<td>13</td>
<td>Left end plate</td>
</tr>
</tbody>
</table>

**Fig. 2/2: Display and connection elements for electrical modules**
On the components in the pneumatic type 03 MIDI-Module, you will find the connection, display and operating elements listed below.

<table>
<thead>
<tr>
<th>Digit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Nodes with LEDs, field bus interface and diagnosis interface,</td>
</tr>
<tr>
<td></td>
<td>more detailed description in Chapter &quot;Electrical Connections&quot;</td>
</tr>
<tr>
<td>2</td>
<td>Yellow LED (for each valve solenoid coil)</td>
</tr>
<tr>
<td>3</td>
<td>Manual override (for each valve solenoid coil)</td>
</tr>
<tr>
<td>4</td>
<td>Valve position in inscription field</td>
</tr>
<tr>
<td>5</td>
<td>Unused valve position with blanking plate</td>
</tr>
<tr>
<td>6</td>
<td>Common tubing connections</td>
</tr>
<tr>
<td>7</td>
<td>Regulator for limitation of pressure for auxiliary pilot air</td>
</tr>
<tr>
<td>8</td>
<td>Operating connections (2 per valve, piggy-back)</td>
</tr>
<tr>
<td>9</td>
<td>Fuse for inputs/sensors</td>
</tr>
<tr>
<td>10</td>
<td>Operating voltage connection</td>
</tr>
</tbody>
</table>

Fig. 2/3: Operating, display and connection elements for pneumatic MIDI modules
On components of pneumatic type 03 MAXI modules, you will find the following connection, display and operating elements.

<table>
<thead>
<tr>
<th>Digit</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Nodes with LEDs, field bus and diagnosis interface,</td>
</tr>
<tr>
<td></td>
<td>more detailed description in Chapter &quot;Electrical Connections&quot;</td>
</tr>
<tr>
<td>2</td>
<td>Yellow LED (for each valve solenoid coil)</td>
</tr>
<tr>
<td>3</td>
<td>Manual override (for each valve solenoid coil)</td>
</tr>
<tr>
<td>4</td>
<td>Valve position inscription field (identification plates)</td>
</tr>
<tr>
<td>5</td>
<td>Unused valve position with blanking plate</td>
</tr>
<tr>
<td>6</td>
<td>Common tubing connections</td>
</tr>
<tr>
<td>7</td>
<td>Operating connections (2 per valve, piggy-back)</td>
</tr>
<tr>
<td>8</td>
<td>Regulator for limitation of pressure for auxiliary pilot air</td>
</tr>
<tr>
<td>9</td>
<td>Common tubing connection</td>
</tr>
<tr>
<td>10</td>
<td>Exhaust air connections</td>
</tr>
</tbody>
</table>

Fig. 2/4: Operating, display and connection elements for pneumatic MAXI modules
2.2 ASSEMBLY
Contents

2.2 ASSEMBLY

Mounting the components . . . . . . . . . . . . . . . . 2-9
Input/output modules
  Dismantling (also refer to following Fig.) . . 2-10
  Mounting (also refer to following Fig.) . . . 2-11
End plates . . . . . . . . . . . . . . . . . . . . . . . . . . . 2-12
Top-hat rail clamping unit . . . . . . . . . . . . . 2-14
Mounting the valve terminal
  Wall mounting. . . . . . . . . . . . . . . . . . . . . 2-15
  Top-hat rail mounting (type 03) . . . . . . 2-16
2.2 ASSEMBLY

Mounting the components

**WARNING**
Switch off the following items before installation and maintenance operations:
- Compressed air supply.
- Operating voltage supply for electronics (Pin 1 of operating voltage connection)
- Operating voltage supply for outputs/valves (Pin 2 of operating voltage connection)

You thereby prevent:
- Uncontrollable movements of unfastened flexible tubes.
- Undesirable movements of connected actuator units.
- Undefined switching modes of the electronics.

**CAUTION**
- The components of the valve terminal contain electrostatically vulnerable components.
- For this reason, avoid touching any of the electrical contact surfaces on the side plug connectors on these components.
- Pay due attention to handling specifications for electrostatically vulnerable components.

In this way you can prevent the valve terminal components from being damaged.
PLEASE NOTE
Treat all modules and components on the valve terminals with care.
Pay particular attention to the following:
• Threaded connectors must not be distorted or subject to mechanical tension.
• Screws must be inserted correctly (otherwise threads can be damaged).
• Comply with specified tightening torques. Avoid offset between the modules (IP65).
• Connection faces must be clean (to prevent leakage and contact faults).
• The contacts on the valve solenoid coils must not be bent (not resistant to reciprocal loading, i.e. break when being bent back).
With modules and components ordered at a later date, please comply with assembly instructions in product pack.

Input/output modules
To expand or convert the valve terminal, it is necessary to dismantle the screw-mounted terminal.

Dismantling (also refer to following Fig.):
• Completely remove screws from the module. The modules are now only held together by the electrical plug connection.
• Carefully remove modules from electrical plug connections without tilting them.
• Replace the damaged seals.
**Mounting** (also refer to following Fig.):

- **PLEASE NOTE**
  - Wherever possible, place subsequently ordered modules behind the last module before the end plate.
  - Always place the AS-i master directly against the left end plate.
  - Always place the CP interface directly at a node.
  - Do not fit more than 12 electrical modules (digital inputs and outputs, analogue modules and AS-i master).

Mount the modules as follows:
- Insert a new seal on contact surface facing right side of node.
- Mount in accordance with following Fig.

![Mounting the electrical modules](image)

**Fig. 2/5: Mounting the electrical modules**
End plates

You require a right and a left end plate to form a mechanical end to the terminal. These end plates perform the following functions:

- Ensure protection class IP65 is provided.
- Contain connections and contacts for the earthing.
- Contain bores for wall mounting and the top-hat rail clamping unit.

Three versions of the right end plate are available:

- MIDI:
  with common tubing connections for compressed air supply of pneumatic modules and built-in regulator for the auxiliary pilot air (5 bar),
- MIDI / MAXI:
  with common tubing connections for the compressed air supply of the pneumatic modules without regulator.
- MAXI:
  without common tubing connections

CAUTION
Prior to reassembly, earth the right end plate. In this way you can avoid faults due to electromagnetic influences.
Earth the end plates as follows:

- **Right end plate:**
  To earth the right end plate, fit preassembled cable on inside to the corresponding contacts on the pneumatic modules and/or the node (refer to following Fig.).

- **Left end plate:**
  The left end plate is connected conductively to the other components by means of preassembled spring contacts.

**Note:**
For instructions on earthing the entire valve terminal, please consult the chapter "Electrical connections".

Following Fig. illustrates the mounting of both end plates:

*Fig. 2/6: Earthing and mounting the end plates*
Top-hat rail clamping unit

You will require the top-hat rail clamping unit, if the terminal is to be fitted to a hat rail (support rail as per EN 50022). The top-hat rail clamping unit is secured to the rear side of the end plates as shown in the following Fig.

Before mounting the unit, check that:
- the bonding surfaces are clean (cleaned with petroleum spirit).

After mounting, ensure that:
- the flat head screws are firmly tightened (digit 6).
- the lever is secured with retaining screw (digit 7).

Fig. 2/7: Mounting the top-hat rail clamping unit

1 Rubber foot, self-adhesive
2 Clamping element
3 Left lever
4 Right lever
5 O-ring
6 Flat head screw
7 Retaining screw
Mounting the valve terminal

Wall mounting

CAUTION
On long valve terminals, place additional mounting brackets about every 200 mm. This avoids any risk of:
- overloading the retaining lugs on the end plates
- sagging the valve terminal
- natural resonances

Proceed as follows:
- Establish the weight of your valve terminal (weigh or calculate). General rule:

<table>
<thead>
<tr>
<th></th>
<th>MIDI</th>
<th>MAXI</th>
</tr>
</thead>
<tbody>
<tr>
<td>per pneumatic module</td>
<td>800 g</td>
<td>1200 g</td>
</tr>
<tr>
<td>per node</td>
<td>1000 g</td>
<td>1000 g</td>
</tr>
<tr>
<td>per electronics module</td>
<td>400 g</td>
<td>400 g</td>
</tr>
</tbody>
</table>

- Ensure that the mounting surface is capable of bearing this weight.
- Secure the terminal with four M6 screws (refer to following Fig., fitting position as desired). Use washers where applicable.

Fig. 2/8: Wall mounting of valve terminal
Top-hat rail mounting (type 03)

The terminal is suitable for mounting on a top-hat rail (support rail as per EN 50022). For this purpose, you will find a guide groove on the back of every module which can be attached to the top-hat rail.

**CAUTION**
- Top-hat rail mounting without a top-hat rail clamping unit is not permitted.
- With angled fitting positions or oscillating loads, also secure top-hat rail clamping unit to prevent it from slipping and with screws provided (7) to prevent it from releasing/opening accidentally.

**PLEASE NOTE**
- With horizontal fitting position and non-moving load, the top-hat rail clamping unit can be secured adequately without screws (7).
- If your valve terminal does not have a top-hat rail clamping unit, this can be ordered and mounted at a later date.
- The use of MIDI or MAXI clamping units depends on the end plates provided (MIDI/MAXI).

Proceed as follows:
- Establish the weight of your valve terminal (weigh or calculate). General rule:

<table>
<thead>
<tr>
<th></th>
<th>MIDI</th>
<th>MAXI</th>
</tr>
</thead>
<tbody>
<tr>
<td>per pneumatic module</td>
<td>800 g</td>
<td>1200 g</td>
</tr>
<tr>
<td>per node</td>
<td>1000 g</td>
<td>1000 g</td>
</tr>
<tr>
<td>per electronics module</td>
<td>400 g</td>
<td>400 g</td>
</tr>
</tbody>
</table>
• Ensure that the mounting surface is capable of bearing this weight.
• Mount a top-hat rail (support rail EN 50022 - 35x15; width 35 mm, height 15 mm).
• Secure the top-hat rail at least every 100 mm to the mounting surface.
• With factory-fitted top-hat rail clamping units, unlock the clamping unit.
• Attach the terminal to the top-hat rail. Secure both sides of the terminal with the top-hat rail clamping unit to prevent it from tilting or slipping (refer to following Fig.).
• In the event of oscillating loads or angled fitting position, secure the top-hat rail clamping unit with two screws (Fig. digit 7) to prevent the unit from releasing or opening accidentally.

Fig. 2/9: Assembling the valve terminal on a top-hat rail
2.3 ELECTRICAL CONNECTIONS
2.3 ELECTRICAL CONNECTIONS ............ 2-21

Connecting up the operating voltages .... 2-22
Calculating the current consumption ..... 2-24
Earthing .................................... 2-27
Connecting the diagnosis interface ..... 2-28
Connecting the field bus interface ..... 2-30
Connecting the inputs and outputs ..... 2-35
DUO cable .................................. 2-36
Designation of inputs and outputs ...... 2-37
2.3 ELECTRICAL CONNECTIONS

**WARNING**
Prior to any installation and maintenance work, switch off the following:
- Compressed air supply.
- Operating voltage supply to electronics (Pin 1).
- Operating voltage supply to outputs/valves (Pin 2).

By doing this, you prevent:
- Uncontrolled movements of unfastened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.

The following connection and display elements are located on the cover of the node:

![Diagram of connection elements](image)

**Fig. 2/10: Cover on control block**

**PLEASE NOTE**
- Settings (field bus baud rate, address and terminating resistor) are performed by the software.
- It is not necessary to open the cover.
Connecting the operating voltages

WARNING

• Use only PELV circuits as per IEC/DIN EN 60204-1 (Protective Extra-Low Voltage, PELV) for the electrical supply. Consider also the general requirements for PELV circuits in accordance with IEC/DIN EN 60204-1.

• Use power supplies which guarantee reliable electrical isolation of the operating voltage as per IEC/DIN EN 60204-1.

By the use of PELV circuits, protection against electric shock (protection against direct and indirect contact) is guaranteed in accordance with IEC/EN 60204-1 (Electrical equipment for machines, General requirements).

CAUTION
The operating voltage supply of outputs and valves (Pin 2) must be fused externally with maximum 10 A. Using external fusing, you avoid functional damage to valve terminals in the event of a short circuit.
Before you start connecting up operating voltages to nodes, please note the following points:

- Calculate the complete current consumption in accordance with the table on the next page and select a suitable mains unit as well as suitable cable cross sections (also refer to Appendix A).

- Avoid large distances between mains unit and valve terminal. Calculate permitted distance where permissible in accordance with Appendix A.

General rule:

<table>
<thead>
<tr>
<th>Consumption</th>
<th>Line cross section</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valve terminal $V_0 = 24,\text{V}$</td>
<td>$1.5,\text{mm}^2$ (AWG 16)</td>
<td>$\leq 8,\text{m}$</td>
</tr>
<tr>
<td>Pin 1 $= 2.2,\text{A}$</td>
<td>$2.5,\text{mm}^2$ (AWG 14)</td>
<td>$\leq 14,\text{m}$</td>
</tr>
<tr>
<td>Pin 2 $= 10,\text{A}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**PLEASE NOTE**

If you use an additional power supply, other values and tables apply. Refer to "Supplementary description of the I/O modules" at the end of this Chapter.
Calculating the current consumption

The following table is used for calculating overall current consumption. Take due account of the different current consumptions of MIDI and MAXI valves. The specified values have been rounded up.

<table>
<thead>
<tr>
<th>Current consumption of electronics nodes and inputs (Pin 1, 24 V ± 25 %)</th>
<th>0.200 A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td></td>
</tr>
<tr>
<td>Number of sensor inputs assigned simultaneously</td>
<td>( \sum ) A ( \times ) 0.010 A</td>
</tr>
<tr>
<td>Power supply to sensors (refer to manufacturer's specifications)</td>
<td>( \sum ) A ( \times ) _____ A</td>
</tr>
<tr>
<td>Current consumption of electronics nodes and inputs (Pin 1) max. 2.2 A</td>
<td>= ( \sum ) A</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Current consumption of valves and outputs (Pin 2, 24 V ± 10 %)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of valve coils MIDI (energised at same time)</td>
<td>( \sum ) A ( \times ) 0.055 A</td>
</tr>
<tr>
<td>Number of valve coils MAXI (energised at same time)</td>
<td>( \sum ) A ( \times ) 0.095 A</td>
</tr>
<tr>
<td>Number of simultaneously energised electr. outputs:</td>
<td>( \sum ) A ( \times ) 0.010 A</td>
</tr>
<tr>
<td>Load current of simultaneously energised electr. outputs:</td>
<td>( \sum ) A ( \times ) _____ A</td>
</tr>
<tr>
<td>Current consumption of outputs (Pin 2) max. 10 A</td>
<td>= ( \sum ) A + ( \sum ) A</td>
</tr>
</tbody>
</table>

Total current of a type 03 valve terminal | = \( \sum \) A |

Fig. 2/11: Calculation of current consumption of a valve terminal
The connection for 24 V operating voltages is located on the lower left edge of the control block.

![Operating voltage connection and fusing for power supply to inputs](image)

*Fig. 2/12a: Location of connections for operating voltage*

Via this connection, the following components of the valve terminal are supplied separately with +24 V direct current (DC):

- Operating voltage for internal electronics, programmable controller and inputs on the input modules
  (Pin 1: DC + 24 V, tolerance ± 25 %).
- Operating voltage for outputs on the valves and outputs on the output modules
  (Pin 2: DC + 24 V, tolerance ± 10 %, external fuse required, max. 10 A).
The following figure illustrates the pin allocation of the operating voltage connection:

![Pin allocation for operating voltage connection](image)

**PLEASE NOTE**

Ensure that, where there is a common power supply for Pin 1 (electronics and inputs) and Pin 2 (outputs/valves), the tolerance of ± 10% must be observed for both circuits.

Check the 24 V operating voltage of the outputs while your system is operating. Ensure that the operating voltage of the outputs remains within permitted tolerance, even during full-power operation.

Recommendation:
Use a closed-loop mains power unit.
Earthing

The valve terminal has the following earth connections for potential compensation:

- On operating voltage connection (Pin 4, incoming contact).
- On the left end plate (M4 thread).

**PLEASE NOTE**

- Always connect the earth potential to Pin 4 of the operating voltage connection.
- Connect the earth connection of the left end plate with low impedance to the earth potential (short cable with large cross-section).
- By means of a low impedance connection, ensure that the housing of the valve terminal and the earth connection on Pin 4 share the same potential and that no compensating currents are flowing.
- Note the connection example illustrated in Appendix A.

By doing this, you prevent:

- Interference from electromagnetic sources.
Connecting the diagnosis interface

For programming the valve terminal, you require:
- PC or Laptop with serial RS 232 interface (V.24)
- Screened connection cable (e.g. Festo diagnostic cable KDI-SB202-BU25 or -BU9)

Connect up the diagnostic cable as follows:
- 4-pin connector to diagnostic interface of valve terminal (DIAG)
- 25-pin or 9-pin socket to the serial RS 232 interface of your PC/Laptop (COM1 or 2)
If you use a different connection cable, please note the following pin assignment:

**Pin assignment on 25-pin diagnostic cable**

- **4-pin rounded connector** for diagnosis interface on control block SF 3
- **25-pin bush on PC/laptop** (serial RS 232 interface/V.24)

```
RxD       1
TxD       2
GND      3
Screening 4
```

**Pin assignment on 9-pin diagnostic cable**

- **4-pin rounded connector** for diagnosis interface on control block SF 3
- **9-pin socket on PC/laptop** (serial RS 232 interface/V.24)

```
RxD       1
TxD       2
GND      3
Screening 4
```

**Pin assignment diagnostic interface on control block**

```
GND
Screening
Rx D
Tx D
```

*Fig. 2/13: Pin assignment on diagnostic cable and interface*
Connecting the field bus interface

**PLEASE NOTE**
Pay attention to the cable specifications. During data transmission, particularly at high speeds, signal reflection and signal damping occurs. Both can lead to errors in transmission.

Possible causes of reflection includes:
- Missing or incorrect terminating resistor.
- Branches.

Possible causes of signal damping can be:
- Transmission over long distances.
- Unsuitable cables.

The cables shown here can be used for data transmission over the distances stated and the appropriate connection socket (PG) is indicated in brackets in each case.

1. Universally suitable for distances of up to 1000 m (note voltage strength):
   
   Belden 984 (socket PG9):
   Paired, dual-screened cable (strand 24 AWG; 30 V)

2. Recommended cable types dependent on baud rate and distance (also refer to following table):

   A Coaxial paired cable (socket PG9)
   (Twinax; strand 20 AWG, 600 V):
   BICC H8106
   Belden 8227 or 1162A
   Helektra HE-TW-K 105 Order no. 1107304
2.3 Electrical connections

B Coaxial, screened paired cable
(socket PG7)
(strand 25 AWG; 300 V)
Belden 9271

C Paired cable with screening
(socket PG7)
(strand 20 AWG; 250 V)
Kabelmetal DUE4001 type no. 444101
Helektra HE-DUE 4CY AWG 20
Order no. 1109401

D Paired, screened control line
(socket PG13)
Belden 9860
Siemens L-2Y2YcY 2.1-1,
(1.5 mm², 900 V) 5kf40
Order no. V45551-F21-B5

<table>
<thead>
<tr>
<th>Baud rate [kBd]</th>
<th>Cable type for distances [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>375.00</td>
<td>AB</td>
</tr>
<tr>
<td>187.50</td>
<td>AB A</td>
</tr>
<tr>
<td>62.50</td>
<td>ABCD ACD ACD D</td>
</tr>
<tr>
<td>31.25</td>
<td>ABCD ACD ACD D</td>
</tr>
</tbody>
</table>

These cables can be obtained from the following German firms:
BICC Deutschland GmbH, Düsseldorferstr. 186, 41460 Neuss.
Belden Electronics GmbH, Fuggerstr. 2, 41468 Neuss.
Helektra GmbH, Boschweg 12-16, 12057 Berlin 44.
Kabelmetal electro GmbH, Schathofstr. 35; 90411 Nürnberg.
Siemens AG, UB NK, Kistlerhofstr. 170, 81379 München 70.
There are two field bus connectors on the node for connecting a programmable valve terminal to the field bus. One of these connections is for the supply wire and the other is intended as a continuation of the field bus wire. The signal wires on both connectors are internally connected.

This makes two connection variants possible:

• Running a field bus wire from terminal to terminal. For this, both field bus connectors are required.

• Connecting the field bus wire to a T-adapter. This only requires one field bus connector.

CAUTION

Branches (e.g. T-adapter) can cause signal reflection at high transmission speeds. This can give rise to "telegram faults" with brief "failure" of valves.

Recommendation:
– Maintain the maximum permitted distance of 15 cm between T-adapter and field bus connector to prevent signal reflections.
– For this, use a ready-made Festo T-adapter FB-TA.
2.3 Electrical connections

**Fig. 2/14a:** Connection variant for field bus (contact pickup)

**Fig. 2/14b:** Connection variant for field bus (T-adapter)
CAUTION
- Pay attention to the correct polarity when connecting up the field bus interface.
- Connect up the screening. This prevents interference from electromagnetic sources.

The following diagram illustrates the pin assignment of the field bus interface. Connect up the field bus wires to the terminals on the bus receptacle, as appropriate.

Fig. 2/15: Pin assignment of field bus interface

If the SF 3 is located at the end of a field bus line, a terminating resistor is required. This has already been integrated and can be activated with FST 200 in online operating mode (refer to your FST 200 manual).
Connecting the inputs and outputs

**WARNING**
Switch off the following items before installation and maintenance operations:
- compressed air supply,
- operating voltage supply for electronics (Pin 1),
- operating voltage supply for outputs/valves (Pin 2).

By doing this, you prevent:
- Uncontrollable movements of unfastened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.

**PLEASE NOTE**
Connection instructions for the supplementary modules (only in conjunction with types 03...05) can be found in the following manuals and chapters:
- input/output modules, high current outputs, multi-I/O module: "Supplementary description of the I/O modules"
- analogue I/Os: Chapter 5
- AS-i master: Chapter 6
- CP interface: Chapter 7 and CP system manual "Installation and commissioning"
DUO cable

The DUO cables are suitable for sensor connections with two inputs. The connectors on the sensor end are intended for M8. There are three different pairs of connector versions.

![Diagram of DUO cable setup](image)

**Fig. 2/16: DUO cables and extension cable for simple connection of sensors**
Designation of inputs and outputs

Use the designation plates to identify the I/Os. This enables you to obtain a better overview:
- During commissioning
- During maintenance
- In circuit diagrams
- During the programming.

Fig. 2/17: Support for identification plate on electrical inputs and outputs (terminal types 03...05)
2.4 ADDRESSING
2.4 ADDRESSING

General remarks ........................................... 2-41
Establishing configuration data ....................... 2-41
Calculating number of local inputs/outputs ........ 2-42
Address assignment of valve terminal ............... 2-43
  Basic rule 1 ............................................ 2-44
  Basic rule 2 ............................................ 2-45
  Basic rule 3 ............................................ 2-45
Address assignment after extension/conversion .... 2-47
2.4 ADDRESSING

General remarks

Before programming, you should compile an allocation list of all connected inputs and outputs. This allocation list simplifies the tasks of addressing and/or programming at a later date. Addressing of valve terminals must follow a precise procedure because the modular structure requires different data to be entered for each terminal. Note the specifications in the following sections.

Addressing of analogue I/O and AS-i modules is described in full in Chapters 5 and 6; addressing of CP interface: Chapter 7 and description of the CP system under "Installation and Start-up".

Establishing configuration data

The SF 3 control block can operate up to 128 local inputs and 128 local outputs with a different number of I/Os being assigned on each module. The following table indicates the required number of I/Os for each module:

<table>
<thead>
<tr>
<th>Type of module</th>
<th>Number of assigned I/Os *</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single manifold</td>
<td>20</td>
</tr>
<tr>
<td>Double manifold</td>
<td>40</td>
</tr>
<tr>
<td>Output module (4 digital outputs)</td>
<td>40</td>
</tr>
<tr>
<td>Input module (4 digital inputs)</td>
<td>4I</td>
</tr>
<tr>
<td>Input module (8 digital inputs)</td>
<td>8I</td>
</tr>
<tr>
<td>CP interface (digital I/Os)</td>
<td>64I 64O</td>
</tr>
</tbody>
</table>

*) The I/Os are assigned automatically within each terminal, regardless of whether an input or output is actually used.

Fig. 2/18: Number of assigned I/Os per module
Calculating number of local inputs/outputs

Copy the table for future calculations and calculate the number of inputs/outputs.

<table>
<thead>
<tr>
<th>Calculation of inputs/outputs on type 03 valve terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>INPUTS:</strong></td>
</tr>
<tr>
<td>1. Number of 4 input modules</td>
</tr>
<tr>
<td>_____ * 4</td>
</tr>
<tr>
<td>2. Number of 8 input modules</td>
</tr>
<tr>
<td>_____ * 8</td>
</tr>
<tr>
<td>3. CP interface (if present)</td>
</tr>
<tr>
<td>_____ * 8</td>
</tr>
<tr>
<td><strong>Total of assigned local inputs</strong></td>
</tr>
<tr>
<td>(software max. 128 / fitting without CP interface max. 96)</td>
</tr>
<tr>
<td><strong>OUTPUTS:</strong></td>
</tr>
<tr>
<td>4. Number of single connection blocks (MIDI + MAXI)</td>
</tr>
<tr>
<td>_____ * 2</td>
</tr>
<tr>
<td>5. Number of double connection blocks (MIDI + MAXI)</td>
</tr>
<tr>
<td>_____ * 4</td>
</tr>
<tr>
<td><strong>Sub-total of 4. + 5.</strong></td>
</tr>
<tr>
<td><strong>7. Number of electric output modules</strong></td>
</tr>
<tr>
<td>_____ * 4</td>
</tr>
<tr>
<td><strong>8. PC interface (if present)</strong></td>
</tr>
<tr>
<td>_____ * 4</td>
</tr>
<tr>
<td><strong>Total of assigned local outputs</strong></td>
</tr>
<tr>
<td>(software max. 128 / fitting without CP interface max. 74 + 2)</td>
</tr>
</tbody>
</table>

Fig. 2/19: Calculation of number of assigned type 03 inputs and outputs
Address assignment of valve terminal

The address assignment of outputs on a modular valve terminal depends on the configuration of the terminal. Account has to be taken of the following configuration variants:

- Valves and digital I/O-modules,
- Only valves,
- Only digital I/O-modules.

For address assignment of these configuration variants, the following basic rules apply.

The addressing of analogue I/O and AS-i modules is described in detail in chapters 5 and 6; the addressing of CP modules can be found in the CP manuals.

**PLEASE NOTE**
With FST 200 programming software, you can display the address assignments on the screen.

If two addresses are assigned to one valve location, the following applies:

- lower-value address → pilot solenoid 14
- higher-value address → pilot solenoid 12
Basic rule 1

With mixed equipment, address assignment takes account of:

• Valves,
• Digital I/O-modules.

1. Outputs:
   Address assignment of outputs is not dependent on the inputs.

1.1 Address assignment of valves
   • Address assigned in ascending serial order,
   • Counting method starting at node from **left to right**,  
   • Single connection blocks always occupy 2 addresses
   • Double connection blocks always occupy 4 addresses
   • A maximum of 26 valve solenoid coils can be addressed.

1.2 Rounding up to 4 bits: case distinction:
   a) If the number of valve addresses can be divided by 4 without leaving any remainder, continue with Point 1.3.
   b) If the number of valve addresses leaves a remainder when divided by four, the 4-bit-oriented addressing must be rounded up to four bits. The rounded up 2 bits in the address area cannot be used.

1.3 Address assignment of output modules:
   After the (rounded up) 4-bit addresses of the valves, the digital outputs can be addressed.
   • Address assignment in ascending numerical order.
   • Counting starting from the node from **right to left**.
   • The individual modules count from top to bottom.
   • Digital output modules always occupy 4 addresses.
2. Inputs:
   Address assignment of inputs is independent of assignment of outputs.

2.1 Address assignment of input modules:
   • Address assignment in ascending numerical order,
   • Counting starts at the node from **right to left**.
   • On individual modules from top to bottom,
   • 4-fold input modules occupy 4 addresses,
   • 8-fold input modules occupy 8 addresses.

**Basic rule 2**

If only valves are used, address assignment is always as described in basic rule 1.

**PLEASE NOTE**
   Maximum of 26 valve solenoid coils are addressable.
   Rounding up the last two digits on the valve end is not required.

**Basic rule 3**

If only electrical I/Os are used, address assignment is always as described in basic rule 1.

**PLEASE NOTE**
   • Counting mode: Counting the addresses starts immediately, from left of node.
   • Rounding up of last two digits on valve end is not required.
   • Maximum of 48 digital outputs or 96 inputs are addressable.
When the operating voltage is switched on, the valve terminal automatically recognises all pneumatic modules present (max. 13) and all I/O-modules (max. 12) and assigns the appropriate local addresses to them. If one valve position remains unoccupied (cover plate) or if an electrical input/output is not connected, the appropriate address is still assigned.

The following diagram illustrates the address assignment for configuration with valves, inputs and outputs:

Comments about the diagram:

- If single valves are mounted on input sub-bases, four addresses are assigned for valve solenoid coils; the next highest address then remains unused (refer to address 3).
- If unused valve locations are fitted with blanking plates, the addresses are nevertheless still assigned (refer to addresses 12, 13).
- Due to the 4-bit-oriented addressing of modular valve terminals, the address at the last valve location is rounded up to the full four bits (if the configuration does not already use a full four bits). This means that, under certain circumstances, two addresses cannot be used (refer to address 14, 15).
Address assignment after extension/conversion

One special feature of modular valve terminals is their flexibility. If the requirements for the machine vary, the configuration of the terminal can be modified to suit.

**CAUTION**

With retrospective extensions or conversions of the terminals, input/output addresses can become displaced. This applies in the following cases:

- when one or more pneumatic modules is/are added or removed at a later date,
- when a pneumatic module with single solenoid valves is replaced by a new module with double solenoid valves or vice-versa,
- when additional inputs/output modules are inserted between node and existing input/output modules,
- when existing 4-input modules are replaced by 8-input modules – or vice-versa.

**PLEASE NOTE**

When extending or converting the valve terminal, note the mechanical limits (12 I/O modules, 13 P modules) and therefore the limit of 96 local inputs or 74(+2) local outputs (without CP interface).
The following diagram illustrates an extension from the standard configuration shown on the previous figure, indicating the changes which occur to address assignment as a result.

Comments on the diagram:
Pressure supply modules and intermediate pressure supply modules do not occupy any addresses.

Fig. 2/21: Address assignment of one valve terminal after extension/ conversion
2.5 TECHNICAL DATA
2.5 TECHNICAL DATA

General ........................................ 2-51
Electromagnetic compatibility (EMC) ........ 2-51
Protection against electric shock ............ 2-51
SF 3 control block ............................. 2-52
Field bus ...................................... 2-52
Operating voltage for electronics
and inputs .................................... 2-53
Operating voltage for outputs
and valves ................................. 2-53
### 2.5 TECHNICAL DATA

#### General

<table>
<thead>
<tr>
<th>Protection class (acc. to DIN 40050)</th>
<th>IP65</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Temperature during</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Operation</td>
<td>- 5 °C...+ 50 °C</td>
</tr>
<tr>
<td>• Storage/transport</td>
<td>- 20 °C...+ 60 °C</td>
</tr>
</tbody>
</table>

| Chem. Resistance | Refer to Festo pneumatics catalogue (resistance table) |

<table>
<thead>
<tr>
<th>Oscillation (acc. to DIN/IEC 68 part 2-6 and IEC 721/part 2-3)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• Transport</td>
<td>3.5 mm path with 2...8 Hz</td>
</tr>
<tr>
<td></td>
<td>1 g acceleration with 8...25 Hz</td>
</tr>
<tr>
<td>• Operation</td>
<td>0.35 mm path with 25...57 Hz</td>
</tr>
<tr>
<td></td>
<td>5 g acceleration with 57...150 Hz and 1 g acceleration with 150...200 Hz</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shock (acc. to DIN (German standards) / IEC 68 part 2-27 and IEC 721)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>30 g with 11 ms duration</td>
</tr>
</tbody>
</table>

#### Electromagnetic compatibility (EMC)

<table>
<thead>
<tr>
<th>• Interference emitted</th>
<th>Tested as per DIN EN 61000-6-4 (industry)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Immunity against interference</td>
<td>Tested as per DIN EN 61000-6-2 (industry)</td>
</tr>
</tbody>
</table>

#### Protection against electric shock

<table>
<thead>
<tr>
<th>protection against direct and indirect contact as per IEC/DIN EN 60204-1</th>
<th>by means of PELV circuits (Protective Extra-Low Voltage)</th>
</tr>
</thead>
</table>

1) The component is intended for industrial use.
### SF 3 control block

<table>
<thead>
<tr>
<th>Diagnosis interface</th>
<th>RS 232, floating serial, asynchronous, full-duplex software-handshake (1 start bit, 8 data-bit, 1 stop bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design</td>
<td></td>
</tr>
<tr>
<td>Type of transmission</td>
<td></td>
</tr>
<tr>
<td>Synchronous method</td>
<td></td>
</tr>
<tr>
<td>Transmission rate</td>
<td>9600 Baud</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User memory</th>
<th>128 kByte</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAM</td>
<td>128 kByte</td>
</tr>
<tr>
<td>EEPROM</td>
<td>1000</td>
</tr>
<tr>
<td>Programming cycles</td>
<td>5 \times 10^3</td>
</tr>
<tr>
<td>Mains shutdowns</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Programmable inputs/outputs (max.)</th>
<th>128I + 128O</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local I/Os</td>
<td>128I + 128O</td>
</tr>
<tr>
<td>AS-i I/Os</td>
<td>1048 I/Os</td>
</tr>
<tr>
<td>Field bus I/Os</td>
<td>36I + 12O</td>
</tr>
<tr>
<td>Analogue I/Os</td>
<td></td>
</tr>
</tbody>
</table>

### Field bus

<table>
<thead>
<tr>
<th>Design</th>
<th>RS 485, floating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol</td>
<td>Festo field bus (Master/Slave)</td>
</tr>
<tr>
<td>Transmission rate</td>
<td>31.25 kBaud (4000 m)</td>
</tr>
<tr>
<td></td>
<td>62.5 kBaud (2000 m)</td>
</tr>
<tr>
<td></td>
<td>187.5 kBaud (1000 m)</td>
</tr>
<tr>
<td></td>
<td>375 kBaud (500 m)</td>
</tr>
<tr>
<td>Cable length (dependent on baud rate)</td>
<td>up to 4000 m</td>
</tr>
<tr>
<td>Cable type (depending on cable length and preset field bus rate)</td>
<td>Refer to cable selection Chapter 2.3</td>
</tr>
</tbody>
</table>
## Operating voltage for electronics and inputs

(Pin 1 – operating voltage connection)
- Nominal value (polarity-safe) DC 24 V
- Tolerance ± 25 % (DC 18 V...30 V)
- Residual ripple 4 Vpp
- Current consumption (at 24 V) 200 mA + total current consumption inputs
- Fuse protection of inputs/sensors internal 2 A, slow-blow

### Power consumption (P)
- Calculation \[ P[W] = (0.2 \text{ A} + \sum I_{\text{inputs}}) \times 24 \text{ V} \]

## Operating voltage for outputs and valves

(Pin 2 – Operating voltage connection)
- Nominal value (polarity-safe) DC 24 V, max. 10 A
- Tolerance ± 10 % (DC 21.6 V...26.4 V)
- Residual ripple 4 Vpp
- Current consumption (at 24 V) 10 mA + total electr. outputs + total current consumption of switched valve solenoid coils (e.g. with MIDI-valve 55 mA)

### Power consumption (P)
- Calculation \[ P[W] = (0.01 \text{ A} + \sum I_{\text{electr. outputs}} + \sum I_{\text{solenoid coil}}) \times 24 \text{ V} \]
Programmable valve terminals
with
control block SF 3

Chapter 3: System description
control block SF 3
Contents

3.1 COMMISSIONING

Requirements .......................... 3-3
Connecting the diagnostic cable ....... 3-3
Working with FST 200 ..................... 3-4
Loading programs ......................... 3-8
Programming EEPROM .................... 3-10
Reading out the control unit (Uploading) 3-12
Loading data into the EEPROM ......... 3-13
Boot mode ............................. 3-15
Automode ............................... 3-16

3.2 PROGRAMMING

Programming with FST 203 ............... 3-19
Function modules ......................... 3-19
Operators, command set .................. 3-20
Operands .................................. 3-23
Summary of address space ............... 3-26
3.3 PROGRAMMING TECHNIQUES

- Structuring ........................................... 3-31
- Hierarchical depth of programs and modules .................... 3-33
- Program processing .................................. 3-34
- Multi-tasking characteristics ............................. 3-36
- Process image ........................................ 3-37
- Cycle time ............................................ 3-38
- Measurement of program run time ......................... 3-38
- Program processing
  - Module call in LDR/STL without STEP .................. 3-39
  - Programs and modules in Statement List with STEP ...... 3-40
- Modules for text output ................................ 3-42
- Additional remanent data words .......................... 3-43
- Indexed processing ................................... 3-43
- Application range for the remanent data words .......... 3-44
- Application example - placing receptacles ............... 3-45
- Interrupt-controlled timers/counters ..................... 3-46
- Application range of interrupt-controlled counters .... 3-49
- Application range of interrupt-controlled timers ........ 3-51

3.4 DIAGNOSIS AND ERROR HANDLING

- Summary of diagnosis options .......................... 3-55
- Diagnosis on site .................................... 3-56
- Diagnosis by program .................................. 3-60
- Function modules ...................................... 3-60
- Diagnosis bytes ...................................... 3-60
- Error handling ....................................... 3-62
3.1 COMMISSIONING
3.1 COMMISSIONING

Requirements ........................................ 3-3
Connecting the diagnostic cable ................. 3-3
Working with FST 200. .............................. 3-4
Loading programs ................................. 3-8
Programming EEPROM ......................... 3-10
Reading out the control unit (Uploading) ... 3-12
Loading data into the EEPROM ................ 3-13
Boot mode ........................................... 3-15
Automode .......................................... 3-16
3.1 COMMISSIONING

Requirements

Here are the requirements for commissioning:

- The programmable valve terminal is completely connected up.
- Operating voltage is applied to the programmable valve terminal.
- Operator must be familiar with the Festo software FST 203.

Rapid, reliable commissioning can be performed using FESTO programming software FST 203. This must be installed on a personal computer which is then connected to the programmable valve terminal by the diagnostic cable.

Connecting the diagnostic cable

You require:

- PC or Laptop with serial RS 232 interface (V.24)
- Screened connecting cable
  (e.g. Festo diagnostic cable SB-202-BU25 or -BU9)

Connect the diagnostic cable as follows:

- 4-pin connector to the diagnostic interface on the valve terminal
- 25-pin or 9-pin socket to the serial RS 232 interface of your PC/Laptop (COM1 or 2)
Working with FST 200

Installation and operation of the software package is explained in detail in the relevant manual. Consult the manual if you wish to know more about Festo Software Tools (FST). The following sections only provide a brief description of the key steps in the commissioning process.

**PLEASE NOTE**

After starting FST 200, check the software mode. When programming control block SF 3, the header must indicate:

FST 203 V 3.2

If this is not the case, press F7 to switch the software mode to FST 203.

---

![Fig. 3/1: Main menu of programming software](image-url)
The FST 200 programming software contains the following functions (among others):

- Programming in statement list (STL)
- Programming in LDR (ladder diagram)
- Setting and resetting individual outputs/operands
- Switching between software modes

FST 203/FST 202C

Online operation

Select the operation mode for Online operation. The following entry screen now appears:

![Fig. 3/2: Menu for Online operation](image)

The entry screen shows the configuration of the control block and the memory capacity still available for storage.

The menu for the bit-by-bit on/off switching (set/reset) of outputs and other operands can be accessed with F1 key (indicator SF 3 Info).
Display SF 3 Info:
The status of the inputs/outputs is displayed in this menu in the following form:

![DIAGRAM]

**WARNING**
When the system is switched on, the outputs immediately respond to the entry on the screen. Ensure that there is no risk to man or machine when the outputs are switched on or off.

Fig. 3/3: Switching on/off outputs (Ax.x→Ox.x; Ex.x→Ix.x)
Switching the outputs on/off:

- Using the **TAB** key (→l), select bit-by-bit input.
- Select the appropriate output with the cursor key or with the mouse.
- Use the **F1** to **F3** keys to set, reset or toggle the appropriate output.
Loading programs

Programs for the programmable valve terminal are grouped in FST 203 to form a project and are loaded into the terminal via the RS 232 interface. Individual steps are illustrated in the following section and an accurate description is provided in the manual of the FST 200.

PLEASE NOTE
Reaction of the programmable valve terminal when loading project/program:
The programmable valve terminal switches off all outputs and stops the program processing after the command "Load project/program".

Procedure:
• Using the cursor keys or the mouse in "Project management", activate the "Load project" function.
• In the next window, select the individual programs or modules which are to be loaded into the programmable valve terminal.
• Using the F1 key, start loading the programs.
If new programs have been generated or altered, these are first translated into machine code. Once all programs are available in machine code, the FST 203 loads them into the valve terminal. Once the loading process is completely finished, the message "Loading was successful" appears.

---

**Fig. 3/4: Loading projects and selecting programs**
Programming EEPROM

General remarks:
Once a program has been loaded and tested, you can store it in non-volatile form in EEPROM and adjust boot mode to EEPROM.

- EEPROM boot mode:
  In EEPROM boot mode the following operands are remanent, i.e. these operand values are saved in the memory in the event of a power failure and are available again when power is restored (POWER ON):
  Flag words FW0-FW31
  Flags F0.0-F31.15
  Counter words CW0-CW31
  Counter preselects CP0-CP31
  Counters C0-C31
  Timer preselects TP0-TP31
  Registers R0-R99

- Setting EEPROM boot mode:
  • Select "EEPROM programming" function from utility program and start the programming process with F4.
  • With F3, set the boot mode to EEPROM
    Next time operating voltage is switched on (POWER ON), the programs are transferred from EEPROM to RAM.
RAM boot mode:
In RAM boot mode, the data available in RAM are valid at POWER ON. In the event of a mains power failure, there is no way of ensuring that data are retained - even if this failure only lasts for a few seconds. You must assume that data will be lost in these cases.

PLEASE NOTE
The EEPROM is usually not programmed until commissioning has been completed. Operate in RAM boot mode during commissioning. Before programming the EEPROM, you must reading out the control unit. (Upload).
Reading out the control unit (Uploading)

Procedure:

- Using the cursor keys or the mouse, activate the function **Upload** in the menu "Utilities".

At this point, the FST 203 reads out the entire program, including directory, from the control unit and stores this data as a file.
Loading data into the EEPROM

The files read from memory can be loaded into the EEPROM.

Procedure:
- Activate the **EEPROM programming** function with the cursor key or with the mouse in the menu “Utilities”.

*Fig. 3/7: EEPROM programming*
At this point, the EEPROM programmer responds with the next input mask.

Procedure:
- Press key **F4** to start the programming process.

*Fig. 3/8: Entry mask for EEPROM programmer*
Boot mode

After the EEPROM has been programmed, it is normally desirable when the system is next booted to transfer this user program from EEPROM to RAM (EEPROM boot mode).

Procedure:
• Select "Boot mode" with F3 key.
A security enquiry appears in FST 200.
Answer with Yes

PLEASE NOTE
The boot mode should be set to EEPROM. In RAM boot mode, there is no way of ensuring that data are retained (programs/operands) in the event of a mains power failure, even if this failure only lasts for a few seconds. You must assume that data will be lost in these cases.
Automode

As a last step in commissioning, the operating mode "Automode ON" can be set. With "Automode ON", the programmable valve terminal automatically starts program processing when the power is switched on. The program with the lowest number (usually program 0) is started.

Procedure:
- Select "Online operation" menu
- Use the F5 key to select the Automode operating mode
- Answer the question "Automode on/off" with Y (= YES).

This completes the commissioning process.
3.2 PROGRAMMING
3.2 PROGRAMMING

Programming with FST 203 .................. 3-19
Function modules .......................... 3-19
Operators, command set ..................... 3-20
Operands ................................. 3-23
Summary of address space ................. 3-26
3.2 PROGRAMMING

Programming with FST 203

Two languages can be used for creating the user programs and program modules: the programming language Statement List (STL) and Ladder Diagram (LDR). You will find a description of the programming requirements for these languages and the programming techniques in the following manual:

Festo Software Tools
Statement List (STL) and Ladder Diagram (LDR) of the SF 3
FST 200 manual, Order no. 165 489

First-time users should first consult the relevant basic information manual for each programming language:

Statement list (STL) Order no. 18352
Ladder diagram (LDR) Order no. 18348

Once a program or a project has been planned and structured, work can start on the program. The following items will be required:

• Function modules
• Operators/command set
• Operands (remanent, non-remanent)

Function modules

Function modules are part of the operating system\(^\dagger\). The function modules are described in Appendix B.

\(^\dagger\) Exception: function modules 90...99 are generated by the display editor or with "Link in module" and are loaded into the control unit with "Load project".
Operators, command set

Depending on the selected program language (STL or LDR), different operators can be used for generating the program. The following section lists the permissible operators (command set).

**PLEASE NOTE**
There is a detailed description of the operators in the FST manual.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEP</td>
<td>For sequencing program: symbolic step marks are permitted</td>
</tr>
<tr>
<td>IF</td>
<td>Introduction to conditional part</td>
</tr>
<tr>
<td>THEN</td>
<td>Introduction of an executive part, provided that the IF condition is satisfied.</td>
</tr>
<tr>
<td>OTHR</td>
<td>Introduction of an alternative executive part, if the IF condition is not satisfied.</td>
</tr>
<tr>
<td>NOP</td>
<td>Blank operation, unconditional in conditional part (IF NOP), substitute symbol in executive part</td>
</tr>
<tr>
<td>CFM</td>
<td>Call function module (CFM 0 ... CFM 255)</td>
</tr>
<tr>
<td>CPM</td>
<td>Call program module (CPM 0 ... CPM 15)</td>
</tr>
<tr>
<td>JMP TO</td>
<td>Jump to a step mark, instruction follows in an executive part.</td>
</tr>
<tr>
<td>SET</td>
<td>One-bit operands are set to logical &quot;1&quot;, timer, counters or programs are started. Instruction follows in an executive part. The instruction stores itself in the memory.</td>
</tr>
<tr>
<td>RESET</td>
<td>One-bit operands are set to logical &quot;0&quot;, timers, counters or programs are stopped. Instruction follows in an executive part. The instruction stores itself in the memory.</td>
</tr>
<tr>
<td>LOAD</td>
<td>With this instruction, single-bit and multi-bit operands and constants are loaded into the accumulator.</td>
</tr>
</tbody>
</table>

*Fig. 3/11a: Operators of control block SF 3*
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SWAP</strong></td>
<td>High-order and low-order bytes are swapped in the multi-bit accumulator.</td>
</tr>
<tr>
<td><strong>SHL</strong></td>
<td>All bits in the multi-bit accumulator are shifted one position to the left. Bits pushed beyond the left edge are lost.</td>
</tr>
<tr>
<td><strong>SHR</strong></td>
<td>All bits in the multi-bit accumulator are shifted one position to the right. Bits pushed beyond the right edge are lost.</td>
</tr>
<tr>
<td><strong>ROL</strong></td>
<td>All bits in the multi-bit accumulator are rotated to the left; the last bit becomes the first bit and the second last becomes the last etc.</td>
</tr>
<tr>
<td><strong>ROR</strong></td>
<td>All bits in the multi-bit accumulator are rotated to the right; the first bit becomes the last bit and the second becomes the first etc.</td>
</tr>
<tr>
<td><strong>PSE</strong></td>
<td>Initiates a program task change, is set as program end.</td>
</tr>
<tr>
<td><strong>BID</strong></td>
<td>Converts the contents of the multi-bit accumulator from binary code to decimal code.</td>
</tr>
<tr>
<td><strong>DEB</strong></td>
<td>Converts the contents of the multi-bit accumulator from decimal code to binary code.</td>
</tr>
<tr>
<td><strong>[X], [Y]</strong></td>
<td>Index register for indexed addressing (simpler programming)</td>
</tr>
<tr>
<td><strong>(</strong></td>
<td>Left bracket, start of a summary of several instructions</td>
</tr>
<tr>
<td><strong>+</strong></td>
<td>Arithmetic instruction for Appendix, also prefix for constants</td>
</tr>
<tr>
<td><strong>-</strong></td>
<td>Arithmetic instruction for subtraction</td>
</tr>
<tr>
<td>*****</td>
<td>Arithmetic instruction for multiplication</td>
</tr>
<tr>
<td><strong>/</strong></td>
<td>Arithmetic instruction for division</td>
</tr>
<tr>
<td><strong>&lt;</strong></td>
<td>Arithmetic comparison (less than ...)</td>
</tr>
<tr>
<td>**&lt;=&quot;</td>
<td>Arithmetic comparison (less than or equal to ...)</td>
</tr>
<tr>
<td>**=&quot;</td>
<td>Arithmetic comparison (equal to ...)</td>
</tr>
<tr>
<td>**=&gt;&quot;</td>
<td>Arithmetic comparison (equal to or greater than ...)</td>
</tr>
<tr>
<td>**=&gt;&quot;</td>
<td>Arithmetic comparison (equal to or greater than ...)</td>
</tr>
</tbody>
</table>

Fig. 3/11b: Operators of control block SF 3
When using **STL** as a programming language, the full range of instructions is available.

When using **LDR**, certain instructions cannot be used, or have to be used in a different form. All arithmetical operations are performed in boxes. The precise syntax is described in detail in the FST 200 manual.

---

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Arithmetic comparison (greater than ....)</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Arithmetic comparison (not equal to ....)</td>
</tr>
<tr>
<td>)</td>
<td>Close bracket, end of a summary of several instructions</td>
</tr>
<tr>
<td>AND</td>
<td>Logical instruction for bit-by-bit AND link</td>
</tr>
<tr>
<td>OR</td>
<td>Logical instruction for bit-by-bit OR link</td>
</tr>
<tr>
<td>EXOR</td>
<td>Logical instruction for bit-by-bit exclusive OR link</td>
</tr>
<tr>
<td>TO</td>
<td>When combined with LOAD, transfers operand 1 to operand 2</td>
</tr>
<tr>
<td>SHIFT</td>
<td>Swaps the one-bit operand indicated next with the value in the one-bit accumulator</td>
</tr>
<tr>
<td>INC</td>
<td>Multi-bit operands have their value increased by 1 (incrementally)</td>
</tr>
<tr>
<td>DEC</td>
<td>The value of multi-bit operand is reduced by 1 (decrementally)</td>
</tr>
<tr>
<td>WITH</td>
<td>Parameter transfer is initiated in response to module calls (CPM .... WITH ....)</td>
</tr>
<tr>
<td>N</td>
<td>Negation: negates the operand, i.e. they are queried as logical ZERO</td>
</tr>
<tr>
<td>CPL</td>
<td>Complements multi-bit operands by applying the method of the 2nd complement</td>
</tr>
<tr>
<td>INV</td>
<td>Complements multi-bit operands by applying the method of the 1st complement</td>
</tr>
</tbody>
</table>

*Fig. 3/11c: Operators of control block SF 3*
Operands

The following table contains the operands managed in the standalone operating mode of the operating system in control block SF 3 (local I/O, diagnostic values for local diagnosis). They can be addressed using Statement List (STL) and Ladder Diagram (LDR). Appendix B contains a summary list of all operands.

RAM boot mode:
In RAM boot mode, there is no way of ensuring that data are retained (programs/operands) in the event of a mains power failure, even if this failure only lasts for a few seconds. You must assume that there will be a data loss.

EEPROM boot mode:
In EEPROM boot mode, the operands shown on the table are remanent (column "Remanent"/Yes). These operands are backed up in the event of a power failure and can be called up again once power is restored (POWER ON). All other data are lost (non-remanent).

The following illustrations provide an overview of:
- The available operands in standalone operating mode
- And the remanent operands (EEPROM boot mode).
### Available Operands in Standalone Operating Mode

<table>
<thead>
<tr>
<th>Operand</th>
<th>Number</th>
<th>Designation</th>
<th>Parameter</th>
<th>Comments</th>
<th>Remain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>128</td>
<td>I</td>
<td>{0...31},{0...7}</td>
<td>One-bit operand, allocation dependent on configuration</td>
<td>no</td>
</tr>
<tr>
<td>Input words</td>
<td>31</td>
<td>IW</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Outputs</td>
<td>128</td>
<td>O</td>
<td>{0...31},{0...7}</td>
<td>One-bit operand, allocation dependent on configuration</td>
<td>no</td>
</tr>
<tr>
<td>Output words</td>
<td>31</td>
<td>OW</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Diagnosis</td>
<td>Inputs</td>
<td>32</td>
<td>I</td>
<td>One-bit operand, for diagnosis of local I/Os and collective errors</td>
<td>no</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IW</td>
<td>{0},{0...3},{0...7}</td>
<td>Multi-bit operand for diagnosis of local I/Os and collective error</td>
<td>no</td>
</tr>
<tr>
<td>Flags</td>
<td>512</td>
<td>F</td>
<td>{0...31},{0...15}</td>
<td>One-bit operand 31 flag words, each with 16 flags</td>
<td>yes</td>
</tr>
<tr>
<td>Flag words</td>
<td>32</td>
<td>FW</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>FI</td>
<td>1</td>
<td>One-bit operand, FI only occurs with LDR (Ladder Diagram)</td>
<td>no</td>
</tr>
<tr>
<td>Index Register</td>
<td>2</td>
<td>X Y</td>
<td>none</td>
<td>Register for index addressing</td>
<td>no</td>
</tr>
<tr>
<td>Counters</td>
<td>32</td>
<td>C</td>
<td>{0...31}</td>
<td>One-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Counter preselection</td>
<td>32</td>
<td>CP</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Counter words</td>
<td>32</td>
<td>CW</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Timer</td>
<td>32</td>
<td>T/TON/TOFF</td>
<td>{0...31}</td>
<td>One-bit operand, all timers can be programmed as pulse timer T, with LDR (Ladder Diagram) also as delayed response timer units (delay switching on (T0N) or off (T0FF))</td>
<td>no</td>
</tr>
<tr>
<td>Timer preselection</td>
<td>32</td>
<td>TP</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Timer words</td>
<td>32</td>
<td>TW</td>
<td>{0...31}</td>
<td>Multi-bit operand</td>
<td>no</td>
</tr>
</tbody>
</table>

Fig. 3/12a: Available operands in standalone operating mode
**3.2 Programming**

<table>
<thead>
<tr>
<th>Operand</th>
<th>Number</th>
<th>Designation</th>
<th>Parameter</th>
<th>Comments</th>
<th>Rema-ent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interrupt-controlled counters/timers</td>
<td>4</td>
<td>see Appendix B</td>
<td></td>
<td>Freely available, interrupt-controlled counters or timers, access only via CFM 10 and CFM 11</td>
<td>no</td>
</tr>
<tr>
<td>Register</td>
<td>128</td>
<td>R</td>
<td>{0...99}</td>
<td>Multi-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Error</td>
<td>1</td>
<td>E</td>
<td>1</td>
<td>One-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Error word</td>
<td>1</td>
<td>EW</td>
<td>1</td>
<td>Multi-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Program</td>
<td>16</td>
<td>P</td>
<td>{0...15}</td>
<td>One-bit operand</td>
<td>yes</td>
</tr>
<tr>
<td>Program modules</td>
<td>16</td>
<td>CPM</td>
<td>{0...15}</td>
<td>Creation on the PC in Statement List (STL) or LDR (Ladder Diagram)</td>
<td>yes</td>
</tr>
<tr>
<td>Remanent data words</td>
<td>512</td>
<td>DW</td>
<td>see Appendix B</td>
<td>Freely available, remanent 16-bit values, access only via CFM 5 and CFM 6</td>
<td>yes</td>
</tr>
<tr>
<td>Function modules</td>
<td>CFM</td>
<td>Refer to Appendix B</td>
<td></td>
<td>Contained in EPROM operating system</td>
<td>yes</td>
</tr>
</tbody>
</table>

Fig. 3/12b: Available operands in standalone operating mode

**PLEASE NOTE**
- Not all inputs and outputs are actually available or able to be connected.
- Appendix B contains a general overview of all operands.

Note about the special operands:
- The special operands FU0...FU4 are reserved for system flags of the operating system.
- The special operands FU32...FU47 are reserved for the transfer parameters of program modules and function modules.
Summary of address space

Available address space is determined by

- The type of valve terminal (02; 03...05)
- The hardware fitted to modular designs of valve terminal (depending on whether an AS-i master or CP module is installed)
- The preset operating mode of the valve terminal (standalone / master / slave).

The following diagram shows the address space for the standalone operating mode (i.e. without AS-i master, without field bus, without CP module). In every diagram, the structure of the valve terminal is shown in schematic form and the address space is also listed in a table.

Further summaries of address space are shown in:

- Chapter 4, for master and slave operating modes (incl. field bus)
- Chapter 6, if an AS-i master is available
- Chapter 7, if a CP interface is available
- Appendix B: overall summary
Fig. 3/13: Summary of SF 3 address space in standalone operating mode
3.3 PROGRAMMING TECHNIQUES
Contents

3.3 PROGRAMMING TECHNIQUES

Structuring . . . . . . . . . . . . . . . . . . . . . . . . . . 3-31
Hierarchical depth of programs
and modules . . . . . . . . . . . . . . . . . . . . . . . . . 3-33
Program processing . . . . . . . . . . . . . . . . . . . . 3-34
Multi-tasking characteristics . . . . . . . . . . . . . . 3-36
Process image . . . . . . . . . . . . . . . . . . . . . . . . . 3-37
Cycle time . . . . . . . . . . . . . . . . . . . . . . . . . . 3-38
Measurement of program run time . . . . . . . . . . . 3-38
Program processing
  Module call in LDR/STL without STEP . . 3-39
  Programs and modules in
    Statement List with STEP . . . . . . . . . . . . 3-40
  Modules for text output . . . . . . . . . . . . . . . 3-42
  Additional remanent data words . . . . . . . . 3-43
  Indexed processing . . . . . . . . . . . . . . . . . . . . 3-43
  Application range for the
    remanent data words . . . . . . . . . . . . . . . . 3-44
  Application example - placing receptacles . 3-45
  Interrupt-controlled timers/counters . . . . . . 3-46
  Application range of
    interrupt-controlled counters . . . . . . . . . . 3-49
  Application range of
    interrupt-controlled timers . . . . . . . . . . . . 3-51
3.3 PROGRAMMING TECHNIQUES

This chapter describes some of the special features of the operating system which are helpful in solving complex programming tasks and permit optimisation of the program sequences.

Structuring

Programmable valve terminals provide the following aids for structuring a program or project:
- Programs (P0...P15)
- Program modules (CPM 0...CPM 15)
- Function modules (CFM 0...CFM 255)

Program modules and function modules can be called up by an active program.

Programs:
The programmable valve terminal can process several programs in parallel (multi-tasking). In this case, up to 16 programs can be processed simultaneously.

Program modules:
For structured programming and to save program memory in the user program, frequently required command sequences can be programmed in the program modules (sub-programs). These can be compiled with the Festo programming software in Statement List (STL) and/or Ladder Diagram (LDR) or as a text module with the text display editor.
Function modules:
A few functions of the programmable valve terminal, e.g. for diagnosis and evaluation of peripheral equipment, are already defined in the operating system (EPROM). This allows easy access (to data) in the peripheral equipment.

Fig. 3/14: Structuring of a program/project
Hierarchical depth of programs and modules

Program modules and function modules can be called up by active programs. Function modules can also be called up by program modules.

PLEASE NOTE
Function modules CFM 90...99 can only be called up directly by programs.

Fig. 3/15: Hierarchical depth of programs and modules
Program processing

Using the RUN instruction or in automode, a program (preferably P0) is started.

Recommendation:

Use program P0 as a control program for the following problems:
- Starting and stopping other programs
- General monitoring

Other programs, e.g. P1, P3, P13, can be called up by the active program P0 (active) or stopped (passive).
Fig. 3/16: Program processing with control program P0
**Multi-tasking characteristics**

The operating system of the SF 3 control block allows for processing of several programs (tasks) at the same time (multi-tasking). Altogether, 16 programs are available to the user. The task change in a Statement List (STL) program always takes place **before** the instruction step or at the same time as an instruction processor change (PW); with an LDR (Ladder Diagram) program, the task change always takes place at the end of the program or at the time of a return jump.

Example with 2 active programs:
Both active programs must be processed alternately, i.e. the program parts (STEP, Ladder Diagram sequence) of all active programs described above must be processed in quick succession.

![Diagram](image)

*Fig. 3/17: Multi-tasking characteristics with two active programs (example)*

Step 1 of program y is not processed until step 1 of program x has run once right through. Processing follows so quickly that it appears to be simultaneous.
Process image

The programmable valve terminal operates with a process image.

At the start of a processing cycle, the status of every input is read and placed in buffer storage in the inputs process image (PII).
If the outputs are modified in the course of the processing cycle, this is placed in buffer storage in the outputs process image (PIO). At the end of each processing cycle, the PIO is transferred to the outputs of the valve terminal.

Advantages:

- The process image is independent of the cycle time of the total project.
- Very fast processing cycles because the PII/PIO is updated/carried out every time there is a change of task.

Fig. 3/18: Process image (PII/PIO) for each change of task
Cycle time

The complete cycle time for the entire project is **not** calculated from the overall length of the project (all programs) but is instead calculated generally from:

- The number of active programs \( x \) (time required for instructions in the active step of the currently active program + time required for task changes)

An accurate observation can be performed using function module CFM 4.

Measurement of program run time

In addition to the measurement of cycle times for programs 0...15 (program run time), function module CFM 4 also permits the measurement of each section of the program.

Due to interrupt-related events during program processing, repeated measurement of the relevant time window (program section) can give rise to differing measurement results.

In contrast to this, for the cycle times of programs 0...15, the maximum value is calculated over the entire measuring period (refer to description of all CFMs, Appendix B).

PLEASE NOTE

*Please note that the program cycle time is increased during the run-time of a program if FST 200 is used in online mode at the same time.*
Program processing

Module call in LDR/STL without STEP

The call of a program module or function module is interpreted as a sub-program call. When the program module or function module has been fully processed, the program is continued from the interrupt point. This applies to:

- Modules in LDR (exception reverse jump)
- Module in STL without STEP

Fig. 3/19: Program processing of modules without STEP
Programs and modules in Statement List with STEP

A task change is performed before the STEP 2 instruction (refer to Fig. 3/20: A') within a program and within a called up module. The following instructions within a called up module and the program are not processed. The task is transferred to the next active program.

When the following active programs have been processed, the task is transferred back to the program (refer to Fig. 3/20: P1). If the executive part of the last sentence is processed before the instruction STEP 2, the instructions are continued at the interrupt point (refer to Fig. 3/20: B). If the executive part of the last sentence is not processed before the instruction STEP 2, the relevant STEP 1 is repeated (refer to Fig. 3/20: A).

PLEASE NOTE

- When a program with STEP instruction is first called up, a task change occurs with the first instruction STEP. The next instruction is processed at the next task transfer point.
- If the last instruction programmed into a program with STEP instructions is the PW command, the task is transferred on an endless loop to the program and the instructions of the last step are processed.
Fig. 3/20: Program processing Statement List (STL) with STEP (example)
Modules for text output

The procedure for text output via the serial interface is similar to that for programs and modules with STEP instructions. The calling program (refer to Fig.: ... P1) is interrupted until all characters are output. Take due account of this during the program design stage.

Fig. 3/21: Module processing with text output (example)
Additional remanent data words

As from hardware status 1097, the data remanent values (saved and protected against loss due to power failure) have been extended by 512 words (16-bit values).

These remanent data words (designation DW) are freely available and are supported by function modules 5 (Read remanent data) and 6 (Write remanent data).

Indexed processing

Access to the remanent data words is made with the function modules, by specifying the relevant starting address as from which the appropriate number of data words (max. 15) are to be read or written.

A detailed description of how to access function modules 5 and 6 (CFM 5, CFM 6) can be found in Appendix B.
Application range for the remanent data words

You can store all user-specific data, which has to be saved and protected against loss due to power failure, in the remanent data words. This data can be e.g. product-specific mixture ratios or also processing data such as the total number of items or other statistical values.
Application example - placing receptacles

![Diagram of mixture ratios]

**Fig. 3/23: Mixture ratios**

Data words:
- DW0 = 30 water [%] | Data record 1
- DW1 = 10 sugar [%]
- DW2 = 1 juice no. [-] | for
- DW3 = 60 juice amount [%] | product 1
- DW4 = 5 mix duration [min]
- ...
- DW10 = 15 water [%]
- DW11 = 5 sugar [%] | Data record 2
- DW12 = 2 juice no. [-] | for
- DW13 = 80 juice amount [%] | product
- DW14 = 2 mix duration [min]
- ...
- DW20 = 70 water [%]
- DW21 = 20 sugar [%] | Data record 3
- DW22 = 3 juice no. [-] | for
- DW23 = 10 juice amount [%] | product
- DW24 = 8 mix duration [min]
- ...
- DW511
Interrupt-controlled timers/counters

As from hardware status 1097, the SF 3 control block has 4 additional interrupt-controlled timers/counters.

These enable the user to register fast counting processes (operating mode counter) or to control time-dependent events (operating mode timer), irrespective of the cycle time required for processing user programs.

The user can decide how to divide the interrupt functions (max. 4) into the operating modes timer or counter. The user can also define the direction of counting (upwards/downwards).

When the event is triggered (underrun or overrun of counter/timer), an output and/or a flag can be set, reset or reactuated (toggle function) as desired.

**Fig. 3/24: Summary of interrupt-controlled timers/counters**
The counters/timers are parametrized and controlled by means of function modules 10 and 11 (CFM 10, CFM 11).

A detailed description of how to access function modules 10 and 11 (CFM 10, CFM 11) can be found in Appendix B.

The use of interrupt-controlled timers/counters has hardly any influence on the time reaction of the SF 3.

**PLEASE NOTE**

If an output is specified as the trigger destination, not only will this output be written in direct access when the trigger event occurs, but the other outputs on this module will also be written with the current values in the process image. This procedure differs from the usual method of operation of the cyclic update of the I/O peripherals after a task change.

In the Reload mode, the parametrized starting value is loaded again automatically into the current counter value or timer value when the event is triggered. The interrupt remains released until blocking is explicitly made by means of function module 11.
In normal mode (operation without reload function), the interrupt is blocked with the trigger event. The starting value is loaded into the current counter or timer value. The counter/timer can be triggered later in the user program if function module 11 is accessed.

It is not necessary to parametrize again by means of function module 10, as the last parametrized "old" values are retained.

<table>
<thead>
<tr>
<th>Parametrize interrupt-controlled counter/timer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number and mode</td>
</tr>
<tr>
<td>Trigger source and function</td>
</tr>
<tr>
<td>Trigger destination of output</td>
</tr>
<tr>
<td>Trigger destination of flag</td>
</tr>
<tr>
<td>Starting value</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Read interrupt-controlled counter/timer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number and mode</td>
</tr>
<tr>
<td>→ CFM 10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Block/release interrupt-controlled counter/timer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number and status (control bits)</td>
</tr>
<tr>
<td>→ CFM 11</td>
</tr>
</tbody>
</table>

Fig. 3/25: Parametrizing/controlling counters/timers

If only one parameter, which corresponds to the counter/timer number, is transferred to function module 10, the return parameter will contain the current counter/timer value. This enables e.g. the impulse duration of an input signal to be measured.
PLEASE NOTE
The values last parametrized with CFM 10 are retained. The parametrization of the counters/timers needs therefore to be carried out only once, if it does not change. This applies especially to counters/timers in normal mode, i.e. they can be reactivated by means of CFM 11.

Application range of interrupt-controlled counters

The following applications are possible with interrupt-controlled counters:
- fast registering of parts (irrespective of cycle time)
- registering of speeds (irrespective of cycle time)
- if you use the "fast" input module (input signal delay of 1ms), you can implement counter inputs up to 300 Hz by triggering on the positive and/or negative edge.
Fig. 3/26: Application example - interrupt-controlled counter

Program example - interrupt-controlled counter:

<table>
<thead>
<tr>
<th>STEP</th>
<th>(1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF</td>
<td>NOP</td>
</tr>
<tr>
<td>THEN</td>
<td>CFM 10</td>
</tr>
<tr>
<td>WITH</td>
<td>KS2001</td>
</tr>
<tr>
<td></td>
<td>&quot;Nominal Mode backwards counter no. 1&quot;</td>
</tr>
<tr>
<td>WITH</td>
<td>K$4104</td>
</tr>
<tr>
<td></td>
<td>&quot;Input I4.1 at negative edge&quot;</td>
</tr>
<tr>
<td>WITH</td>
<td>K$4203</td>
</tr>
<tr>
<td></td>
<td>&quot;Output O3.2 set at triggering&quot;</td>
</tr>
<tr>
<td>WITH</td>
<td>K$0000</td>
</tr>
<tr>
<td></td>
<td>&quot;Flag is not affected&quot;</td>
</tr>
<tr>
<td>WITH</td>
<td>K$000A</td>
</tr>
<tr>
<td></td>
<td>&quot;Nominal value 10&quot;</td>
</tr>
<tr>
<td>CFM 11 WITH</td>
<td>K3</td>
</tr>
<tr>
<td></td>
<td>&quot;Enable counter no. 1&quot;</td>
</tr>
</tbody>
</table>

Fig. 3/27: Blocking/releasing the interrupt
Application range of interrupt-controlled timers

- In the timer operating mode, time-dependent events with a resolution of ±1 ms can be controlled (fast impulse output, time value 0...65535 ms).
- A gate function is possible by means of direct AND-linking with an input or flag. The gate function is determined as an input or flag by means of the trigger source ascertained in transfer parameter P2.
- Automatic reloading of the starting value is possible (reload mode).

Program example - interrupt-controlled timer:

```
STEP (1) IF NOP THEN CFM 10
        WITH RS4003 "Reload Mode Timer down no. 3
        WITH KS4003 "Input I3.0 Gate low level
        WITH KS6201 "Output O1.2 toggle at triggering
        WITH KS0000 "Flag is not affected
        WITH KS0014 "Nominal value 20
        CFM 11 WITH KS0300 "Enable timer no. 3
```

Fig. 3/28: Application example - interrupt-controlled timer
3.4 DIAGNOSIS AND ERROR HANDLING
3.4 Diagnosis and error handling

Contents

3.4 DIAGNOSIS AND ERROR HANDLING

Summary of diagnosis options ........... 3-55
Diagnosis on site ......................... 3-56
Diagnosis by program .................... 3-60
Function modules ......................... 3-60
Diagnosis bytes .......................... 3-60
Error handling ............................ 3-62
3.4 DIAGNOSIS AND ERROR HANDLING

Summary of diagnosis options

The programmable valve terminals offer comprehensive and convenient options for diagnosis and error handling. Irrespective of the equipment on the terminal, the following options are available:

<table>
<thead>
<tr>
<th>Programmable valve terminals</th>
<th>Diagnosis possibilities</th>
<th>Diagnosis bytes (IW0.0-0.3) Function modules CFM 1 CFM 2</th>
<th>Error word EW (multi-bit operand)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEDs</td>
<td>The LEDs indicate a direct configuration error, hardware error etc.</td>
<td>The status byte and the function modules must be read out, controlled by the user program and evaluated.</td>
<td>The error word must be read out, controlled by the program and evaluated.</td>
</tr>
<tr>
<td>Brief description</td>
<td>Fast &quot;on site&quot; error recognition</td>
<td>Detailed error detection by program for faults in electrical system</td>
<td>Detailed error recognition by program for faults in program or operating system.</td>
</tr>
<tr>
<td>Advantages</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailed description</td>
<td>3.4.1</td>
<td>3.4.2</td>
<td>3.4.3</td>
</tr>
</tbody>
</table>

Fig. 3/29: Options for diagnosis and error handling
Diagnosis on site

LED displays

Control block SF 3

The light emitting diodes (LED) on the cover of the field bus node provide information about the operating condition of the valve terminal:

![Diagram of LED displays on control block SF 3]

*Fig. 3/30: LEDs on control block SF 3*
This diagram illustrates the possible LED displays which show the operating condition of the programmable valve terminal:

<table>
<thead>
<tr>
<th>LED</th>
<th>Status</th>
<th>Operating condition</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>POWER (green)</td>
<td>Lit</td>
<td>Operating voltage for electronics normal (pin 1). Exception: Despite a switched output (yellow LED on valve lit), a valve fails to switch. Possible cause: • Operating voltage on outputs not in permitted tolerance or not applied (pin 2). • Compressed air supply not O.K. • Pilot exhaust blocked.</td>
<td>None Check the... • Operating voltage of outputs (pin 2). Tolerance range 21.6...26.4 V. • Compressed air supply (work air and where applicable, auxiliary pilot air) • Pilot exhaust ducts.</td>
</tr>
<tr>
<td>Dark (not lit)</td>
<td>None</td>
<td>Operating voltage of electronics not applied.</td>
<td>Check operating voltage connection for electronics (pin 1).</td>
</tr>
<tr>
<td>RUN (green)</td>
<td>Lit</td>
<td>RUN mode active. A program is running.</td>
<td>None</td>
</tr>
<tr>
<td>Dark (not lit)</td>
<td>None</td>
<td>No program started or Program started stopped due to error</td>
<td>None or start program (adjust Automode) or Read error from error word (refer to error message)</td>
</tr>
<tr>
<td>ERROR (red)</td>
<td>Lit</td>
<td>Error in control unit or program</td>
<td>Read error from error word (refer to error message)</td>
</tr>
<tr>
<td>Dark (not lit)</td>
<td>None</td>
<td>Control/program without error</td>
<td>None</td>
</tr>
</tbody>
</table>

Fig. 3/31: LED display - operating condition
Valves

There is a yellow LED on every valve solenoid coil. These LEDs indicate the operating status of the valve solenoid coil. A type 03 valve terminal is used here to illustrate the LEDs: they have the same meaning on other valve terminals as well.

<table>
<thead>
<tr>
<th>LED</th>
<th>Switching position valve solenoid coil</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yellow (not lit)</td>
<td>Normal position</td>
<td>Logical 0 (signal not applied)</td>
</tr>
<tr>
<td>Yellow lit</td>
<td>• Switching position or Normal position</td>
<td>Logical 1 (signal applied)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Logical 1 but:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Operating voltage of outputs is below the permitted tolerance range (DC 21.6 V...26.4 V) or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Compressed air supply not O.K. or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Pilot exhaust blocked or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Servicing required</td>
</tr>
</tbody>
</table>

Fig. 3/32: Switching modes of valve solenoid coils
Input/output modules

Next to the relevant connections on the input/output modules, there are one or two LED (status displays) in the following colours:

- Green (status display for digital inputs).
- Yellow (status display for digital outputs).
- Red (error display for digital outputs).

The yellow and green LEDs display the current signal at the relevant input or output. The red LEDs for the outputs indicate short circuits/overloads on the relevant output. A type 03 valve terminal is used here to illustrate the LEDs: they have the same meaning on other valve terminals as well.

<table>
<thead>
<tr>
<th>LED</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yellow (not lit) or Green (not lit)</td>
<td>Logical 0 (signal not applied)</td>
</tr>
<tr>
<td>Yellow lit or Green lit</td>
<td>Logical 1 (signal applied)</td>
</tr>
</tbody>
</table>

Fig. 3/33: LED displays for input/output modules
### Diagnosis by program

Diagnosis by program is performed by function modules or by diagnosis bytes using local I/O diagnosis.

#### Function modules

Function modules CFM 1 and CFM 2 are particularly suitable for use in the event of a short circuit.

Recommendation:
Proceed as follows in the program:
1. Short circuit (s.c.) detected in diagnosis byte (collective error I0.0.2)
2. Localise short circuit (s.c.) with CFM 1 or with IW0.2 (bit number) and IW0.3 (word number)
3. Reset relevant output and, if applicable, set again with CFM 2. The function modules are listed and described in Appendix B.

### Diagnosis bytes

The local diagnosis information for a valve terminal is grouped into diagnosis bytes and represented as input words 0.0 to 0.3 [IW0.0 - IW0.3]. With these diagnosis bytes, the following local error modes are identified and communicated back to the control block:

<table>
<thead>
<tr>
<th>Diagnosis information</th>
<th>Meaning</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{\text{valves (Vval)}}$</td>
<td>Monitors the tolerance of operating voltage of the valves and electrical outputs.</td>
<td>Operating voltage on pin 2 of the operating voltage connection &lt; 21.6 V</td>
</tr>
<tr>
<td>$V_{\text{outputs (Vout)}}$</td>
<td>Monitors the operating voltage of the valves and electrical outputs (no power connected).</td>
<td>Operating voltage on pin 2 of the operating voltage connection &lt; 10 V</td>
</tr>
<tr>
<td>$V_{\text{inputs (Vinp)}}$</td>
<td>Monitors the supply voltage to inputs/sensors.</td>
<td>Internal fuse triggered.</td>
</tr>
<tr>
<td>sc/o</td>
<td>Monitors the electric outputs of output modules.</td>
<td>Short circuit (s.c.) or overload</td>
</tr>
</tbody>
</table>

*Fig. 3/34: Local error modes of the valve terminal*
### 3.4 Diagnosis and error handling

Structure and significance of the local diagnosis bytes:

#### IW0.0

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>None</td>
<td>$V_{\text{out}}$</td>
<td>$V_{\text{val}}$</td>
<td>$V_{\text{inp}}$</td>
<td>Terminal 02: sc/o O0.0</td>
<td>None</td>
<td>None in stand-alone operating mode</td>
<td>None in stand-alone operating mode</td>
</tr>
</tbody>
</table>

- Terminal 02: sc/o O0.0
- Terminal 03: Collective signal sc/o refer to IW0.2 and IW0.3 or CFM 1

<table>
<thead>
<tr>
<th>Signal status</th>
<th>0 or 1</th>
</tr>
</thead>
</table>

**Meaning**
- Signal status 0: no error
- Signal status 1: error

#### IW0.1

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>None</td>
<td>Collective signal, individual error CP component(s) - refer to CFM 25 -</td>
<td>Collective signal failure of CP component(s) - refer to CFM 25 -</td>
<td>Booting of CP system</td>
</tr>
</tbody>
</table>

#### IW0.2

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>Short circuit (s.c.) local electrical outputs bit number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### IW0.3

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>Short circuit (s.c.) local electrical outputs byte number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 3/35: Structure of diagnosis bytes (input words IW0.0 to IW0.3)*
Error handling

Error messages in the operating system

The red error display (ERROR) lights up whenever the error status is not equal to 0. The error numbers are entered in the error word.

The complete list of all error messages in the operating system is to be found in Appendix B.
Programmable valve terminal
with
control block SF 3

Chapter 4: System description
field bus module
4.1 COMMISSIONING .............................. 4-3
   Requirements .............................. 4-3
   System structure with field bus module .... 4-3
   Address allocation for valve terminal in
   operating modes master and slave........ 4-4
   Basic rule 4
   (extension to basic rules in chapter 2) .... 4-4
   Online operation with field bus .......... 4-8
   Setting master or slave operating mode.... 4-9
   Switching outputs on and off ............. 4-10
   Commissioning steps in
   master operating mode .................... 4-11
   Configuring the field bus with FST 200 . 4-12
   Planning the field bus configuration ..... 4-13
   Commissioning steps in
   slave operating mode .................... 4-15

4.2 PROGRAMMING ............................... 4-19
   Field bus operands ....................... 4-19
   Field bus function modules 40 to 49 .... 4-20
   Representation of address
   space allocation ........................ 4-21

4.3 PROGRAMMING TECHNIQUES ............... 4-25
   Communication between SF 3
   as master and slave ..................... 4-25
   Cyclic communication .................... 4-27
   Acyclic communication ................... 4-32
   Starting up after Power on .............. 4-34
   Reaction to transmission errors .......... 4-36
   Reaction to station errors .............. 4-38
4.4 DIAGNOSIS AND ERROR HANDLING . . . 4-43

  Summary of diagnosis options . . . . . . . . . . 4-43
  Diagnosis by program . . . . . . . . . . . . . 4-44
  Diagnosis in master operating mode . . . . . 4-44
  Function module for field bus diagnosis . . 4-48
  Diagnosis in slave operating mode . . . . . 4-50
  Error words . . . . . . . . . . . . . . . . . . 4-50
4.1 COMMISSIONING
4.1 COMMISSIONING

Requirements .................................................. 4-3
System structure with field bus module ............. 4-3
Address allocation for valve terminal in
operating modes master and slave ............... 4-4
Basic rule 4
(extension to basic rules in Chapter 2) ...... 4-4
Online operation with field bus ................. 4-8
Setting master or slave operating mode .... 4-9
Switching outputs on and off ....................... 4-10
Commissioning steps in master
operating mode ............................................. 4-11
Configuring the field bus with FST 200 .... 4-12
Planning the field bus configuration .......... 4-13
Commissioning steps in slave
operating mode ............................................. 4-15
4.1 COMMISSIONING

Requirements

In addition to the commissioning steps named in chapter 3 for standalone operating mode the following steps are required for field bus systems:

• Compiling an allocation listing for masters and all active slaves
• Configuring the master

System structure with field bus module

With built-in field bus module, the programmable terminal can be used as either a master or slave on the Festo field bus. In addition to the standalone operating mode, there are two other possible uses for your automation tasks:

• Valve terminal in master operating mode:
  In this autonomous standalone solution, passive slaves are added to the field bus (e.g. valve terminal, I/O extensions). The master controls its local I/Os and centrally controls all field bus I/Os.

• Valve terminal in slave operating mode:
  Active slaves are added to a master on a field bus (e.g. programmable valve terminal with field bus connection SF 3). The master only controls its local I/Os and exchanges diagnosis information via the field bus. Every active station controls its local I/Os itself (independently).

A mixed operating of active and passive slaves is possible.
**Address allocation for valve terminal in operating modes master and slave**

The SF 3 programmable valve terminal can activate up to 128 local inputs and 128 local outputs. For addressing, the rules listed in chapter 2 and 3 always have to be applied. Supplementary instructions for addressing the field bus are summarized in the subsequent basic rule 4.

Other summaries can be found in:
- Chapter 6, if an AS-i master is available in addition to the field bus.
- Chapter 7, if a CP interface is available in addition to the field bus.
- Appendix B: Complete summary.

**Basic rule 4**
*(extension to basic rules in Chapter 2)*

1. **Local I/Os:**
   A total of 128 inputs and 128 outputs can be addressed on the terminal. For the address allocation of these local I/Os, basic rules 1 to 3 apply in both master and slave operating modes.

2. **Field bus I/Os:**
   The address allocation of the inputs and outputs on the field bus depends on the kind of slaves connected. A max. total of 1048 I/Os can be addressed on the field bus (distributed on up to 31 slaves, each with a max. 128 inputs and 128 outputs). Depending on the field bus address of the slaves, appropriate numerals are placed before the FST-identifiers. Example:

   ![Field bus address diagram]

   - Bit 2
   - Byte 4
   - Field bus address
Field bus inputs which are not available are always "logical 0".

**PLEASE NOTE**

Note the addressing differences between byte-oriented and word-oriented slaves.

Example:
- Byte-oriented slave with address 1:
  I1.0.0...I1.15.7
  O1.0.0...O1.15.7
- Word-oriented slave with address 1:
  I1.0.0...I1.7.15
  O1.0.0...O1.7.15

Note:
Passive slaves are addressed by the master (direct addressing). If the equipment fitted onto a passive slave changes, the addressing in the master must be adapted to reflect this change. Active slaves have a built-in PLC and are addressed and controlled independently by the programs in this PLC. The master cannot gain direct access to the local I/Os on these slaves. Only indirect addressing/communication is possible.

Advantage:
If the equipment fitted onto an active slave changes, only the addressing in the slave needs to be adapted; the other addressing in the field bus (master) is not affected.

The I/O address range with the FB address 0 can be used as follows:
- Master: local diagnosis and cyclic field bus diagnosis (refer to Chapter 4.4 "Diagnosis and error handling").
- Slave: local diagnosis and cyclic communication to master (0...12 bytes, refer to Chapter 4.3 "Programming techniques").

The following diagrams illustrate typical allocations of I/Os between masters and slaves as well as their addressing facilities with FST.
Example:
Direct addressing of passive slaves (byte-oriented and word-oriented) and indirect addressing (cyclic communication) of active slaves (SF 3).

**Visf 3 4.1 Commissioning**

**Fig. 4/1: Example - Direct and indirect addressing**
Fig. 4/2: Example – Address passive slaves (byte-oriented)
Online operation with field bus

In online operation, the current status of inputs and outputs (I/Os) can be displayed. Simple tests involving the setting of outputs are possible. These functions can be used to the following extent:

- On SF 3 as master for local I/Os and field bus I/Os.
- On SF 3 as slave for local I/Os.

In addition, using function key F5, the system configuration of the control unit (field bus settings, auto mode settings) can be carried out.

Fig. 4/3: Online-operation - System configuration field bus master/slave
Setting master or slave operating mode

Master:
For the master operating mode, the system mode is set using the function key F1, selecting "Field Bus Master". You can also adjust:
• The field bus baud rate with F2 (31.25/62.5/187.5/375 kBaud)
• The bus terminating resistor ON/OFF with F3 *)
• Auto mode ON/OFF with F7

*) Note:
If the terminal is at the start or end of a field bus line, a terminating resistor is required. This is already built in to the control block on the SF 3 and it is activated by pressing F3 (switched on/off).

Slave:
For slave operating mode, the system mode is set to "Field Bus Slave" with function key F1. In addition to baud rate, bus terminal and auto mode, the following items can be set:
• The field bus address with F4
• The input bytes for cyclic data transmission with F5 (0...12; default: 2 bytes IW0.4 and 0.5)
• The output bytes for cyclic data transmission with F6 (0...12; default: 2 bytes OW0.4 and 0.5).

WARNING
Before you adjust the operating mode "Auto mode" in master or in an active slave:
Ensure that the automatic program will not trigger accidental or dangerous procedures starts in either master/slave or in the passive field bus station.
Switching outputs on and off

WARNING
When the system is switched on, the outputs respond immediately to your screen input!
Ensure that there is no risk to people or machinery, when switching the outputs on or off.

With this function, simple and direct tests are possible before and during the commissioning process. The operating modes for master and slave are identical.

With these tests, proceed as follows:
• With key F1, select the mask for bit-by-bit switching outputs on and off in the "Online operation" menu.
• With the TAB key, select bit-by-bit input.
• With mouse or cursor keys, select the appropriate output.
• Use keys F1 to F3 to change the relevant output/operand (optional: set, reset or reverse/"toggle").
• Only on master: Press key F7 to call up inputs/outputs of field bus on screen as well. Thereafter, these values can also be changed with keys F1 to F3.
For subsequent commissioning (configuration, creating the allocation lists), different procedures are required for master and slave operation. Refer to the following sections for this:

- Commissioning steps in master operating mode
- Commissioning steps in slave operating mode

**Commissioning steps in master operating mode**

Before a valve terminal SF 3 can be operated as a master, the following requirements must be met:

- Operating mode set to master.
- Field bus station connected and field bus address set.
- Field bus baud rate of all stations set to same value.
- Terminal resistors installed or switched through at start and end of field bus.
Configuring the field bus with FST 200

During configuration, a list of all connected field bus stations is produced. This establishes a logical relationship between the field bus stations (slaves) and the master.

In the configuration list, the following items are recorded:

- Field bus address of all stations.
- Type of participant involved (inputs, outputs or mixed).
- Size of all stations (requisite number of bytes for addressing - only queried for types on which the number of inputs/outputs can vary).

Due to this list, a comparison can be made between set point and actual configuration to eliminate connection errors. The field bus configuration is described in detail in the manual for the FST 200, Chapter 8.

Fig. 4/4: Field bus configuration with FST 200
Planning the field bus configuration

An SF 3 programmable valve terminal acting as a master can manage up to 31 stations, each with 128 inputs and 128 outputs on the field bus, but no more than a maximum of 1048 I/Os. If this I/O limit is exceeded during configuration, this will give rise to an error message. For this reason, calculate before the configuration whether the number of required I/Os can be addressed (refer to the table on the next page).
### Field bus station (type) (Fig.) | Number of valve places | Number of assigned field bus I/Os
--- | --- | ---
#### Passive slaves
- **Valve terminal type 02**
  - 4...8
  - 10...14
  - 16 (2 bytes)
- **Valve sensor terminal type 02**
  - 4...6
  - 8...14
  - 32 (4 bytes)
- **Valve terminal type 03...05**
  - 0...26
  - El. outputs 0...48
  - El. inputs 0...60
  - max. 64 I (8 bytes)
  - + 64 O (8 bytes)
- **Field bus 202 (I/O)**
  - with 1 extension
  - with 2 extensions
  - with 3 extensions
  - 32 (4 bytes)
  - 64 (8 bytes)
  - 96 (12 bytes)
  - 128 (16 bytes)

#### Active slaves
- **Programmable valve terminal SF 202 type 02...05 (slave)**
  - Assigned I/Os independent of valve positions
  - 32 (4 bytes, IW4, 5 OW4, 5)
- **Programmable valve terminal SF 3 type 03...05 (slave)**
  - 0...12 bytes (I)
  - 0...12 bytes (O)
  - Input/output bytes can be configured independently of one another (IW/OIW0.4...0.15)
  - (default value: 4 bytes, IW/OIW 0.4 + 0.5)

*Fig. 4/5: Number of assigned I/Os on field bus*
Commissioning steps in slave operating mode

Before an SF 3 valve terminal can be used as a slave in the Festo field bus, the following requirements must be satisfied:

- Operating mode set to slave.
- Field bus address set to 1...31.
- Number of cyclic I/Os has been defined (IW/OW 0.4...0.15).
- Where applicable, terminating resistor activated/switched through.

If you use the SF 3 valve terminal as a slave, the following steps must be performed during commissioning on every SF 3 acting as a slave:

- Compile an allocation listing.
- Compile the programs, program the EEPROM.
- Set the boot mode and auto mode.

These steps are described in detail in the manual of FST 200.
4.2 PROGRAMMING
Contents

4.2 PROGRAMMING

Field bus operands . . . . . . . . . . . . . . . . . . . 4-19
Field bus function modules 40 to 49, . . . . . . 4-20
Representation of address space allocation . . . . . . . . . 4-21
4.2 PROGRAMMING

Field bus operands

To supplement the table in chapter 3, the following section contains details of field bus operands (field bus I/Os, special operands, function modules). They can be addressed using statement list (STL) and ladder diagram (LDR).

<table>
<thead>
<tr>
<th>Significance</th>
<th>Designation</th>
<th>Parameters</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field bus inputs</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- byte-oriented</td>
<td>I</td>
<td>{1...31}.{0...15}.{0...7}</td>
<td>One-bit operand</td>
</tr>
<tr>
<td>- word-oriented</td>
<td>IW</td>
<td>{1...31}.{0...15}</td>
<td>Multi-bit operand</td>
</tr>
<tr>
<td>Input words</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- byte-oriented</td>
<td>I</td>
<td>{1...31}.{0...15}</td>
<td>One-bit operand</td>
</tr>
<tr>
<td>- word-oriented</td>
<td>IW</td>
<td>{0}.{0...15}</td>
<td>Multi-bit operand</td>
</tr>
<tr>
<td><strong>Diagnosis and/or cyclic communication (byte-oriented)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Inputs</td>
<td>I</td>
<td>{0}.{0.15}</td>
<td>(For further information refer to Chapter 4.4)</td>
</tr>
<tr>
<td>- Input words</td>
<td>IW</td>
<td>{0}.{0...15}</td>
<td></td>
</tr>
<tr>
<td><strong>Field bus outputs</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- byte-oriented</td>
<td>O</td>
<td>{1...31}.{0...15}.{0...7}</td>
<td>One-bit operand</td>
</tr>
<tr>
<td>- word-oriented</td>
<td>OW</td>
<td>{1...31}.{0...15}</td>
<td>Multi-bit operand</td>
</tr>
<tr>
<td>Output words</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- byte-oriented</td>
<td>O</td>
<td>{1...31}.{0...15}</td>
<td>One-bit operand</td>
</tr>
<tr>
<td>- word-oriented</td>
<td>OW</td>
<td>{0}.{0...15}</td>
<td>Multi-bit operand</td>
</tr>
<tr>
<td><strong>Diagnosis and/or cyclic communication (byte-oriented)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Outputs</td>
<td>O</td>
<td>{0}.{0.15}</td>
<td>(For further information refer to Chapter 4.4)</td>
</tr>
<tr>
<td>- Output words</td>
<td>OW</td>
<td>{0}.{0...15}</td>
<td></td>
</tr>
<tr>
<td><strong>Function modules</strong></td>
<td>CFM</td>
<td>{40...44, 47...49}</td>
<td>Contained in the EEPROM operating system, refer to Appendix B.</td>
</tr>
<tr>
<td><strong>Special operands</strong></td>
<td>FU</td>
<td>{0, 1, 2}</td>
<td>Multi-bit operands for field bus information, refer to Appendix B.</td>
</tr>
</tbody>
</table>

*Fig. 4/7: Additional operands for programmable valve terminal with field bus module*
Field bus function modules 40 to 49

The SF 3 has a few function modules which simplify communication with the connected field bus stations. Activation of these function modules is performed using the user program.

PLEASE NOTE
The syntax of function modules 41 and 42 differs between master and slave.

The function modules are listed and described in Appendix B.
Representation of address space allocation

The following representation illustrates the address space for the master and slave operating modes. This illustrates the structure of the valve terminal in schematic form and lists the address space in a table.

Notes on the operands (I/Os):
- Not all local I/Os and field bus I/Os are physically available.
- Note the addressing differences in the byte-oriented and word-oriented addressing formats (dependent on field bus station).

**Fig. 4/8a: Address space in master operating mode (example for byte-oriented addressing)**
SF 3 in slave operating mode

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>Slave operating mode</th>
</tr>
</thead>
</table>
| IW0..15 OW0...15 | Local Inputs  
                             Local Outputs  
                             - Valves  
                             - Electrical Outputs |
| IW0.0...0.3       | Diagnosis  
                             local I/Os |
| IW0.4...0.15 OW0.4...0.15 | Cyclic data |

Fig. 4/8b: Address space in slave operating mode
4.3 PROGRAMMING TECHNIQUES
Contents

4.3 PROGRAMMING TECHNIQUES

Communication between SF 3
as master and slave .................. 4-25
Cyclic communication ................ 4-27
Acyclic communication ............... 4-32
Starting up after Power on .......... 4-34
Reaction to transmission errors ..... 4-36
Reaction to station errors .......... 4-38
4.3 PROGRAMMING TECHNIQUES

Communication between SF 3 as master and slave

Data transmission on the field bus is divided into cyclic and acyclic communication.

With these two communication modes, bit and word information is transferred, although communication for passive slaves is restricted to the processing of I/Os and the reading of diagnosis bytes. "Real" communication is only possible with active slaves and the following details should be considered for individual communication routes:

Cyclic communication:
Regular processing of inputs/outputs on a field bus system. In this process, approx. every 4 ms, the operating system automatically:
- transmits the status of the output,
- reads the actual status of the inputs.

In this context, the process is referred to as a "process image" (PIO, PII). Cyclic communication makes it possible to have bit-by-bit communication between master and slaves. Within a transmission cycle, a distinction must be made between the following processes:
- Master:
  In the master, the local I/Os (physical I/Os) and all field bus I/Os (logical I/Os) are processed.
- Slave:
  - In passive slaves, the field bus signals are received by the master and are displayed directly to the local (physical) I/Os.
  - In active slaves the field bus signals are received from the master and displayed on the logical I/Os. Thereafter, further processing can be performed by the internal PLC of the slave.
Acyclic communication:
Acyclic communication must be called up by program. With it, data are transferred in word length (word information). For this, the programmable valve terminal has field bus function modules CFM 40...49 at its disposal. For true communication between master and active slaves, CFM 41 and 42 are best suited. With them, the contents of timers, counters and registers can be transferred via the field bus. Transmission takes place only at the time when a function module is called up.

PLEASE NOTE
Acyclic communication is inserted in the field bus listing between the cyclic communication. Extensive or frequent use of acyclic communication delays the processing time required for cyclic communication.

The next section describes both forms of communication in greater detail.
Cyclic communication

The transmission of cyclic information differs between master and slaves as follows:

SF 3 as master:

The I/O address space of the master is subdivided into local I/Os and field bus I/Os. This means that a memory area is assigned to every field bus station in the master station in which the information from all inputs and outputs is stored in a cyclical manner. This includes:

- Logical inputs and outputs for active slaves (SF 3 acting as slaves, depending on the configuration of the slave).
- Physical inputs and outputs of all passive slaves.

The addressing of a field bus input or output takes place in the master on a byte-oriented or word-oriented manner in accordance with the following diagram:

<Field bus address>.<Byte no.>.<Bit no.>

Example: I1.0.0 or O1.0.7
**VISF 3**

4.3 Programming techniques

---

**Fig. 4/9: Master station – Cyclic communication (example with byte-by-byte addressing)**

- **Read/write**
  - cyclic approx. all 4 ms
- **Local I/Os**
- **Physical outputs**
  - (local valves and output modules)
  - O0.0...O0.7
  - O15.0...O15.7
  - O1.0.0...O1.0.7
  - O1.15.0...O1.15.7
  - O2.0.0...O2.0.7
  - O2.15.0...O2.15.7
  - O31.0.0...O31.0.7
  - O31.15.0...O31.15.7
- **Field bus I/Os**
  - Outputs, field bus slave 1
  - Outputs, field bus slave 2
  - Outputs, field bus slave 31
- **Physical inputs**
  - (local input modules)
  - I0.0...I0.7
  - I15.0...I15.7
  - I1.0.0...I1.0.7
  - I1.15.0...I1.15.7
  - I2.0.0...I2.0.7
  - I2.15.0...I2.15.7
  - I31.0.0...I31.0.7
  - I31.15.0...I31.15.7
- **Outputs, field bus slave 1**
- **Outputs, field bus slave 2**
- **Outputs, field bus slave 31**
- **inputs, field bus slave 1**
- **inputs, field bus slave 2**
- **inputs, field bus slave 31**

---

Excerpt from data memory/address field

---

**Fig. 4/9: Master station – Cyclic communication (example with byte-by-byte addressing)**

---

4-28 9809c
SF 3 as slave

In every SF 3 programmable valve terminal acting as a slave (active slave), in addition to the physical inputs and outputs, logical (field bus) inputs and outputs are also available (byte IW/OW0.4...0.15). The logical I/Os are transferred via the field bus and can then be used for cyclic communication between master and slave. Communication of this kind can be advantageous for operations such as:

- Starting/stopping programs in the slave.
- Transferring status and error messages to master.
- Transferring complete messages to master.
- Establishing an additional handshake operation to secure acyclic communication.

Every slave station can use its internal PLC to read logical inputs and to set the outputs. These logical field bus I/Os IW/OW0.4 to 0.15 can be configured freely (IW and OW independently of one another) from 0 to 12 bytes. The default values are 2 bytes for each (IW/OW0.4 and 0.5).
The contents of these bytes are transferred to the master every 4 ms.

Cyclic communication between master and active slaves is illustrated in the following figure, with 5 bytes configured for each slave.

Fig. 4/10: Slave station – Cyclic communication (byte-by-byte addressing)
Fig. 4/11: Example of cyclic communication between master/active slaves SF 3
Acyclic communication

In every slave station, a memory area (parameter field) is provided in which 16-bit information can be filed. The master stations as well as their related slaves have access authorisation to these parameter fields. This parameter field can be used for exchanging data, storage, counters or registers. The parameter field covers a storage area of 256 words, each with 16 bits. The transmission of memory contents is performed by function modules 41 and 42. These function modules are called up by program and are processed as acyclic communication.
Fig. 4/12: Principle representation of "acyclic communication" between master/active slaves
Starting up after Power on

When the SF 3 programmable valve terminal is used as master, it must be switched on at the same time as, or after, the other field bus stations. This is the only way to record the correct actual configuration and store it in memory. The start-up procedure is represented in the adjacent flow chart.

After recording the actual configuration, a comparison is made with the set point configuration and the result is stored in special operand FU0. Here is a key to the entries in FU0:

- FU0 = 0 Error occurred
- FU0 = 2 Actual list = set point list
- FU0 = 3 Actual list < > set point list
- FU0 = 4 There is only one actual list in existence

**PLEASE NOTE**

- The set point configuration must be set up with the field bus configurator of FST 203 before commissioning.
- Evaluation of the set point/actual comparison (FU0) must take place in the program. This is the only way to initiate appropriate error messages or other measures.
- If a field bus station is not recorded when the actual configuration is created, this can only be checked by the program by comparing it with the set point list.

**Recommendation:**
For this reason, always operate with a set point list.
Switch on all field bus stations
Switch on SF 3
SF 3 internal interval 2 s
Record ACTUAL configuration
ACTUAL config. = SET POINT config?
yes
FU0 = 2
Start field bus
Start Program processing evaluation FU0
no
FU0 = 0 (3,4)

Re-entry point for "reconfiguration" by CFM 48

Fig. 4/13: Start-up procedure after Power on
Reaction to transmission errors

Errors can occur during the transmission of data via the field bus. Typical causes are:

- Field bus station has not reacted twice in succession.
- Field bus cable interrupted (defective or disconnected).
- No voltage supply to field bus station.

If transmission errors occur, the collective error bit I0.0.0 is set to 1 (collective error: field bus failed).

The transmission error or failed station can then be localised more accurately using:

- IW0.4...0.7, the bit no. in this byte indicates the failed station (refer to Chapter 4.4, "Diagnosis byte IW0.4...IW0.7").
- Special operand FU1
  For field bus stations 1 to 16, the failed field bus station is displayed (refer to Chapter 4.4, "Diagnosis with special operands FU1 and FU2").
The response of programmable valve terminal SF 3 in master mode to errors in transmission can be adjusted by means of function module 47 (CFM 47). The following settings are possible:

- **“Hard” response?**
  - Yes (default)
    - SF 3 stops
    - SF 3: error 4
    - Error must be rectified
    - All field bus outputs and local outputs are switched off!
  - No
    - Programs continue to run
    - “Input” I0.0.0 = log. 1
    - Faulty FB-station bit-encoded in IW0.4...IW0.7.
    - Error must be handled by program

Settings via CFM 47

**Fig. 4/14: Response to errors in transmission**
Reaction to station errors

Errors can occur in field bus stations during operation. Typical station errors are:

- Operating voltage of valves < 21.6 volts
- Operating voltage of valves < 10 volts
- Supply of sensors < 10 volts (internal fuse tripped)
- Short circuit (s.c.) on an electrical output (output module for valve terminal type 03/05 or additional output on valve sensor terminal type 02)

If an error occurs in a field bus station, the collective error bit I0.0.1 is set to 1 (collective error: station diagnosis).

The faulty station can be localised with:

- IW0.8...0.11, the bit no. in this bytes indicates the faulty station (refer to Chapter 4.4 "Diagnosis byte IW0.8...IW0.11").
- Special operand FU2
  The faulty field bus station is displayed for field bus stations 1 to 16 (refer to Chapter 4.4, "Diagnosis with special operands FU1 and FU2").
- Function module 44 (CFM 44)
  If function module 44 is accessed without a transfer parameter, the address of the first failed field bus station, as well as that of the first faulty field bus station will be transferred to the return parameters.
Errors in the field bus station can be responded to as follows:

- Bit I0.0.1 set?
  - No: Not a field bus station error
  - Yes: Evaluation of IW0.8...0.11
      - Faulty field bus station already bit-encoded in IW0.8...0.11.
      - Error must be handled by program

*Fig. 4/15: Response to station error*
Diagnosis information on field bus stations

If a faulty field bus station is localised, function module 44 (CFM 44) can be used to read the corresponding diagnosis byte. This diagnosis byte contains detailed information about the status of the relevant station.

Diagnosis information for the different types of terminals are described in Chapter 4.4.
4.4 DIAGNOSIS AND ERROR HANDLING
Contents

4.4 DIAGNOSIS AND ERROR HANDLING

Summary of diagnosis options .......... 4-43
Diagnosis by program .................. 4-44
Diagnosis in master operating mode .... 4-44
Function module for field bus diagnosis .. 4-48
Diagnosis in slave operating mode ...... 4-50
Error words ............................. 4-50
4.4 DIAGNOSIS AND ERROR HANDLING

Summary of diagnosis options

The programmable valve terminal with SF 3 offers comprehensive and convenient options for diagnosis and error handling. The following options are available, irrespective of equipment on the terminal:

<table>
<thead>
<tr>
<th>Programmable valve terminal</th>
<th>Diagnosis options</th>
<th>LED Diagnosis bytes</th>
<th>Error word</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>IW 0.0...0.3</td>
<td>EW</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(local diagnosis)</td>
<td>(Multi-bit operand)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IW 0.4...0.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(field bus errors in transmission)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>IW0.8...0.11</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(field bus station diagnosis)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Field bus function module</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CFM 44</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Brief description</th>
<th>The red LED directly indicates transmission and station errors (for &quot;hard&quot; error characteristics).</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The status byte and the function modules must be read and evaluated, controlled by the user program.</td>
</tr>
<tr>
<td></td>
<td>The error word must be read and evaluated, controlled by the user program.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Rapid &quot;on site&quot; error recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Detailed error detection by program for faults in the electrical system and in the field bus.</td>
</tr>
<tr>
<td></td>
<td>Detailed error detection by program with program or operating system faults.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Detailed description</th>
<th>Chapter 3.4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Chapter 4.4</td>
</tr>
<tr>
<td></td>
<td>List of all signals in Appendix B</td>
</tr>
</tbody>
</table>

Fig. 4/16: Options for diagnosis and error handling
Diagnosis by program

The valve terminal SF 3, depending on whether it is in master or slave operating mode, offers different options for diagnosis by program.

Diagnosis in master operating mode

- Collective error messages I0.0.0 (errors in transmission) and I0.0.1 (station errors)
- Diagnosis bytes IW0.4 to IW0.7 (localising the station in the case of errors in transmission)
- Diagnosis bytes IW0.8 to IW0.11 (localising the station in the case of station errors)
- Special operands FU1 and FU2.
  To ensure that valve terminal SF 3 is compatible with SF 202, the special operands FU1 and FU2 make it possible to perform diagnosis of field bus stations 1 to 16.
- CFM 44 (function module for individual interrogation of status bytes of field bus stations)
- Local diagnosis IW0.0 to IW0.3 (refer to Chapter 3.4)

Recommendation:
Proceed in the program as follows:
- Continuously interrogate collective errors I0.0.0 and I0.0.1. In the event of errors: interrogate IW0.4...0.11 or FU1+2.
- Determine the number of the station
- Interrogate station status with CFM 44
4.4 Diagnosis and error handling

Master: Structure of diagnosis byte IW0.0:
(collective error messages, bit I0.0.0 to I0.0.1)

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>Local Diagnosis, refer to Chapter 3.4</td>
<td>Master mode Collective signal for field bus station errors/diagnosis. Refer to IW0.8 to IW0.11</td>
<td>Master mode Collective signal, transmission error. Refer to IW0.4 to IW0.7.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4/17: Structure of diagnosis byte IW0.0 in master operating mode

Errors in transmission

If I0.0.0 displays a "1" in master operating mode, the failed station can be localised using IW0.4...0.7 (stations 1...31).

Diagnosis byte IW0.4...0.7

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>x</td>
</tr>
<tr>
<td>0.4</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>x</td>
</tr>
<tr>
<td>0.5</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>0.6</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>0.7</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>8</td>
</tr>
</tbody>
</table>

x = Bit content without significance

Fig. 4/18: Errors in transmission - station is present in bit-code format
Station error/diagnosis

If I0.0.1 indicates a "1" in master operating mode, IW0.8...0.11 can be used to localise the station from which the diagnosis information is coming (1...31).

Diagnosis byte IW0.8...0.11

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>x</td>
</tr>
<tr>
<td>0.8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>x</td>
</tr>
<tr>
<td>0.9</td>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>0.10</td>
<td>23</td>
<td>22</td>
<td>21</td>
<td>20</td>
<td>19</td>
<td>18</td>
<td>17</td>
<td>16</td>
</tr>
<tr>
<td>0.11</td>
<td>31</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>24</td>
</tr>
</tbody>
</table>

x = Bit content without significance

*Fig. 4/19: Station error - station is present in bit-code format*

When performing diagnosis, please check:

- Whether the correct operating mode (master/slave) has been set.
- Whether the correct station number has been assigned.
- Whether the baud rate is the same for all field bus stations.
- Whether the field bus line has interference or an interrupt (open circuit).
Diagnosis with special operands FU1 and FU2

For compatibility reasons, special operands FU1 and FU2 are available on valve terminals SF 3 and SF 202. However, they only permit diagnosis of field bus stations 1 to 16. During diagnosis operations, the bit set to "1" indicates the number of the field bus station.

Using the number of the field bus station from IW0.4...0.7, IW0.8...0.11 or FU1/FU2, accurate diagnosis information about each of the slaves (station status) can be requested via CFM 44.
Function module for field bus diagnosis

In master operating mode, CFM 44 is well suited to reading the diagnosis information (station status) of all field bus stations.

1. Detecting an error/short circuit (s.c.) in field bus station with CFM 44
2. Resetting and, where applicable, setting local outputs by program.

Appendix B contains a detailed description of the function modules. This next section lists the return codes to CFM 44 of the different slaves:

<table>
<thead>
<tr>
<th>Valve terminal type 03/04-B/05</th>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>none</td>
<td>V_{out}</td>
<td>V_{Val}</td>
<td>V_{inp}</td>
<td>none</td>
<td>sc/o</td>
<td>none</td>
<td>none</td>
<td>none</td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cyclic station

Fig. 4/22: Status information of valve terminal type 03/04-B/05 (passive slave)

<table>
<thead>
<tr>
<th>Valve terminal type 02</th>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>none</td>
<td>V_{OUT}</td>
<td>V_{Val}</td>
<td>V_{INP}</td>
<td>SA1</td>
<td>SA0</td>
<td>none</td>
<td>none</td>
<td>none</td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Cyclic station

Fig. 4/23: Status information of valve terminal type 02 (passive slave)
### 4.4 Diagnosis and error handling

**FB-202**

<table>
<thead>
<tr>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>Output error</td>
<td>none</td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4/24: Status information of FB-202 (passive slave)

**SF 202 als Slave**

<table>
<thead>
<tr>
<th>Bit-no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>none</td>
<td>V_Out</td>
<td>V_Val</td>
<td>V_Inp</td>
<td>SA1</td>
<td>SA0</td>
<td>(type 02) and/or sc/o (type 03)</td>
<td>Run</td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4/25: Status information of SF 202 as slave (active slave)

**SF 3 als Slave**

<table>
<thead>
<tr>
<th>Bit-no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diagnosis information</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>none</td>
<td>Run</td>
<td>Stop</td>
</tr>
<tr>
<td>Signal status</td>
<td>0 or 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meaning</td>
<td>Signal status 0: no error</td>
<td>Signal status 1: error</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4/26: Status information of SF 3 (active slave)
**Diagnosis in slave operating mode**

- Collective fault messages I0.0.0 (errors in transmission, field bus failed)
- Local diagnosis IW0.0 to IW0.3 (refer to Chapter 3.4)

**Recommendation:**
Proceed as follows in the slave program:

- Continuously interrogate local diagnosis
- Continuously interrogate collective fault messages I0.0.0
- If the field bus is not adversely affected, transfer local diagnosis information via cyclic communication (configured diagnosis bytes OW0.4 to 0.15) to master.

**Note:**
The diagnosis information of bytes IW0.0...0.3 in the slave is more significant than the information which the master obtains via CFM 44 from SF 3 acting as a slave (refer to previous figure).

**Error words**
The red error display (ERROR) lights up as soon as the error status is not equal to 0. The error numbers are entered in the error word.

Appendix B contains a complete listing of all operating system error messages.
Programmable valve terminal

with

control block SF 3

Chapter 5: Description of analogue modules

For valve terminal type 03...05 only
Contents

5.1 SUMMARY OF ANALOGUE I/Os ........ 5-3
   Advantages of analogue I/Os ........ 5-3
   Component description of analogue
   input/output modules .................. 5-4

5.2 ASSEMBLY OF ANALOGUE I/Os ........ 5-7
   Analogue input/output modules ....... 5-9

5.3 INSTALLATION OF ANALOGUE I/Os ... 5-13
   Cable selection for analogue signals... 5-13
   Screening .............................. 5-14
   Connecting up analogue I/O modules ... 5-15
   Pin assignment of Festo
   proportional valves .................... 5-16
   Pin assignment of analogue
   voltage I/Os ........................... 5-17
   Pin assignment of analogue
   current I/Os ........................... 5-18

5.4 COMMISSIONING ANALOGUE I/Os ...... 5-21
   Basic points about addressing ........ 5-21
   Addressing analogue inputs/outputs .... 5-22
   Channel addressing analogue I/Os ...... 5-23
   Address allocation after
   extension/conversion .................... 5-25
   Activation of analogue I/Os with SF 3 .. 5-26
   Programming with FST 200 .............. 5-26
   Characteristics of analogue I/O
   modules after switching on ............ 5-27
   Function modules ........................ 5-27
5.5 DIAGNOSIS AND ERROR HANDLING
ANALOGUE I/Os ........................ 5-39
   Diagnosis on location ................ 5-39
   LED displays on analogue module PROP. .................. 5-39
   LED displays on analogue module for current I/Os (VIAU-I) .................. 5-40
   LED displays on analogue module for voltage I/Os (VIAU-U) .................. 5-41
   Remedying overload / short circuit of analogue voltage outputs .................. 5-42
   Overload / short circuit in actuator power supply .................. 5-43
   Diagnosis by program .................. 5-44

5.6 TECHNICAL DATA FOR ANALOGUE I/Os .................. 5-47
5.1 SUMMARY OF ANALOGUE I/Os
Contents

5.1 SUMMARY OF ANALOGUE I/Os
   Advantages of analogue I/Os . . . . . . . 5-3
   Component description of analogue
   input/output modules . . . . . . . . 5-4
5.1 SUMMARY OF ANALOGUE I/Os

Advantages of analogue I/Os
The modules with analogue inputs/outputs offer the following advantages:

- Optimised connection type option for Festo proportional valves (MPPE or MPYE).
- Individual and universal scope for adaptation to detection and processing of analogue signals.
- Detection and processing of analogue current and voltage signals.
- Very simple programming and diagnosis using specified Festo function modules.
- Subsequent extension/conversion possible.
- Inspected unit.

PLEASE NOTE
The modules with analogue inputs/outputs can only be mounted to type 03...05 valve terminals.
Component description of analogue input/output modules

Type 03...05 valve terminals comprise the following modules. The analogue input/output modules offer the following connection elements:

<table>
<thead>
<tr>
<th>Digit</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VIAU-U: Analogue voltage input/output module (0...10 V, 116 Hz)</td>
</tr>
<tr>
<td>2</td>
<td>VIAU-I: Analogue current input/output module (4...20 mA, 1116 Hz)</td>
</tr>
<tr>
<td>3</td>
<td>VIAP: Analogue current input/output module (4...20 mA, 100 Hz)</td>
</tr>
<tr>
<td>4</td>
<td>LEDs (for further information, refer to Chapter 5.5 &quot;Diagnosis&quot;)</td>
</tr>
<tr>
<td>5</td>
<td>Female connector for MPPE/MPYE (current-I/O + actuator power supply)</td>
</tr>
<tr>
<td>6</td>
<td>Female connectors for analogue current input + sensor power supply</td>
</tr>
<tr>
<td>7</td>
<td>Female connectors for analogue current I/Os + actuator power supply</td>
</tr>
<tr>
<td>8</td>
<td>Female connectors for analogue voltage I/Os + actuator power supply</td>
</tr>
<tr>
<td>9</td>
<td>Female connectors for analogue voltage input + sensor power supply</td>
</tr>
</tbody>
</table>

Fig. 5/1: Display and connection elements for analogue modules
5.2 ASSEMBLY OF ANALOGUE I/Os
5.2 Assembly of analogue I/Os

Analogue input/output modules .... 5-9
5.2 ASSEMBLY OF ANALOGUE I/Os

**WARNING**
Before starting assembly work, switch off the following:
- Compressed air supply
- Operating voltage supply to outputs (Pin 2)
- Operating voltage supply to electronics (Pin 1).

By doing this, you prevent:
- Uncontrolled movements of loosened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.

**CAUTION**
The components of the valve terminal contain parts which are vulnerable to electrostatic interference.
- For this reason, do not touch any contact surfaces on side connectors of these components.
- Comply with the handling specifications for electrostatically vulnerable components.

You thereby avoid damaging the valve terminal components.
Please note

The modules with analogue inputs/outputs are designed for mounting on type 03...05 valve terminals.

- Do not mount more than 12 electrical (digital or analogue) modules on each valve terminal.
- Mount modules ordered retrospectively behind the last electrical module before the left end plate.
- On valve terminals with AS-i master: The AS-i master must always be mounted as the outermost module directly before the left end plate.

You prevent:

- Addressing and system faults.
- A displacement of I/O addresses for I/O modules which have already been installed.

Please note

Treat all modules and components on valve terminals with care. Pay particular attention to the following points:

- Threaded connector without distortion and mechanical stress.
- Precise location of screws (otherwise damage to thread).
- Compliance with specified torques.
- Prevention of offset between the modules (IP65).
- Clean connection faces (prevention of leakage and contact faults).

In the case of retrospectively ordered modules and components, please note the assembly instructions enclosed in the product package.
Analogue input/output modules

To extend or convert the valve terminal, it is necessary to dismantle the screw-mounted terminal.

Disassemble (also refer to following figure):

- Completely unfasten screws in relevant modules. The modules are now only held together by the electrical plug connection.
- Pull the modules carefully off the electrical plug connections without tilting them.
- Replace the damaged seals.

Assemble (also refer to following figure):

- Insert a (new) seal on the right contact surface facing the node.
- Mount in accordance with following figure.
For information about assembly and earthing of the various electrical modules, please consult the relevant chapter for each module.

The Pneumatics Manual contains instructions about assembly of the pneumatic components.
5.3 INSTALLATION OF ANALOGUE I/OS
5.3 INSTALLATION OF ANALOGUE I/Os

- Cable selection for analogue signals: 5-13
- Screening: 5-14
- Connecting up analogue I/O modules: 5-15
- Pin assignment of Festo proportional valves: 5-16
- Pin assignment of analogue voltage I/Os: 5-17
- Pin assignment of analogue current I/Os: 5-18
5.3 INSTALLATION OF ANALOGUE I/Os

Cable selection for analogue signals

Recommendation:
Use the pre-assembled cables and plugs supplied by Festo.

With the following pre-packaged cables, you can connect the Festo proportional valves:

<table>
<thead>
<tr>
<th>Lead for</th>
<th>Part number</th>
<th>Type</th>
<th>Length in m</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proportional valve MPPE-..</td>
<td>163882</td>
<td>KVIA-MPPE-5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>163883</td>
<td>KVIA-MPPE-10</td>
<td>10</td>
</tr>
<tr>
<td>Proportional valve MPYE-..</td>
<td>161984</td>
<td>KVIA-MPYE-5</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>161985</td>
<td>KVIA-MPYE-10</td>
<td>10</td>
</tr>
</tbody>
</table>

For connection of analogue signal assemblies of other manufacturers, use the following cables:

<table>
<thead>
<tr>
<th>Cable for connection to analogue signal assemblies of other manufacturers</th>
<th>Festo part number</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>163960</td>
<td>KVIA-5</td>
</tr>
<tr>
<td></td>
<td>163961</td>
<td>KVIA-10</td>
</tr>
</tbody>
</table>
If you wish to use cables packaged in-house, only use the following cables and plugs for transmission of analogue signals:

- Screened cables
- Twisted-pair cables (input wires paired with input wires, output wires paired with output wires and supply wires)
- Plug with metal housing

Following plugs are suitable:
From Binder
Miniature round-plug connector, design in acc. with DIN 45322, 6-pin with gold contacts

Order no. (Binder):
99-5621-19-06(PG9)
99-5121-19-06(PG7)

Screening

**PLEASE NOTE**
Connect one side of cable shield on plug casing to analogue input/output module.

By doing this, you prevent:
- Faults caused by electromagnetic interference.
Connecting up analogue I/O modules

**WARNING**
Before any installation and maintenance work, switch off the following:
- Compressed air supply.
- Operating voltage supply for electronics (Pin 1).
- Operating voltage supply outputs/valves (Pin 2).

By doing this, you prevent:
- Uncontrolled movements of loosened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.

**PLEASE NOTE**
Ensure that unused lines of voltage inputs connected to the I/O plug of the analogue module are short-circuited.

By doing this, you prevent:
- Faults caused by electromagnetic interference.
Pin assignment of Festo proportional valves

For optimised use of Festo proportional valves, you can choose from three variants for your applications. The following figure illustrates the pin assignment of the module for proportional valves:

**Pin assignment VIAP-03-FB**

1. **II0+** Positive current input signal
2. **II0-** Negative current input signal
3. **O10+** Positive current output signal
4. **OGND** Current output signal (ground)
5. **24 Vp** 24 V actuator voltage supply
6. **0 V** 0 V actuator voltage supply
7. **Housing** Cable screen connection

---

**Fig. 5/3: Pin assignment for analogue modules VIAP-03-FB**
Pin assignment of analogue voltage I/Os

**CAUTION**
Please note the special assignment for female connector 2.

The following figure illustrates the pin assignment of sockets for voltage I/Os:

<table>
<thead>
<tr>
<th>Pin assignment VIAU-03-FB-U</th>
</tr>
</thead>
<tbody>
<tr>
<td>IUx+ Positive voltage input signal</td>
</tr>
<tr>
<td>IUx- Negative voltage input signal</td>
</tr>
<tr>
<td>OU0+ Positive voltage output signal</td>
</tr>
<tr>
<td>OGND Voltage output signal (ground)</td>
</tr>
<tr>
<td>24 Vp 24 V actuator voltage supply</td>
</tr>
<tr>
<td>24 VSen 24 V sensor voltage supply</td>
</tr>
<tr>
<td>0 V 0 V actuator/sensor voltage supply</td>
</tr>
<tr>
<td>Housing Cable screen connection</td>
</tr>
</tbody>
</table>

*Fig. 5/4: Pin assignment of analogue module VIAU-03-FB-U (voltage I/Os)*
Pin assignment of analogue current I/Os

The following figure illustrates the pin assignment of sockets for current I/Os:

Pin assignment VIAU-03-FB-I

<table>
<thead>
<tr>
<th>Pin assignment VIAU-03-FB-I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Il0-</td>
</tr>
<tr>
<td>Il0+</td>
</tr>
<tr>
<td>Il1-</td>
</tr>
<tr>
<td>Il2-</td>
</tr>
<tr>
<td>0 V</td>
</tr>
<tr>
<td>OI0+</td>
</tr>
<tr>
<td>24 V P</td>
</tr>
<tr>
<td>0 V</td>
</tr>
<tr>
<td>Thread for plug housing</td>
</tr>
</tbody>
</table>

Ilx+  Positive current input signal
Ilx-  Negative current input signal
OI0+  Positive current output signal
OGND  Current output signal (ground)
24 V Sen  24 V-sensor voltage supply
24 V P  24 V-actuator voltage supply
0 V  0 V-actuator/sensor voltage supply
Housing  Cable screen connection

Fig. 5/5: Pin assignment of analogue module VIAU-03-FB-I (current I/Os)
5.4 COMMISSIONING ANALOGUE I/Os
Contents

5.4 COMMISSIONING ANALOGUE I/Os

Basic points about addressing . . . . . . 5-21
Addressing analogue inputs/outputs . . . . . . 5-22
Channel addressing analogue I/Os . . . . . . . . 5-23
Address allocation after extension/conversion . . . . 5-25
Activation of analogue I/Os with SF 3 . . . . . . . . 5-26
Programming with FST 200 . . . . . . . . . . . . . . . . 5-26
Characteristics of analogue I/O modules after switching on . . . . . 5-27
Function modules . . . . . . . . . . . . . . . . . . 5-27
5.4 COMMISSIONING ANALOGUE I/Os

Basic points about addressing
Before programming, ascertain the precise number of analogue inputs/outputs. A modular valve terminal has different numbers of I/Os, depending on the equipment fitted.

The following table indicates the analogue I/Os required for programming, dependent on the module used:

<table>
<thead>
<tr>
<th>Type of module</th>
<th>Number of analogue I/Os</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analogue module, proportional</td>
<td>1 analogue input</td>
</tr>
<tr>
<td>VIAP-03-FB</td>
<td>1 analogue output</td>
</tr>
<tr>
<td>Analogue module, voltage inputs/outputs</td>
<td>3 analogue inputs</td>
</tr>
<tr>
<td>VIAU-03-FB-U</td>
<td>1 analogue output</td>
</tr>
<tr>
<td>Analogue module, current inputs/outputs</td>
<td>3 analogue inputs</td>
</tr>
<tr>
<td>VIAU-03-FB-I</td>
<td>1 analogue output</td>
</tr>
</tbody>
</table>
Addressing analogue inputs/outputs

PLEASE NOTE
The maximum number of analogue inputs/outputs is restricted by:

- The maximum permissible number of electrical modules (12)
- The maximum current consumption for outputs (10A).

The address allocation of inputs/outputs of a modular valve terminal depends on the equipment fitted on the terminal. Detailed information about addressing rules for valves and digital I/Os can be found in Chapter 2.4 "Addressing".

The rules described below apply to address allocation of valve terminals with analogue I/Os.
Channel addressing analogue I/Os

A maximum of 12 electrical modules can be fitted in the SF 3 valve terminal. The terminal may be fitted simply with analogue modules. The SF 3 valve terminal prepares the addresses required for analogue I/Os, referred to in the following section as input and output channels respectively. These are a maximum of:

- 36 input channels (12 * 3)
- 12 output channels (12 * 1).

Addressing analogue I/Os:

1. **The addresses of analogue I/Os are allocated separately from digital I/Os.**

1.1 The analogue inputs/outputs are addressed (counted) depending on the position of the analogue I/O modules.

1.2 The addressing (counting) of the analogue inputs and outputs respectively is performed separately.
   - Counting starts on node from **right to left**.
   - Address assignment in ascending numerical order.
   - On each individual module, counting runs from top to bottom.
The following figure illustrates the address allocation (numbering of analogue channels) when equipped with digital and analogue I/Os:

**Fig. 5/6: Address allocation (channel number) for analogue I/O modules**

AI = analogue inputs  
AO = analogue outputs
Address allocation after extension/conversion
A special feature of modular valve terminals is their flexibility. If the machine requirements alter, the equipment attached to the terminal can also be altered.

CAUTION
With subsequent extensions or conversions to the terminal, channel numbers can be altered for analogue inputs and outputs. This definitely applies in the following cases:
• If additional analogue input/output modules are inserted between the node and existing analogue input/output modules.
• If existing analogue input/output modules (e.g. proportional modules) are replaced with universal I/O modules or vice versa.
Activation of analogue I/Os with SF 3

Programming with FST 200
To create the user programs and program modules, there is a choice of two programming languages: Statement List (STL) und Ladder Diagram (LDR). Please consult the handbook for a full description of prerequisites for the programming languages and programming techniques:

- FESTO Software Tools Statement List and Ladder Diagram of the SF 3 FST 200 Manual

The following sections contain specific information about analogue I/O modules:

- Characteristics after switching on
- Activation using SF 3 function modules
- Diagnosis/error handling
Characteristics of analogue I/O modules after switching on

When the operating voltage is switched on, the following data access options in the analogue I/O modules are available:

<table>
<thead>
<tr>
<th>Data area</th>
<th>Access option</th>
<th>Data output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input data</td>
<td>Immediate</td>
<td>-</td>
</tr>
<tr>
<td>Output data</td>
<td>- Voltage</td>
<td>0 V</td>
</tr>
<tr>
<td>- Current</td>
<td></td>
<td>0 mA</td>
</tr>
</tbody>
</table>

Only after function module 61 has been called is it possible to output from a module with current outputs, a valid current output value in the nominal range between 4...20 mA.

Function modules

- Function modules form part of the operating system. To activate the analogue I/O modules, the following function modules have been implemented in SF 3:
  - CFM 60: Reading of analogue values
  - CFM 61: Output of analogue values
  - CFM 63: Diagnosis of analogue modules
Input format
THEN CFM 60
WITH <P1>

Parameter
P1 = input channel number (0,...,35)

Return code parameters
P1 (FU32)=
{-1} processing successful
{0} processing faulty
P2 (FU33)=
digital input value
or error number (100, 101, 112), if P1 = 0

Example for current/voltage values

<table>
<thead>
<tr>
<th>Input current</th>
<th>Numerical value (input current = 16 mA x numerical value / 4096) + 4 mA</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 19.992 mA</td>
<td>4094</td>
</tr>
<tr>
<td>12.000 mA</td>
<td>2048</td>
</tr>
<tr>
<td>...</td>
<td>2 (lowest resolution)</td>
</tr>
<tr>
<td>&lt; 4.0078 mA</td>
<td>0</td>
</tr>
<tr>
<td>&lt; 4.000 mA</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input voltage</th>
<th>Numerical value (input voltage = 10 V x numerical value / 4096)</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 9.9975 V</td>
<td>4095</td>
</tr>
<tr>
<td>5.000 V</td>
<td>2048</td>
</tr>
<tr>
<td>...</td>
<td>1 (lowest resolution)</td>
</tr>
<tr>
<td>&lt; 0.00244 V</td>
<td>0</td>
</tr>
<tr>
<td>&lt; 0.000 V</td>
<td></td>
</tr>
</tbody>
</table>
**Example for current/voltage values**

### Input format

**THEN**

CFM 61

**WITH** &lt;P1&gt;

**WITH** &lt;P2&gt;

### Parameter

P1 = output channel no. (0,..., 11)
P2 = output value (0 ... 4095)

### Return code parameters

- **P1 (FU32)=**
  - {-1} processing successful
  - {0} processing faulty
- **P2 (FU33)=**
  - &lt;without significance&gt;
  - error number (100, 101, 102, 112), if P1 = 0

### Output current

<table>
<thead>
<tr>
<th>Numerical value (numerical value = 4096 x (output current-4 mA)/ 16 mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 19.996 mA</td>
</tr>
<tr>
<td>10.000 mA</td>
</tr>
<tr>
<td>4.0078 mA</td>
</tr>
<tr>
<td>&lt; 4.000 mA</td>
</tr>
</tbody>
</table>

### Output voltage

<table>
<thead>
<tr>
<th>Numerical value (numerical value = 4096 x (output voltage)/ 10 V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt; 9.9975 V</td>
</tr>
<tr>
<td>5.000 V</td>
</tr>
<tr>
<td>0.00244 V</td>
</tr>
<tr>
<td>&lt; 0.000 V</td>
</tr>
</tbody>
</table>
Using this module enables you to receive diagnosis information about the analogue I/O modules for your valve terminal.

The module provides 6 diagnosis functions. The diagnosis result, depending on the diagnosis function used, is prepared in special function units FU33 to FU35. By using appropriate programming techniques, you can either receive collective information or information about individual channels.

**Input format**

THEN    CFM 63
WITH <P1>
WITH <P2>

**Parameters**

P1 = channel number
   (0...11) for output channels
   (0...35) for input channels
   or -1 for all available channels
P2 = diagnosis function (0...5)

**Return code parameters**

P1 (FU32)=  {-1} processing successful
            (0) processing faulty
P2 (FU33)=  diagnosis result
            error number (100, 101, 102, 112), if P1 = 0

With diagnosis function 2,4,5
P3 (FU34)=  diagnosis result
The diagnosis results described in the following section are only obtained if function module CFM 63 reports successful processing (FU32 = -1).

**Diagnosis function 0**

Overload/short circuit (s.c.) in analogue voltage outputs; representation by channel

**Diagnosis result in FU33 (FU32 = -1)**

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0</td>
</tr>
<tr>
<td></td>
<td>0 0 0 0 O11 O10 O9 O8 O7 O6 O5 O4 O3 O2 O1 O0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value O0...O11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Voltage output O... not overloaded/not short-circuited</td>
</tr>
<tr>
<td>1</td>
<td>Voltage output O... overloaded, output value cannot be issued</td>
</tr>
</tbody>
</table>

Data bits from channels which are not available and from channels which have not been selected supply the logic value "0".

Data bits greater than D11 also always supply the logic value "0". They are not relevant because no more than a maximum of 12 analogue output channels can be available (O0...O11).
Diagnosis function 1

Overloading / short circuit / undervoltage of DC 24 V actuator supply voltage for existing analogue I/O modules; representation by module.

Diagnosis result in FU33 (FU32 = -1)

<table>
<thead>
<tr>
<th>Module numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D6 D5 D4 D3 D2 D1 D0</td>
<td>0 0 0 0 V11 V10 V9 V8 V7 V6 V5 V4 V3 V2 V1 V0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value V0...V11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>24 V actuator power supply of module V... not overloaded, not short circuited, no undervoltage</td>
</tr>
<tr>
<td>1</td>
<td>Overload, short circuit and/or undervoltage of DC 24 V actuator supply voltage of module V...</td>
</tr>
</tbody>
</table>

Data bits from unavailable modules and unselected modules supply the logic value "0".

Data bits greater than D11 also always supply the logic value "0". They are not relevant, since a maximum of 12 analogue I/O modules can be available (V0...V11).
Diagnosis function 2

Wire break diagnosis of analogue current inputs, input current < 2 mA; representation by channel

Diagnosis result in FU33...FU35 (FU32 = -1)

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15  D14  D13  D12  D11  D10  D9  D8  D7  D6  D5  D4  D3  D2  D1  D0</td>
<td></td>
</tr>
<tr>
<td>15   14   13   12   11   10   9   8   7   6   5   4   3   2   1   0</td>
<td></td>
</tr>
</tbody>
</table>

FU34:

<table>
<thead>
<tr>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>I31  I30  I29  I28  I27  I26  I25  I24  I23  I22  I21  I20  I19  I18  I17  I16</td>
</tr>
</tbody>
</table>

FU35:

<table>
<thead>
<tr>
<th>Log. value I0...I35</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Current input signal &gt;= 2 mA</td>
</tr>
<tr>
<td>1</td>
<td>Current input signal &lt; 2 mA</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0".

Channel numbers greater than I35 also always supply the logic value "0". These are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
Diagnosis function 3

Wire break diagnosis of analogue current outputs: idling/excessive burden resistance; representation by channel

**PLEASE NOTE**
Idling is not recognised until after one set of analogue values has been issued.

Diagnosis result in FU33 (FU32 = -1)

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15  D14  D13  D12  D11  D10  D9  D8  D7  D6  D5  D4  D3  D2  D1  D0</td>
<td></td>
</tr>
<tr>
<td>0    0    0    0    O11  O10  O9   O8   O7   O6   O5   O4   O3   O2   O1   O0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value O0...O11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No idling</td>
</tr>
<tr>
<td>1</td>
<td>Idling of current output; output value cannot be issued</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0".

Data bits greater than D11 also always supply the logic value "0". They are not relevant because no more than a maximum of 12 analogue output channels can be available (O0...O11).
Diagnosis function 4

Determining the analogue voltage input; representation by channel. This diagnosis function makes it possible to inspect the valve terminal configuration with regard to the fitted voltage inputs.

Diagnosis result in FU33...FU35 (FU32 = -1)

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td></td>
</tr>
<tr>
<td>I15 I14 I13 I12 I11 I10 I9 I8 I7 I6 I5 I4 I3 I2 I1 I0</td>
<td></td>
</tr>
</tbody>
</table>

FU34:

| D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0 |
| I31 I30 I29 I28 I27 I26 I25 I24 I23 I22 I21 I20 I19 I18 I17 I16 |

FU35:

| D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |

<table>
<thead>
<tr>
<th>Log. value I0...I35</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No voltage input</td>
</tr>
<tr>
<td>1</td>
<td>Voltage input</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0". Channel numbers greater than I35 also always supply the logic value "0". These are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
Diagnosis function 5

Determining the analogue current inputs; representation by channel. This diagnosis function makes it possible to inspect the valve terminal configuration with regard to the fitted current inputs.

Diagnosis result in FU33...FU35 (FU32 =-1)

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
<td>D14</td>
</tr>
<tr>
<td>115</td>
<td>114</td>
</tr>
<tr>
<td>I15</td>
<td>I14</td>
</tr>
</tbody>
</table>

FU34:

<table>
<thead>
<tr>
<th>Channel numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
</tr>
<tr>
<td>131</td>
</tr>
</tbody>
</table>

FU35:

<table>
<thead>
<tr>
<th>Channel numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>035</td>
</tr>
</tbody>
</table>

**Log. value I0...I35**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0". Channel numbers greater than I35 also always supply the logic value "0". These are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
5.5 Diagnosis and Error Handling of Analogue I/Os
Contents

5.5 DIAGNOSIS AND ERROR HANDLING
ANALOGUE I/Os

Diagnosis on location ............... 5-39
LED displays on analogue
module PROP.......................... 5-39
LED displays on analogue
module for current I/Os (VIAU-I) . . 5-40
LED displays on analogue
module for voltage I/Os (VIAU-U) . . 5-41
Remedying overload/short circuit
of analogue voltage outputs ........ 5-42
Overload/short circuit in actuator
power supply ...................... 5-43
Diagnosis by program ............... 5-44
5.5 Diagnosis and Error Handling Analogue I/Os

Diagnosis on location

LED displays on analogue module PROP

<table>
<thead>
<tr>
<th>LED</th>
<th>Operating condition</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Lit</td>
<td>Ready for operation</td>
</tr>
<tr>
<td>Red</td>
<td>Lit</td>
<td>Overload/short circuit on the actuator power supply</td>
</tr>
<tr>
<td>Dark</td>
<td>Dark</td>
<td>Hardware error</td>
</tr>
<tr>
<td></td>
<td>Dark</td>
<td>Actuator power supply ok</td>
</tr>
</tbody>
</table>
## LED displays on analogue module
for current I/Os (VIAU-I)

<table>
<thead>
<tr>
<th>LED</th>
<th>Operational condition</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Green LED] Lit</td>
<td>Ready for operation</td>
<td>None</td>
</tr>
<tr>
<td>![Green LED] Dark</td>
<td>Hardware error</td>
<td>Servicing required</td>
</tr>
<tr>
<td>![Yellow LED] Dark</td>
<td>Current signal range 4...20 mA</td>
<td>---</td>
</tr>
<tr>
<td>![Red LED] Lit</td>
<td>Overload/short circuit on the actuator power supply</td>
<td>Remedy overload / short circuit (s.c.)</td>
</tr>
<tr>
<td>![Red LED] Dark</td>
<td>Actuator power supply ok</td>
<td>None</td>
</tr>
</tbody>
</table>
### LED displays on analogue module for voltage I/Os (VIAU-U)

<table>
<thead>
<tr>
<th>LED</th>
<th>Operational condition</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="red" alt="Green LED" /></td>
<td>Lit</td>
<td>Ready for operation</td>
</tr>
<tr>
<td><img src="red" alt="Dark LED" /></td>
<td>Dark</td>
<td>Hardware error</td>
</tr>
<tr>
<td><img src="red" alt="Yellow LED" /></td>
<td>Lit</td>
<td>Voltage signal range 0...10 V</td>
</tr>
</tbody>
</table>
| ![Red LED](red) | Lit | Overload/short circuit (s.c.)  
- on the actuator power supply  
- on voltage output | Remedy overload / short circuit (s.c.) |
| ![Dark LED](red) | Dark | Ok (no overload/s.c.) | None |
Remedying overload / short circuit of analogue voltage outputs

The analogue voltage outputs are continuously monitored for overloading and/or short circuits (s.c.). If a short circuit occurs or if the output is overloaded, proceed as follows to continue processing analogue I/O data:

<table>
<thead>
<tr>
<th>Errors</th>
<th>Response</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overload/short circuit on the voltage output</td>
<td>• 0 V-output on analogue voltage output</td>
<td>1. Remedy overload/s.c.</td>
</tr>
<tr>
<td></td>
<td>• Red LED lit</td>
<td>2. Output of 0 V on relevant analogue output</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. Output of desired value.</td>
</tr>
</tbody>
</table>

Supply a 0 V-value (digital valve 0) to the voltage output via function module 61. This enables you to restore complete functionality for the output of analogue voltage values at this output.
Overload / short circuit in actuator power supply

**PLEASE NOTE**
Please note the different error handling methods for modules VIAP-.. and VIAU-.. after an overload has ended.

The 24 V actuator voltage supply is protected internally by an electronic fuse. If an error occurs, the following takes place:

<table>
<thead>
<tr>
<th>Overloading</th>
<th>Response</th>
<th>Error handling</th>
</tr>
</thead>
</table>
| Analogue module VIAP-.. | • Switch off actuator supply voltage for the duration of an overload/short circuit  
                          | • Red LED lit                                                            | • Remedy overload/short circuit; after a thermal recovery period has elapsed, the actuator supply voltage is switched on again |
| Analogue module VIAU-.. | • Switch off actuator supply voltage                                      | 1. Remedy overload/short circuit; 2. Switch off and switch 24 V power supply back on at node (Pin 2) |
                          | • Red LED lit                                                            |                                                                                   |
Diagnosis by program
Diagnosis information about function module CFM 63 can be called up by program. For detailed information consult Section 5.4.
5.6 TECHNICAL DATA FOR ANALOGUE I/Os
## 5.6 Technical data for analogue I/Os

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>5-47</td>
</tr>
<tr>
<td>Operating voltage electronics</td>
<td>5-48</td>
</tr>
<tr>
<td>Actuator voltage supply</td>
<td>5-48</td>
</tr>
<tr>
<td>Analogue current inputs</td>
<td></td>
</tr>
<tr>
<td>VIAP-.. module</td>
<td>5-49</td>
</tr>
<tr>
<td>Analogue current outputs</td>
<td></td>
</tr>
<tr>
<td>VIAP-.. module</td>
<td>5-49</td>
</tr>
<tr>
<td>Analogue voltage inputs</td>
<td></td>
</tr>
<tr>
<td>VIAU-...U module</td>
<td>5-50</td>
</tr>
<tr>
<td>Analogue voltage outputs</td>
<td></td>
</tr>
<tr>
<td>VIAU-...U module</td>
<td>5-50</td>
</tr>
<tr>
<td>Analogue current inputs</td>
<td></td>
</tr>
<tr>
<td>VIAU-...I module</td>
<td>5-51</td>
</tr>
<tr>
<td>Analogue current outputs</td>
<td></td>
</tr>
<tr>
<td>VIAU-...I module</td>
<td>5-51</td>
</tr>
<tr>
<td>Electromagnetic compatibility (EMC)</td>
<td>5-52</td>
</tr>
<tr>
<td>Electrical isolation</td>
<td>5-52</td>
</tr>
<tr>
<td>Protection against electric shock</td>
<td>5-52</td>
</tr>
<tr>
<td>Maximum permissible potential differences</td>
<td>5-52</td>
</tr>
</tbody>
</table>
5.6 TECHNICAL DATA FOR ANALOGUE I/Os

<table>
<thead>
<tr>
<th>General</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Protection class</td>
<td>IP65</td>
</tr>
<tr>
<td>(acc. to DIN 40050)</td>
<td></td>
</tr>
<tr>
<td>Temperature for</td>
<td></td>
</tr>
<tr>
<td>• Operation</td>
<td>-5 °C...+50 °C</td>
</tr>
<tr>
<td>• Storage/transport</td>
<td>-20 °C...+70 °C</td>
</tr>
<tr>
<td>Rel. humidity</td>
<td>Max. 95 % (25 °C, without condensation)</td>
</tr>
<tr>
<td>Chem. resistance</td>
<td>Refer to Festo pneumatics catalogue (resistance table)</td>
</tr>
</tbody>
</table>
### Operating voltage electronics
(Pin 1 – operating voltage connection)

- **Nominal value** (polarity-safe)  
  DC 24 V
- **Tolerance**  
  ± 25%  
  (DC18 V...30 V)
- **Residual ripple**  
  4 Vpp
- **Current consumption of analogue I/O modules for 24 V (max. analogue input/output currents)**  
  64 mA
- **Fuse protection of power supply for inputs/sensors**  
  Internal 2 A, slow-blow
- **Bridge time in event of drop in logic voltage**  
  Min. 20 ms

### Actuator voltage supply
(Pin 2 – operating voltage connection)

- **Nominal value** (polarity-safe)  
  External fuse required  
  DC 24 V (typ. 10 A)
- **Tolerance**  
  ± 10%  
  (DC 21.6 V...26.4 V)
- **Residual ripple**  
  4 Vpp
- **Current consumption (at 24 V)**  
  14.5 mA quiescent current
- **Actuator power supply**  
  VIAP-.. VIAU-..  
  0.5 A 1.0 A  
  1.0 A 1.0 A  
  2.5 V 2.5 V
- **Max. permissible medium continuous loading**  
  0.5 A 1.0 A
- **Max. permissible momentary peak current**  
  1.0 A 1.0 A
- **Max. voltage drop relative to input feed point in node at I_load = 1 A**  
  2.5 V 2.5 V
### Analogue current inputs VIAP-.. module

<table>
<thead>
<tr>
<th>Current input</th>
<th>Differential input</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Signal range</td>
<td>4 ... 20 mA</td>
</tr>
<tr>
<td>• Resolution</td>
<td>11 Bit</td>
</tr>
<tr>
<td>• Number of units</td>
<td>2048</td>
</tr>
<tr>
<td>• Absolute accuracy</td>
<td>0.45 %</td>
</tr>
<tr>
<td>• Input resistance</td>
<td>50 Ohm</td>
</tr>
<tr>
<td>• Max. permissible input current</td>
<td>65 mA</td>
</tr>
<tr>
<td>(destruction limit)</td>
<td></td>
</tr>
<tr>
<td>Input signal</td>
<td>100 Hz</td>
</tr>
<tr>
<td>Angular frequency</td>
<td></td>
</tr>
<tr>
<td>Linearity</td>
<td></td>
</tr>
<tr>
<td>• Differential non-linearity</td>
<td>2 LSB</td>
</tr>
<tr>
<td>• Integral (absolute) non-linearity</td>
<td>3 LSB</td>
</tr>
</tbody>
</table>

### Analogue current outputs VIAP-.. module

<table>
<thead>
<tr>
<th>Current output range</th>
<th>4...20 mA</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Signal range</td>
<td>12 Bit</td>
</tr>
<tr>
<td>• Resolution</td>
<td>4096</td>
</tr>
<tr>
<td>• Number of units</td>
<td>0.5 %</td>
</tr>
<tr>
<td>• Absolute accuracy</td>
<td>≤ 250 Ohm</td>
</tr>
<tr>
<td>• Load resistance (burden)</td>
<td></td>
</tr>
<tr>
<td>Linearity</td>
<td></td>
</tr>
<tr>
<td>• Differential non-linearity</td>
<td>2 LSB</td>
</tr>
<tr>
<td>• Integral (absolute) non-linearity</td>
<td>4 LSB</td>
</tr>
<tr>
<td>• Integral (absolute) non-linearity</td>
<td></td>
</tr>
</tbody>
</table>
### Analogue voltage inputs VIAU-..-U module

<table>
<thead>
<tr>
<th>Voltage input</th>
<th><strong>Differential input</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal range</td>
<td>0...10 V</td>
</tr>
<tr>
<td>Resolution</td>
<td>12 bits</td>
</tr>
<tr>
<td>Number of units</td>
<td>4096</td>
</tr>
<tr>
<td>Absolute accuracy</td>
<td>0.4 %</td>
</tr>
<tr>
<td>Input resistance</td>
<td>≥ 20 kOhm</td>
</tr>
<tr>
<td>Max. permissible input voltage (destruction limit)</td>
<td>30 V</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input signal</th>
<th><strong>Angular frequency</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Differential input</td>
<td>116 Hz</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Linearity</th>
<th><strong>Differential non-linearity</strong></th>
<th><strong>Integral (absolute) non-linearity</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2 LSB</td>
<td>3 LSB</td>
</tr>
</tbody>
</table>

### Analogue voltage outputs VIAU-..-U module

<table>
<thead>
<tr>
<th>Voltage output</th>
<th><strong>Short-circuit-proof</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal range</td>
<td>0...10 V</td>
</tr>
<tr>
<td>Resolution</td>
<td>12 bit</td>
</tr>
<tr>
<td>Number of units</td>
<td>4096</td>
</tr>
<tr>
<td>Absolute accuracy</td>
<td>0.45 %</td>
</tr>
<tr>
<td>Load resistance (burden)</td>
<td>≥ 3.3 kOhm</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Linearity</th>
<th><strong>Differential non-linearity</strong></th>
<th><strong>Integral (absolute) non-linearity</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2 LSB</td>
<td>3 LSB</td>
</tr>
</tbody>
</table>
### Analogue current inputs VIAU-..-I module

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal range</td>
<td>4...20 mA</td>
</tr>
<tr>
<td>Resolution</td>
<td>11 bit</td>
</tr>
<tr>
<td>Number of units</td>
<td>2048</td>
</tr>
<tr>
<td>Absolute accuracy</td>
<td>0.45 %</td>
</tr>
<tr>
<td>Input resistance</td>
<td>50 Ohm</td>
</tr>
<tr>
<td>Max. permissible input current (destruction limit)</td>
<td>65 mA</td>
</tr>
<tr>
<td>Input signal</td>
<td>116 Hz</td>
</tr>
<tr>
<td>Angular frequency</td>
<td></td>
</tr>
</tbody>
</table>

### Linearity

<table>
<thead>
<tr>
<th>Non-linearity</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Differential</td>
<td>2 LSB</td>
</tr>
<tr>
<td>Integral (absolute)</td>
<td>3 LSB</td>
</tr>
</tbody>
</table>

### Analogue current outputs VIAU-..-I module

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal range</td>
<td>4...20 mA</td>
</tr>
<tr>
<td>Resolution</td>
<td>12 bit</td>
</tr>
<tr>
<td>Number of units</td>
<td>4096</td>
</tr>
<tr>
<td>Absolute accuracy</td>
<td>0.5 %</td>
</tr>
<tr>
<td>Load resistance (burden)</td>
<td>( \leq 250 ) Ohm</td>
</tr>
</tbody>
</table>

### Linearity

<table>
<thead>
<tr>
<th>Non-linearity</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Differential</td>
<td>2 LSB</td>
</tr>
<tr>
<td>Integral (absolute)</td>
<td>4 LSB</td>
</tr>
</tbody>
</table>
### Electromagnetic compatibility (EMC)

- Interference emitted
- Immunity against interference

Tested as per DIN EN 61000-6-4 (industry)\(^1\)
Tested as per DIN EN 61000-6-2 (industry)

\(^1\) The component is intended for industrial use.

### Electrical isolation

- Analogue input and internal 5 V logic
- Analogue output and internal 5 V logic

There is **no** electrical isolation
- Between analogue input and analogue output
- When using the I/O plug connected to 24 V\(_{\text{Sen}}\) and/or 24 V\(_{\text{P}}\)

### Protection against electric shock

Protection against direct and indirect contact as per IEC/DIN EN 60204-1 by means of PELV circuits (Protective Extra-Low Voltage)

### Maximum permissible potential differences

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analogue inputs against one another</td>
<td>1 V</td>
</tr>
<tr>
<td>Analogue inputs/outputs against one another</td>
<td>1 V</td>
</tr>
<tr>
<td>Analogue inputs/outputs against 0 V operating voltage and/or PE (pneumatic-electric) (central grounding point)</td>
<td>30 V</td>
</tr>
</tbody>
</table>
Programmable valve terminal with SF 3 control block

Chapter 6: Description of AS-I master

AS-i master only for valve terminal type 03...05
PLEASE NOTE
- This description supplements the documentation on your valve terminal with the information you require for the AS-i master/bus systems.
- In sections with the adjacent pictogram, please refer also to the important information in the Electronics manual for your valve terminal.

Contents

6.1 USER INSTRUCTIONS AND SYSTEM SUMMARY .......... 6-1
6.1.1 Important user instructions ............... 6-3
Instructions on this manual ............... 6-3
6.1.2 System summary .................. 6-4
General remarks ....................... 6-4
AS-i – General system data ............... 6-5
AS-i bus systems ....................... 6-6
Basic variants of an AS-i bus system ...... 6-9
AS-i master on a valve terminal .......... 6-10
Component description ............... 6-11
Functional description ............... 6-12

6.2 ASSEMBLY ....................... 6-15
6.2.1 Mounting the components ................ 6-17
AS-i master ....................... 6-18
Assembly ....................... 6-18
Earthing ....................... 6-18
Wall mounting and top-hat rail mounting .... 6-18
6.3 INSTALLATION

6.3.1 General connection techniques
   General remarks
   Cable selection on the AS-i bus
   Cable selection for diagnosis
   interface
   Connecting the cables
to the plugs/sockets.
   Connecting the flat cables
   using the insertion method
   Connecting up flat cable
to M12 round plug connectors

6.3.2 AS-i master
   Connecting the diagnosis interface
   Configuration plug
   Connecting the AS-i bus
   EMERGENCY STOP
   characteristics on AS-i bus system
   Pin assignment of AS-i interface
   Connecting the AS-i power pack
   General remarks
   Location on the AS-i bus
   Advantages of Festo AS-i
   mains power assembly

6.4 COMMISSIONING

6.4.1 Before commissioning
   General remarks
   Check list "before commissioning"
   Aspects of an AS-i which relate to
   configuration
   Addressing AS-i slaves
   Addressing AS-i slaves
   with AS-i addressing device
   Addressing the AS-i slaves
   with FST 200
   (without AS-i addressing device)
6.4.2 Basic points on commissioning
Switching on the complete system
Summary of AS-i address
range/periphery and assignment
Tips relating to AS-i address survey
Festo tables (summary of address
range and assignment)
Festo table for control block SF 3
Festo table for control block SF 3
Assignment of AS-i I/O ⇔ PLC-I/O

6.4.3 Commissioning with FST 200
Requirements for commissioning
AS-i operands
Summary of AS-i address space
Addressing on the AS-i bus
Configuration of the AS-i bus
system with FST 200
Menu "Planning AS-i slaves"
(configuration)
Function for "SET POINT-ACTUAL
comparison"
Menu "Assign/modify
AS-i slave address"
Menu "SF 3 Online Mode"
Transmitting AS-i slave
parameters

6.4.4 Programming the startup/
performance
Characteristics of valve terminal
with AS-i master after Power ON
Characteristics of valve terminal
with AS-i master during operation
Characteristics in the absence
of operating voltage
What to do when a slave
error occurs
6.5 DIAGNOSIS AND ERROR HANDLING ............... 6-83
6.5.1 Summary of diagnosis options ............ 6-85
6.5.2 Diagnosis on location ..................... 6-86
   LED displays on AS-i master ............... 6-86
   LED displays on the AS-i slaves ......... 6-87
   FST 200 .................................. 6-88
6.5.3 AS-i-specific error handling ............. 6-91
   Localisation of defective slaves ......... 6-91
   Addressing by means of the
   AS-i addressing device ................. 6-92
   Addressing with the FST 200 ............ 6-93
   Addressing by means of
   autoprogramming ...................... 6-94
6.5.4 Error handling with CFM 35 ............. 6-96

6.6 TECHNICAL DATA ............................ 6-99
6.1 USER INSTRUCTIONS AND
SYSTEM SUMMARY
Contents

6.1 USER INSTRUCTIONS AND SYSTEM SUMMARY

6.1.1 Important user instructions ............ 6-3
   Instructions on this manual ............. 6-3

6.1.2 System summary ...................... 6-4
   General remarks ....................... 6-4
   AS-i – General system data ............ 6-5
   AS-i bus systems ..................... 6-6
   Basic variants of an AS-i bus system .. 6-9
   AS-i master on a valve terminal ...... 6-10
   Component description ................. 6-11
   Functional description ............... 6-12
6.1.1 Important user instructions

Instructions on this manual
In sections with this pictogram, please note the important information in Chapters 3 und 4.

Text designations
• The listing point identifies activities which can be performed in any order.
1. Numerals indicate activities which must be performed in the specified order.
   – Hyphens indicate general activities.

The following product specific abbreviations are used in this description:

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS-i</td>
<td>Actuator sensor interface</td>
</tr>
<tr>
<td>PLC</td>
<td>Programmable logic controller. Valve terminals can also be purchased with an integral PLC.</td>
</tr>
<tr>
<td>D_in</td>
<td>Digital input</td>
</tr>
<tr>
<td>D_out</td>
<td>Digital output</td>
</tr>
<tr>
<td>AS-i I/O</td>
<td>Input and/or output on the AS-i bus system</td>
</tr>
</tbody>
</table>

Further information and basic knowledge about the AS-i bus system and its specification can be found in the relevant literature, e.g. in the book: "AS-i – The Actuator Sensor Interface For Automation" by Werner Kriesel and Otto W. Made- lung (Hanser-Verlag, English edition ISBN 3-446-18265-9).
6.1.2 System summary

General remarks
AS-i, the actuator sensor interface, is a bus system at the lowest level of the automation system hierarchy. This bus system combines the following advantages:

- Low-cost networking of individual binary sensors and actuators.
- Flexible installation in remotely distributed units.
- Very simple and fast (real-time capability) protocol.
- Data and power can be transmitted down one cable.

AS-i can therefore be used as an easy-to-install variant or as a sub-system for existing field bus systems.

Fig. 6/1: AS-i on the lowest level of automation technology pyramid
AS-i – General system data

The following table indicates the most important system data of an AS-i bus system:

<table>
<thead>
<tr>
<th>System data of an AS-i bus system</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>max. current on AS-i bus</td>
<td>2 A per rung</td>
</tr>
<tr>
<td>max. current per slave</td>
<td>max. 100 mA per slave</td>
</tr>
<tr>
<td>AS-i bus cable length</td>
<td>max. 100 m per rung, multiple extension possible due to repeater</td>
</tr>
<tr>
<td>Number of slaves</td>
<td>max. 31 AS-i slaves per master</td>
</tr>
<tr>
<td>Number of connectable sensors/actuators (I/Os)</td>
<td>up to 4 inputs and/or 4 outputs per slave (max. 124 inputs and/or 124 outputs per AS-i master)</td>
</tr>
<tr>
<td>Cycle time</td>
<td>&lt; 5 ms when fully fitted</td>
</tr>
<tr>
<td>Network structure</td>
<td>star, line or tree structure</td>
</tr>
<tr>
<td>Transmission medium</td>
<td>unscreened two-wire line, power and data on one cable</td>
</tr>
</tbody>
</table>

Fig. 6/2: Table of most important AS-i system data
AS-i bus systems
The use of AS-i-capable pneumatics is cost-effective in many cases. Examples of application:
- In critical transmission paths.
- The activation of small valve groups.
- With actuators which are widely distributed in the machine/unit.

Application fields include:
- Conveyor belts
- Conveyor technology
- Assembly and packaging lines

The following diagram illustrates the requirements for constructing and commissioning an AS-i bus system:
Fig. 6/3: System summary of an AS-i bus system with SF 3 as master
The construction of AS-i bus systems is simple and flexible. The following diagram illustrates the permissible bus structures and network topologies:

- Star
- Line (with/without branch lines)
- Tree

Fig. 6/4: Possible topologies for AS-i bus system
Basic variants of an AS-i bus system
The following masters or higher-order control units can be used for an AS-i bus system:

- PLC with AS-i master module or valve terminal with built-in PLC (control block SF 3) and AS-i master
- PC card with AS-i master module
- Connection of an existing field bus system to a control unit (e.g. via a Festo valve terminal as coupler/gateway)

Fig. 6/5: Possible basic variants of AS-i bus system
AS-i master on a valve terminal

The Festo AS-i master offers coupler/gateway functionality for valve terminals using field bus protocols (such as e.g. Festo field bus, PROFIBUS-DP, INTERBUS-S). This means that AS-i bus systems can be used upwards in the hierarchy of automation technology. In addition, valve terminals with built-in PLC (control block SF 3) are available. This means that independent control of AS-i bus systems is possible.

Fig. 6/6: Festo AS-i master as coupler/gateway to field bus or with built-in PLC (control block SF 3)
Component description

The modular valve terminals comprise individual modules. Different functions, connection, display and operating elements are assigned to every module. On the AS-i master, you will find the following components:

### No. Significance

1. Diagnosis interface V.24/RS 232  
   (do not use in conjunction with node SF 3)
2. Yellow LED
3. Inscription field of AS-i master
4. AS-i logo
5. AS-i bus connection with AS-i cable socket  
   (included in delivery)
6. Green LED
7. Configuration plug  
   (do not use in conjunction with node SF 3)

**Fig. 6/7:** Display, connector and operating components of AS-i master

**PLEASE NOTE**

The configuration plug must not be used in conjunction with the SF 3 node.
Functional description
The Festo AS-i master performs the following functions:

- Controls data transmission on the AS-i bus system.
- Controls data exchange with the node on the valve terminal.
- Adapts the addresses of all AS-i slaves to the addressing diagram of the field bus protocol or the PLC.
- Makes possible certain important commissioning operations of the AS-i bus system, such as
  - configuring AS-i bus system
  - setting parameters of AS-i slaves
  - automatic readdressing of replaced AS-i slaves
  - alteration/assignment of AS-i slave address.
- Makes it possible to diagnose the AS-i bus system.
1. Configuration, diagnosis and commissioning with PC via the diagnosis/programming interface DIAG on control block SF 3
2. Exchanging data with the node
3. AS-i bus line
4. Data transmission to all AS-i bus users

**Fig. 6/8: Function summary of AS-i master**
6.2 ASSEMBLY
6.2 ASSEMBLY

6.2.1 Mounting the components . . . . . . . . . . 6-17
  AS-i master . . . . . . . . . . . . . . . . . . . . . . 6-18
  Assembly . . . . . . . . . . . . . . . . . . . . . . . 6-18
  Earthing . . . . . . . . . . . . . . . . . . . . . . . 6-18
  Wall mounting and top-hat rail
  mounting . . . . . . . . . . . . . . . . . . . . . . . 6-18
6.2.1 Mounting the components

**WARNING**

Before starting assembly work, switch off the following:
- Compressed air supply
- Operating voltage supply to outputs (Pin 2)
- Operating voltage supply to electronics (Pin 1)

By doing this, you prevent:
- Uncontrolled movements of loosened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.

**CAUTION**

The components of the valve terminal contain parts which are vulnerable to electrostatic interference.
- For this reason, do not touch any electrical contact surfaces on the side plugs of these components.
- Comply with the handling specifications for electrostatically vulnerable components.

This enables you to prevent damage to the valve terminal components.

**PLEASE NOTE**

Comply with assembly instructions in Chapter 2.
AS-i master

**PLEASE NOTE**
- Always place the AS-i master directly alongside the left end plate.
- Do not mount more than 12 electrical modules (incl. AS-i master).

Assembly
The AS-i master is secured with three M4 cap screws.

Earthing
The AS-i master is electrically connected to the other components by means of prefitted spring contacts.

Wall mounting and top-hat rail mounting
When calculating the total weight (general rule in valve terminal manual), always add approx. 600 g for the AS-i master.
6.3 INSTALLATION
Contents

6.3 INSTALLATION ......................... 6-19

6.3.1 General connection techniques ...... 6-21
  General remarks ...................... 6-22
  Cable selection on the AS-i bus ...... 6-22
  Cable selection for diagnosis interface .... 6-23
  Connecting the cables to the plugs/sockets.. 6-23
  Connecting the flat cables using the insertion method .... 6-24
  Connecting up flat cable to M12 round plug connectors..... 6-25

6.3.2 AS-i master ......................... 6-26
  Connecting the diagnosis interface ... 6-26
  Configuration plug .................... 6-26
  Connecting the AS-i bus ............. 6-27
  EMERGENCY STOP characteristics on AS-i bus system .... 6-27
  Pin assignment of AS-i interface .... 6-29
  Connecting the AS-i power pack .... 6-30
  General remarks .................... 6-30
  Location on the AS-i bus .......... 6-31
  Advantages of Festo AS-i mains power assembly .......... 6-34
6.3.1 General connection techniques

WARNING
Before any installation and maintenance work, switch off the following power sources:
• Compressed air supply
• Operating voltage for electronics (Pin 1)
• Operating voltage for outputs/valves (Pin 2)
• Operating voltage for AS-i bus system (AS-i power pack)
• Auxiliary power supply on AS-i bus system

By doing this, you prevent:

- Uncontrolled movements of unfastened flexible tubes.
- Accidental movements of connected actuator units.
- Undefined switching modes of the electronics.
General remarks
Special flat cables with mechanical coding have been defined for the AS-i bus systems. Power and signals can be transmitted at the same time through these cables. These flat cables make it possible to obtain the correct polarity and simple connection of all AS-i bus users. The bonding complies with IP65 using special AS-i plugs connected by the insertion method. Alternatively, they can also be connected using PG threaded connectors. Ensure that suitable gaskets and seals are used (IP65).

If you use other cables, always check the polarity of the AS-i interface.

Cable selection on the AS-i bus
For the AS-i bus and the auxiliary power supply, Festo can supply a flat cable in either yellow or black. Use the cables in the following manner:

<table>
<thead>
<tr>
<th>Flat cable</th>
<th>Parts no./type</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS-i flat cable</td>
<td>18 940</td>
<td>AS-i bus</td>
</tr>
<tr>
<td>yellow</td>
<td>VASI 1.5-Y-100</td>
<td></td>
</tr>
<tr>
<td>Flat cable</td>
<td>18 941</td>
<td>Auxiliary power supply can be switched off</td>
</tr>
<tr>
<td>black</td>
<td>VASI 1.5-Z-100</td>
<td>separately or for consumers with higher</td>
</tr>
<tr>
<td></td>
<td></td>
<td>current consumption</td>
</tr>
</tbody>
</table>

PLEASE NOTE
These versions of the AS-i flat cable are not suitable for use as drag chains.
Cable selection for diagnosis interface

PLEASE NOTE
- The diagnosis interface of the AS-i master must **not** be connected up in conjunction with the node "control block SF 3".
- Only use the diagnosis interface on the node.

This prevents configuration and addressing errors.

Connecting the cables to the plugs/sockets

Once you have selected suitable cables, connect these to plugs/sockets in accordance with the following steps.
Connecting the flat cables using the insertion method

Using the example of the Festo AS-i socket, the following section illustrates how an AS-i flat cable is connected up using the insertion method:

Note: In position 2, the wire can still be moved.

Fig. 6/9: Connect the AS-i flat cable using the insertion method – example: Festo
Connecting up flat cable to M12 round plug connectors

To connect to M12 round plug connectors, you must use appropriate seals due to the special geometry of the AS-i flat cable. These seals must be inserted in the threaded connectors or you must use special plugs to comply with IP65.

Recommendation:
Use the Festo socket AS-i SD-PG-M12, Parts No. 18789 (e.g. on branch lines) or AS-i SD-FK-M12, Parts No. 18788 (e.g. for extension/contact pickup of lines). These have appropriate special seals.

Fig. 6/10: Connecting the AS-i flat cable to M12 round plug connector – example: Festo
6.3.2 AS-i master

Connecting the diagnosis interface

**PLEASE NOTE**
- The diagnosis interface of the AS-i master must **not** be used in conjunction with the node "control block SF 3".
- Only use the diagnosis interface on the node.

This prevents configuration and addressing errors.

**Configuration plug**

If you still have a configuration plug for the AS-i master from other projects, you will not require this plug in conjunction with the "control block SF 3" node, because the FST 200 offers many options to facilitate commissioning the entire system, incl. AS-i master.
Connecting the AS-i bus

EMERGENCY STOP characteristics on AS-i bus system

A bus system should not be switched off completely when EMERGENCY STOP is activated. This means that important functions remain operational, e.g.

- Data transmission to the bus users
- Display of process status settings.

Since power and data transmission on the AS-i bus system take place on a shared cable, outputs connected to the bus at a hardware level cannot be switched off separately.

CAUTION

- Outputs supplied with electrical power via the AS-i bus system cannot be switched off at hardware level by means of the EMERGENCY STOP circuit.
- If the AS-i master fails during operation, the slave outputs remain set.
- Whenever AS-i voltage to a slave is interrupted, the outputs are switched off.
Before connecting up the operating voltage, check which outputs on your AS-i bus system have to be switched off separately. In these cases, use AS-i slaves which have an auxiliary power supply. Take this auxiliary power supply into account when planning your EMERGENCY STOP circuit.

<table>
<thead>
<tr>
<th>No.</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valve terminal with control block SF 3</td>
</tr>
<tr>
<td>2</td>
<td>AS-i master</td>
</tr>
<tr>
<td>3</td>
<td>AS-i power pack with built-in data disconnection</td>
</tr>
<tr>
<td>4</td>
<td>24 V auxiliary power pack</td>
</tr>
<tr>
<td>5</td>
<td>Relay for switching off the auxiliary power supply</td>
</tr>
<tr>
<td>6</td>
<td>AS-i slave without auxiliary power supply</td>
</tr>
<tr>
<td>7</td>
<td>AS-i slave with auxiliary power supply</td>
</tr>
</tbody>
</table>

Fig. 6/11: Example - AS-i slaves with and without EMERGENCY STOP
Pin assignment of AS-i interface

**PLEASE NOTE**
Only connect to the AS-i interface slaves which comply with the AS-i specifications.

This prevents malfunctions of AS-i master and slaves.

To connect up the AS-i interface, you require the Festo AS-i socket AS-i SD-FK (included in delivery). Missing AS-i sockets can be ordered by quoting Parts No. 18785. Please comply with specified max. torque.

*Fig. 6/12: Pin assignment of AS-i interface (bus connection) and Festo AS-i socket*
Connecting the AS-i power pack

**WARNING**

To provide reliable electrical isolation, use an AS-i power pack with isolating transformer acc. to EN 60742 (DIN/VDE 0551, IEC 742) with at least 4 kV insulation resistance.

**General remarks**

Special power packs for AS-i bus systems make it possible to transmit power and signals simultaneously on one cable. Watch for the AS-i logo when selecting equipment.

Recommendation:

Use the Festo AS-i mains power assembly AS-i CNT-115/230VAC, Parts No. 18 949. This offers the following advantages:

- provides reliable isolation in accordance with EN 60742
- complies with EMC guideline of EU (CE symbol)
- equipped with a built-in AS-i power supply module with data disconnection
- equipped with an additional 24 V output (6 A) for circuits which must be switched off separately and slaves with higher current consumption.

On the following pages, take note of the recommendations for locating the AS-i power pack and the other points relating to design of the bus system.
Location on the AS-i bus

PLEASE NOTE
The AS-i power pack can fundamentally be situated at any location on the AS-i bus.

However, note the following limit values and operating situations. Possible resultant restrictions on the location of an AS-i power pack:

- The maximum overall length of the AS-i bus system per rung (incl. branch lines) is 100 m. By using repeaters, additional 100 m rungs can be fitted. An additional AS-i power pack is required for each rung.

<table>
<thead>
<tr>
<th>No.</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AS-i master</td>
</tr>
<tr>
<td>2</td>
<td>AS-i power pack with built-in data disconnection</td>
</tr>
<tr>
<td>3</td>
<td>Repeater</td>
</tr>
<tr>
<td>4</td>
<td>AS-i slave</td>
</tr>
</tbody>
</table>

Fig. 6/13: Example – of AS-i bus system with repeater
The maximum permissible current consumption per slave amounts to 100 mA.

Slaves with higher current consumption (e.g. large valves/solenoid coils) must be supplied by a separate 24 V power pack (auxiliary power supply). All Festo slaves with higher current consumption have an appropriate additional 24 V connection and must be connected to an auxiliary power supply. With these slaves, EMERGENCY STOP concepts can also be implemented.

<table>
<thead>
<tr>
<th>No.</th>
<th>Significance</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AS-i master</td>
</tr>
<tr>
<td>2</td>
<td>AS-i power pack with built-in data disconnection</td>
</tr>
<tr>
<td>3</td>
<td>Repeater</td>
</tr>
<tr>
<td>4</td>
<td>AS-i slave</td>
</tr>
<tr>
<td>5</td>
<td>24 V power pack for auxiliary power supply</td>
</tr>
<tr>
<td>6</td>
<td>Contacts for switching off the auxiliary power supply</td>
</tr>
<tr>
<td>7</td>
<td>AS-i slave with higher current consumption and 24 V auxiliary power supply</td>
</tr>
</tbody>
</table>

Fig. 6/14: Example - AS-i slaves with higher current consumption and 24 V auxiliary power supply
- The max. current consumption of the AS-i bus system per rung amounts to 2 A.
  Recommendation:
  - Select a favourable arrangement of the AS-i power pack within the AS-i bus system.
  - Arrange slaves with higher current consumption beside the power pack.

If in doubt, examine current distribution and voltage characteristics at the furthest point of the AS-i bus system (rung) carefully.
Advantages of Festo AS-i mains power assembly

The Festo AS-i mains power assembly provides the AS-i with operating voltage in accordance with the AS-i specification and an additional 24 V power supply (output voltage raised to 26 V, to compensate for voltage drop). This 24 V power supply is suitable for EMERGENCY STOP circuits or as an auxiliary power supply for devices with higher current consumption. The technical data of the mains power assembly are listed in Chapter 6.6.

Fig. 6/15: Festo AS-i mains power assembly, part No. 18949
6.4 COMMISSIONING
Contents

6.4 COMMISSIONING ............................. 6-35

6.4.1 Before commissioning ................. 6-37
  General remarks ....................... 6-37
  Check list "before commissioning" ...... 6-40
  Aspects of an AS-i which relate to
  configuration ............................ 6-42
  Addressing AS-i slaves .................. 6-44

6.4.2 Basic points on commissioning ...... 6-49
  Switching on the complete system .... 6-49
  Summary of AS-i address range/
  periphery and assignment ............. 6-51
  Tips relating to AS-i address survey .... 6-52
  Festo tables (summary of address
  range and assignment) ................. 6-53

6.4.3 Commissioning with FST 200 ......... 6-58
  Requirements for commissioning ...... 6-59
  AS-i operands .......................... 6-60
  Summary of AS-i address space ...... 6-61
  Addressing on the AS-i bus .......... 6-62
  Configuration of the AS-i bus
  system with FST 200 ................... 6-63
  Menu "Planning AS-i slaves"
  (configuration) ....................... 6-64
  Function for
  "SET POINT- ACTUAL comparison" .... 6-66
  Menu
  "Assign/modify AS-i slave address" .... 6-69
  Menu "SF 3 Online Mode" ............... 6-73
  Transmitting AS-i slave parameters .. 6-75

6.4.4 Programming the startup/performance 6-76
  Characteristics of valve terminal
  with AS-i master after Power ON ...... 6-76
  Characteristics of valve terminal
  with AS-i master during operation .... 6-79
  Characteristics in the absence of
  operating voltage ...................... 6-79
  What to do when a slave error occurs .. 6-80
6.4.1 Before commissioning

General remarks

Before commissioning an AS-i bus system, a few basic points must be taken into consideration. The following points should assist you in avoiding errors and in taking account of important points.

- Addressing AS-i slaves:
  All AS-i slaves are supplied with slave address 00, unless otherwise indicated. Before connecting up the AS-i bus, you must assign all slaves a slave address (1...31). For this, you can use a tool such as the Festo addressing device. Always take note of the instructions in the documentation relating to the slave in question.

CAUTION

During commissioning, any slave still bearing the slave address 00 is ignored by the master. This means that the slave is not recorded in the ACTUAL list, the inputs are not read in and the outputs are not set.
There is no error message from the master.

If a different, correctly-addressed slave fails during commissioning or during operation (with identical IO and ID code), autoprogramming is performed on the slave with address 00 (refer to Chapter 6.5.3 "Addressing by means of auto programming") and all I/Os are activated immediately. In the menu "Assigning/altering AS-i slave address" in FST 200, you have the alternative option of readdressing the slave by using the software.
Assigning slave addresses:
The slave addresses must only be assigned once on an AS-i bus system. Assigning duplicate addresses leads to errors during commissioning and to undefined process states.

**CAUTION**

If slave addresses are duplicated, outputs may display undefined switching states during commissioning and operation. Outputs may be set in parallel fashion or be reset. 
For this reason avoid duplicating the assignment of slave addresses.

The slave addresses can fundamentally be assigned in any desired order: they do not have to be issued in consecutive order.
Planning the power supply:
The voltage supply to AS-i slaves is always provided by the yellow bus cable. This cable is connected to the AS-i power supply unit for this purpose.

Slaves with higher current consumption or with additional power supply should be supplied from a separate connection and via separate power pack.

The voltage supply must either be switched on in parallel fashion or in the following sequence:

1. AS-i bus
2. AS-i master (via nodes on the terminal)
3. Higher-order control units and systems.
Check list "before commisioning"

<table>
<thead>
<tr>
<th>Check list</th>
</tr>
</thead>
<tbody>
<tr>
<td>□ Are the addresses of all slaves set to 1...31?</td>
</tr>
<tr>
<td>□ Have slave addresses been clearly assigned?</td>
</tr>
<tr>
<td>□ Has an EMERGENCY STOP circuit been provided where required for all relevant slaves in accordance with prevailing specifications?</td>
</tr>
<tr>
<td>□ Has a separate (auxiliary) power supply been connected to all slaves with higher current consumption?</td>
</tr>
<tr>
<td>□ Is it possible to switch on the power supply in parallel fashion or in the correct sequence?</td>
</tr>
<tr>
<td>□ Have AS-i specifications been complied with?</td>
</tr>
<tr>
<td>- max. current consumption on AS-i bus 2 A</td>
</tr>
<tr>
<td>- max. cable length without repeater 100 m</td>
</tr>
<tr>
<td>□ Has the AS-i bus been configured in FST 200?</td>
</tr>
<tr>
<td>- has a NOMINAL list been compiled with AS-i slave addresses, ID codes and I/O codes?</td>
</tr>
</tbody>
</table>

Fig. 6/16: Table/check list "Before commissioning"

The following diagram shows a selection of some AS-i components required for commisioning.
Fig. 6/17: Selecting a few AS-i components for commissioning
Aspects of an AS-i which relate to configuration

For unique identification of an AS-i user, AS-i specifications and AS-i profiles have been defined. In an open bus system, all manufacturers are bound by these specifications. An AS-i slave is uniquely identified by means of an ID code and I/O code. These codes are saved by the manufacturer in permanent memory and are printed on the rating plate. These state the following:

- **ID code**
  The ID code provides information about the slave type (e.g. simple I/O module, intelligent sensor, motor switch). The ID code also defines which profiles (e.g. data and parameter bits) are used by this slave. The ID code therefore makes it possible to interchange slaves made by various different manufacturers and is usually quoted in hexadecimal form (e.g. F\text{H}).

**PLEASE NOTE**
- The AS-i master recognises all ID codes automatically and issues the correct protocol variant in response.
- Slaves with ID code F_{\text{H}}:
  - do not conform to any AS-i profile
  - and are individually defined by manufacturer and user.
I/O code
The I/O code defines how the four data bits from the AS-i protocol are to be used. Together with the ID code, this makes the AS-i slave uniquely identifiable. The following table indicates the defined I/O codes; the Festo slaves are indicated as examples.

<table>
<thead>
<tr>
<th>I/O code</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>Festo example</th>
</tr>
</thead>
<tbody>
<tr>
<td>0H</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>AS-i 4I module</td>
</tr>
<tr>
<td>1H</td>
<td>O</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>AS-i 2I2O module</td>
</tr>
<tr>
<td>2H</td>
<td>I/O</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>AS-i combi socket 4O,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i valve terminal, AS-i 4O module</td>
</tr>
<tr>
<td>3H</td>
<td>O</td>
<td>O</td>
<td>I</td>
<td>I</td>
<td>AS-i combi socket 2O2I or 1O2I</td>
</tr>
<tr>
<td>4H</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>5H</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>6H</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>7H</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>8H</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>9H</td>
<td>I</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>AH</td>
<td>I/O</td>
<td>O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>BH</td>
<td>I</td>
<td>I</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>CH</td>
<td>I/O</td>
<td>I/O</td>
<td>O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>DH</td>
<td>I</td>
<td>I</td>
<td>I</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>EH</td>
<td>I/O</td>
<td>I/O</td>
<td>I/O</td>
<td>O</td>
<td></td>
</tr>
<tr>
<td>FH</td>
<td>TRI</td>
<td>TRI</td>
<td>TRI</td>
<td>TRI</td>
<td></td>
</tr>
</tbody>
</table>

Key to figures:
I   Binary input from process
O   Binary output to process
I/O Bi-directional characteristics of port
TRI No configuration and/or no configuration identified after reset

Fig. 6/18: AS-i specification of permissible I/O codes

Recommendation:
Slaves with 2 inputs and 2 outputs can bear the following I/O codes:
- 3H (e.g. Festo 2I2O module)
- B_H (e.g. Festo combi socket 2O2I).
These slaves are therefore not mutually interchangeable.
Addressing AS-i slaves

The Festo AS-i master, in conjunction with the SF 3 control block and the FST 200, offers two options for addressing the AS-i slaves:

- **with** AS-i addressing device, i.e. the AS-i slave addresses are set on every slave using the AS-i addressing device.

- **without** AS-i addressing device. For this, the AS-i slave addresses of all slaves are set individually in the AS-i bus configurator in FST 200.

After this, a comparison can take place between SET POINT and ACTUAL configurations, the ACTUAL configuration can be adopted as the new SET POINT configuration and commissioning of the AS-i bus system can be completed.

Recommendation:

Autoprogramming makes automatic configuration of any defective/replaced slaves possible on the operational AS-i bus. Autoprogramming is described in Chapter 6.5.3 "Diagnosis and error handling".
Addressing AS-i slaves with AS-i addressing device

For this, you require an AS-i addressing device, e.g. the Festo Addressing device, part no. 18959.

PLEASE NOTE
Address each individual AS-i slave before the installation on the AS-i bus.

Proceed as follows (also refer to following diagram):

1. Connect the AS-i addressing device to the AS-i slave (where applicable, refer to description of your AS-i slaves).
2. Read the current address of the slave (with new slaves, usually address #00)
3. Allocate the slave an AS-i address which has not yet been assigned (1...31), address #07 in the diagram.
4. Install the addressed slave on the AS-i bus.
Fig. 6/19: Addressing the AS-i slaves with an AS-i addressing device
Addressing the AS-i slaves with FST 200
(without AS-i addressing device)

For this, you do not require an AS-i addressing device.
Call up the AS-i bus configurator on the FST 200
(refer to the manual for the FST 200, Chapter 8).

PLEASE NOTE
Address every single AS-i slave immediately after it has been installed.

Proceed as follows (also refer to following diagram):

1. Connect the slave to the AS-i bus (where applicable, refer to description of your appropriate AS-i slave).
   The slave should have slave address 0.
2. Start the AS-i bus configurator on your FST 200.
   The PC must be connected to the diagnosis interface on the SF 3 control block. In the menu "Assigning/altering AS-i slave address",
   you can now assign a new address to the slave with address 0.
3. Select slave address 0 with F2.
4. Use the cursor keys to select an AS-i address which has not yet been assigned (1...31) and press F3 to readdress the new slave to this address (in diagram #07).
5. Now install the next slave which has to be addressed.
Fig. 6/20: Addressing the AS-i slaves with FST 200 (without AS-i addressing device)
6.4.2 Basic points on commissioning

Switching on the complete system

**PLEASE NOTE**
- Comply with switching instructions in Chapters 2, 3 and 4.
- The following commissioning steps assume knowledge of the previous chapters, especially 6.4.1 "Before commissioning".

Before you switch on the complete system, the configuration and commissioning of the AS-i master/AS-i bus system must be completed. Switch on the complete system as follows:

- **Shared power supply:**
  Shared power supply for control system, all field bus stations and all AS-i participants (incl. AS-i power pack) using a central (combination) power pack and/or a central switch.

- **Separate power supply:**
  If the control system, field bus stations and AS-i participants have separate power supplies, switch them on in the following sequence:
  1. Switch on all AS-i slaves (AS-i power pack and, where applicable, the auxiliary power supply).
  2. Switch on all field bus stations.
  3. Switch on higher-order control unit.
Assignment of AS-i I/Os to the FST I/O operands

PLEASE NOTE
AS-i I/Os of the terminal:
- The AS-i I/Os always occupy the address range IW/OW16...31.
- Note that the AS-i master also provides four AS-i status bits for diagnosis.
  The symmetry of AS-i addressing means that four additional AS-i outputs are always reserved (which cannot however be used).
- These AS-i status bits also occupy four AS-i input addresses and four AS-i output addresses (IW/OW16.0...16.3).
- The AS-i status bits (4I, 4O) are always assigned automatically if an AS-i master is available.

Extension/conversion:
- Note that the AS-i I/Os always occupy the address range IW/OW16...31, irrespective of the equipment fitted on the terminal.
Summary of AS-i address range/periiphery and assignment

All I/O areas (local I/Os, AS-i I/Os) represent individual blocks during configuration. These blocks are independent of one another. This means the following number of I/Os can be used on terminals with an AS-i master:

<table>
<thead>
<tr>
<th>Node</th>
<th>Local I/Os per terminal</th>
<th>AS-i I/Os per terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control block SF 3</td>
<td>max. 128 I &lt;br&gt; max. 128 O</td>
<td>max. 124+4 I* &lt;br&gt; max. 124+4 O*</td>
</tr>
</tbody>
</table>

* incl. AS-i status bits

Recommendation:
- During configuration of an AS-i slave, four input and four output addresses are always assigned in the AS-i address range, regardless of the I/O code of the slave.
- The AS-i master assigns four input addresses for the AS-i status bits as well as four output addresses (which cannot however be used).
Tips relating to AS-i address survey

- Obtain a precise overview of the address space and the I/O assignment of all AS-i slaves in order to prevent addressing errors (refer to example 1 and the Festo tables which follow it).

- The Festo tables which follow example 1 serve as reproducible copies and help you with planning and configuration of the required address range. They also assist with assignment of individual slaves to the I/O bytes (I/O words) in your control system.
Festo tables (summary of address range and assignment)

The Festo tables which follow example 1 help you to plan your address range. They also help to assign individual slaves to the I/O bytes (I/O words) on the control unit.

Example 1 (working with the Festo tables):
Four AS-i slaves with the AS-i addresses 1, 2, 3 and 7 and the master are entered in the I/O table to establish the address range and the relevant physical inputs and outputs accurately. Following this, the AS-i I/Os can be assigned accurately to the I/Os in the control unit.

Fig. 6/21a: Example for working with the following Festo tables
Continuation of example 1:

Entries in the summary table (AS-i I/O address range):

<table>
<thead>
<tr>
<th>PLC byte</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>I4</td>
<td>I3</td>
<td>I2</td>
<td>I1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AS-i slave 1 (IO code 0H)</td>
<td></td>
<td></td>
<td></td>
<td>AS-i master</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>I2</td>
<td>I1</td>
<td>X</td>
<td>O1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AS-i slave 3 (IO code B0H)</td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 2 (IO code 31H)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>O4</td>
<td>O3</td>
<td>O2</td>
<td>O1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>AS-i slave 7 (IO code 88H)</td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 4</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 6/21b: Example of a summary in AS-i I/O address range
Continuation of example 1:

Entries in the assignment table:
IW/OW 16...31: Configuration of control block SF 3
D0...D3: Data bits of the AS-i slaves

<table>
<thead>
<tr>
<th>AS-i slave 1 (IO code 3H)</th>
<th>AS-i master</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>I4 I3 I2 I1</td>
<td>x x x x</td>
</tr>
<tr>
<td>four AS-i status bits (I)</td>
<td>four outputs reserved (symmetry)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AS-i slave 3 (IO code BH)</th>
<th>AS-i slave 2 (IO code 3H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>17</td>
</tr>
<tr>
<td>I2 I1 x x</td>
<td>x x I2 I1</td>
</tr>
<tr>
<td>O2 O1 x x</td>
<td></td>
</tr>
</tbody>
</table>

1 = output on slave physically not available (combi socket 2I1O)
x = bit content not relevant/not usable

<table>
<thead>
<tr>
<th>AS-i slave 5</th>
<th>AS-i slave 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>18</td>
</tr>
<tr>
<td>x x x x x</td>
<td>x x x x</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AS-i slave 7 (IO code 8H)</th>
<th>AS-i slave 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>O4 O3 O2 O1</td>
<td>x x x x</td>
</tr>
</tbody>
</table>

Copy the following tables for later calculations; enlarge them to make them easier to fill in.
**Festo table for control block SF 3**

Summary of the AS-i I/O address range

<table>
<thead>
<tr>
<th>PLC byte</th>
<th>Bit 7</th>
<th>Bit 6</th>
<th>Bit 5</th>
<th>Bit 4</th>
<th>Bit 3</th>
<th>Bit 2</th>
<th>Bit 1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS-i slave 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i master</td>
</tr>
<tr>
<td>16</td>
<td>AS-i slave 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>six status bits</td>
<td>AS-i slave 2</td>
</tr>
<tr>
<td>17</td>
<td>AS-i slave 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 4</td>
</tr>
<tr>
<td>18</td>
<td>AS-i slave 7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 6</td>
</tr>
<tr>
<td>19</td>
<td>AS-i slave 9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 8</td>
</tr>
<tr>
<td>20</td>
<td>AS-i slave 11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 10</td>
</tr>
<tr>
<td>21</td>
<td>AS-i slave 13</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 12</td>
</tr>
<tr>
<td>22</td>
<td>AS-i slave 15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 14</td>
</tr>
<tr>
<td>23</td>
<td>AS-i slave 17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 16</td>
</tr>
<tr>
<td>24</td>
<td>AS-i slave 19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 18</td>
</tr>
<tr>
<td>25</td>
<td>AS-i slave 21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 20</td>
</tr>
<tr>
<td>26</td>
<td>AS-i slave 23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 22</td>
</tr>
<tr>
<td>27</td>
<td>AS-i slave 25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 24</td>
</tr>
<tr>
<td>28</td>
<td>AS-i slave 27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 26</td>
</tr>
<tr>
<td>29</td>
<td>AS-i slave 29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 28</td>
</tr>
<tr>
<td>30</td>
<td>AS-i slave 31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>AS-i slave 30</td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 6/22: Summary of AS-i I/O address range
### Festo Table for Control Block SF 3
Assignment of AS-i I/O ⇔ PLC-I/O

<table>
<thead>
<tr>
<th>PLC byte</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW</td>
<td>OW</td>
<td>Bit 7</td>
<td>Bit 6</td>
<td>Bit 5</td>
<td>Bit 4</td>
<td>Bit 3</td>
<td>Bit 2</td>
<td>Bit 1</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AS-i slave 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 6/23: Assignment for AS-i I/O ⇔ PLC I/O*
6.4.3 Commissioning with FST 200

The FST 200 contains an AS-i bus configurator and extended AS-i functions in online operation. This makes it easy to configure and commission the AS-i network. Following options are available:

- Project planning and configuration of AS-i network on PC.
- Conducting of a SETPOINT-ACTUAL comparison.
- Loading of configuration data from PC to the SF 3 control block.
- Addressing of slaves connected to the AS-i network.
- Testing the AS-i inputs and outputs in online operation.
- Presentation of AS-i error messages.
- Transmission of slave parameters to appropriate designs of AS-i slaves (via CFMs; e.g. for testing sensors).

PLEASE NOTE
The configuration and commissioning of the AS-i master and/or AS-i bus system with FST 200 is always performed via the diagnosis interface on the SF 3 control block.

The SF 3 control block then exchanges data internally with the AS-i master.
Requirements for commissioning

In addition to the commissioning steps listed in Chapters 3 and 4, the following additional steps are required for AS-i bus systems:

- Addressing of all AS-i slaves
- Configuring the AS-i master/AS-i bus system (compiling a SETPOINT list, SETPOINT-AC-TUAL comparison etc.)
- Commissioning the AS-i bus system

In addition to the 128 local inputs and the 128 local outputs, valve terminals with SF 3 control blocks are capable of activating an additional 128 AS-i inputs and 128 AS-i outputs. For the addressing of local I/Os, the rules listed in Chapters 2 and 3 always apply. Supplementary notes for the addressing of AS-i I/Os are summarised in the following section.

Additional summaries are listed as follows:

- CP description, if CP circuitry is available in addition to an AS-i master.
- Appendix B, overview of all operands (I/Os).
### AS-i operands

<table>
<thead>
<tr>
<th>Operands</th>
<th>Number</th>
<th>Designation</th>
<th>Parameters</th>
<th>Comments</th>
<th>Remanent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td></td>
<td>I</td>
<td>{16...31}, {0...7}</td>
<td>one-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Input words</td>
<td>128</td>
<td>IW</td>
<td>{16...31}</td>
<td>multi bit operand</td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Outputs</td>
<td>128</td>
<td>O</td>
<td>{16...31}, {0...7}</td>
<td>one-bit operand</td>
<td>no</td>
</tr>
<tr>
<td>Output words</td>
<td>16</td>
<td>OW</td>
<td>{16...31}</td>
<td>multi bit operand</td>
<td></td>
</tr>
</tbody>
</table>

Recommendation:
- The AS-i inputs I16.0...I16.3 are assigned automatically by the AS-i master and act as AS-i status bits. They contain diagnosis information about the AS-i bus (refer to Chapter 6.5).
- The AS-i outputs O16.0...O16.3 are assigned automatically by the AS-i master for reasons of symmetry and are not usable.
- The AS-i slaves are addressed starting with operands I16.4 and/or O16.4.

Fig. 6/24: Available AS-i operands on valve terminal with control block SF 3

---

**PLEASE NOTE**

Not all AS-i inputs and outputs are physically present and connectable.
Summary of AS-i address space

The available AS-i address space is independent of the operating mode of the programmable valve terminal SF 3 (standalone, master or slave) and always occupies address space/FST operands IW/OW16...31.

The addresses of the local I/Os and the local diagnosis are also always independent. The following figure provides an overview:

---

**Fig. 6/25: Summary of AS-i address space and local address space**
Addressing on the AS-i bus

The AS-i I/Os are always addressed in the same way as the local I/Os. The following diagram illustrates the addressing of four AS-i slaves with the AS-i slave addresses 1, 2, 3 and 7 (also refer to the Festo tables in Chapter 6.4.2):

![Diagram showing the addressing of four AS-i slaves with the AS-i slave addresses 1, 2, 3, and 7.](image)

Fig. 6/26: Example – addressing the AS-i slaves with FST 200
Configuration of the AS-i bus system with FST 200

Once you have opened or selected a project in FST 200, you can call up the AS-i bus configurator and configure an AS-i bus system and/or first design one, without AS-i hardware. To do this, proceed as follows:

- Select the "Configuration" function in the "Utilities" menu.
- Select the function "Field bus/AS-i" with F6 in the "Configuration" menu.
- Select the function "AS-i bus" with F2 in the "Bus configuration" menu.

**PLEASE NOTE**

You can always call up a Help screen in any menu by pressing F9. The Help function provides you with further information about the AS-i bus configuration. The Help screen can also be opened by clicking the mouse on any point of the message line.

The functions of the AS-i bus configurator are described in the FST 200 manual. The following steps summarise the main points and assume knowledge of the FST 200 manual.
Menu "Planning AS-i slaves" (configuration)

When this menu is called, the following mask appears:

![Image of the mask](image)

Fig. 6/27: FST 200 – menu "Project planning AS-i slaves"

This menu enables you to configure the AS-i slaves (i.e. to define AS-i slave address, ID code and I/O code). You can carry out the planning before the AS-i bus system has been installed. The finalised AS-i bus system (SETPOINT list) is subsequently loaded into the SF 3 control block for commissioning.
PLEASE NOTE
All information about the ID and I/O codes of your slave can be found on the rating plate of your slaves or in the relevant slave manual.

Once you have completed the actual configuration procedure, you can conduct a SETPOINT-ACTUAL comparison for the AS-i bus in order to identify installation or configuration faults.

CAUTION
During a SETPOINT-ACTUAL comparison, the AS-i bus system is reconfigured.
The control unit goes into the stop status.
All AS-i outputs are switched off.
Function for "SETPOINT-ACTUAL comparison"

**CAUTION**

During a SETPOINT-ACTUAL comparison, the AS-i bus system is reconfigured.
The control unit goes into the stop status.
All AS-i outputs are switched off.

For the "SETPOINT-ACTUAL comparison" function, an AS-i bus system must be installed and the PC must be connected to the diagnosis interface of the SF 3 control block.

After this function has been called, the following mask appears:

![Mask Image](image)

**Fig. 6/28: Project planning AS-i slaves – Function SETPOINT-ACTUAL comparison**
Comments on the SETPOINT-ACTUAL comparison:

- Reconfiguration of the AS-i bus system takes place. All AS-i outputs are switched off for this.

- Comparison of configured list (SETPOINT list on PC) with the list of slaves identified by the AS-i master.

- Every identified variance from the SETPOINT list on the PC is displayed. Using the F4 function key, you can transfer every deviation (ACTUAL) to the configuration list to form a new SETPOINT specification.

Recommendation:
The SETPOINT-ACTUAL comparison identifies incorrect or missing slaves. If more slaves are installed on the AS-i bus system than were configured, this does not give rise to an error message.

- If the configured list (SETPOINT list on PC) contains no entries, the list of identified slaves is automatically transferred to the configured list.

PLEASE NOTE
After the SETPOINT-ACTUAL comparison, ensure with "Load config." that the SETPOINT list for the SF 3 matches the SETPOINT list on the PC.
Load configuration (function key F5)
- The displayed list of configured slaves is loaded into RAM on the SF 3. This then represents the SETPOINT status for the AS-i bus.
- A final SETPOINT-ACTUAL comparison is then performed automatically.

**PLEASE NOTE**
To conclude the entire commissioning run, all data in the control unit are read out of RAM and the EEPROM is programmed. When the EEPROM is programmed, data from the AS-i bus slaves also enter the EEPROM of the SF 3 control block.
(refer to Chapter 3 "Programming EEPROM" and FST 200 manual, Chapter 7 "Dialogue with the control unit").

F7 - With this F7 function key you enter the menu for addressing AS-i slaves with the PC. This procedure is described in detail in the next section.
Menu "Assign/modify AS-i slave address"

With the F7 function key "Prog. address", you call up the menu for changing the addresses of preinstalled AS-i slaves. For this, the AS-i bus is polled continuously and all identified AS-i slaves are displayed. In this menu, you can:

- Address every individual slave connected to the AS-i bus from the PC (without using an AS-i addressing device). Assign a new AS-i slave address to an AS-i slave identified as having address 0 before any other slaves can be readdressed.
- Assign a new address to AS-i slaves which already have an address assigned to them (from the PC).
- Spotting the cross-reference to a configured list at a glance.

All slaves configured in the SETPOINT list have a "P" beside their address. If the slaves on the AS-i bus differ, the display shows one of the following options:
- a blank line, i.e. no slave was found.
- a question mark beside the ID and/or I/O code, i.e. the slave located deviates at this point.
After this menu is called, the following mask appears:

![Fig. 6/29: FST 200 – Assign/modify AS-i slave address](image)

Fig. 6/29: FST 200 – Assign/modify AS-i slave address
With the F5 key "SETPOINT ID/IO", you can display the data of the configured slave for inspection purposes.

You must always correct any deviation. Proceed in one of these optional manners:

- Remove the incorrectly installed slave.
- Replace it with the correct configured type of slave.

or

- Transfer the installed slave to the configured SETPOINT list. For this, call up the AS-i configuration menu ("Planning AS-i slaves") and perform another SETPOINT-ACTUAL comparison at this location. Then press F4 to transfer the deviating slave.

The following example illustrates how to assign a new address to a slave retrospectively.
Example (assigning a new slave address):
- ACTUAL address (factory setting): 0
- SETPOINT address: 23

General procedure:
1. Assign a new slave address.
2. Enter the new slave in the configuration list.

Proceed as follows:
1. Assign a new slave address:
   - Select the slave address 0 which is to be altered.
   - Select it using the F2 key "Select" (the selected line is highlighted with a different screen colour).
   - Now select the desired (target) address 23.
   - Then press the F3 key "Modify address" to transfer slave 0 to address 23.

2. Enter the new slave in the planning list:
The configuration of the readdressed slave must now be entered in the SETPOINT list as well.
   - Call up the "Planning AS-i slaves" menu.
   - Enter the new slave, as appropriate.

This completes the readdressing of an AS-i slave.
Menu "SF 3 Online Mode"

In this menu, you can also call up the familiar functions:

- Display all slaves installed on the AS-i bus system (ACTUAL list). To display the AS-i I/Os, you may need to page forwards with F1 or F2.
- Check AS-i inputs and set the AS-i outputs.
- Transmit slave parameters to the appropriate AS-i slaves (via CFM as macro or in terminal mode)

When online mode is accessed from the main menu of FST 200, the following mask appears:

![SF Online Mode](Fig. 6/30: FST 200 – SF 3 Online Mode with AS-i I/Os)

Please note that the ACTUAL installation is displayed when this menu is called.
Page through several I/Os, where applicable with F1/F2, until you reach the display of AS-i I/Os. The AS-i outputs are then set/reset like local outputs.

Recommendation:
- I/Os which are not available have a grey background.
- Whenever an AS-i slave fails during operation, it is designated with an E after the address.

**WARNING**
- During online operation, only set outputs if you know their effect/reaction.
- When the system is switched on, the outputs respond immediately to your screen entry.
- Ensure that there is no risk of injury or damage when the outputs are switched on and off.

At this point, you will be familiar with all the steps required for easy commissioning.
Transmitting AS-i slave parameters

From within a program, you will find it very convenient to use function modules CFM 31 and 32 to read or write the parameters into suitable AS-i slaves.

During the commissioning process, you have two options for influencing slave parameters:

- Calling CFM 31/32 using a macro
- Calling CFM 31/32 in terminal mode.

**CAUTION**

- Only change parameters if you are familiar with the effect they have.
- When the system is switched on, the relevant slaves respond immediately to your screen entry.
- Ensure that there is no risk of injury or damage when altering slave parameters.

Appendix B of this manual describes the syntax of function modules CFM 31/32. The handling of macros or terminal mode is described in the FST 200 manual.

**PLEASE NOTE**

Consult the manual for your AS-i slave to find details about scope for setting parameters of AS-i slaves and the effects of this operation.
6.4.4 Programming the startup/performance

Characteristics of valve terminal with AS-i master after Power ON

The programmable valve terminal SF 3 with AS-i master must be switched on at the same time, or after the AS-i slaves. This is the only way of correctly recording the ACTUAL configuration and of storing it in RAM. When recording the ACTUAL configuration, a comparison is made with the SETPOINT list and the result is stored in the AS-i status bits I16.0...I16.3. Here is a key to the entries:

<table>
<thead>
<tr>
<th>Input I16.1</th>
<th>Input I16.0</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>Collective error:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Individual error on one AS-i slave</td>
</tr>
<tr>
<td>1</td>
<td>X</td>
<td>Power failure</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AS-i line</td>
</tr>
</tbody>
</table>

X bit contents not relevant

<table>
<thead>
<tr>
<th>Input I16.3</th>
<th>Input I16.2</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>SETPOINT = ACTUAL (registered slaves)</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>SETPOINT &lt; &gt; ACTUAL</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>no SETPOINT list available</td>
</tr>
</tbody>
</table>

The startup characteristics are presented in the following diagram.
Switch on all AS-i slaves

Switch on SF 3

Record ACTUAL configuration

SETPOINT = ACTUAL

SETPOINT list present?

I16.2 = 0
I16.3 = 0

I16.2 = 0
I16.3 = 1

I16.2 = 1
I16.3 = 0

I16.0 or I16.1 set?

Error response "hard"?

SF 3 goes into stop status
Reset all AS-i I/Os
ERROR-LED ON
Error message 13 with I16.1 = 1
Error message 12 with I16.0 = 1

For "Automode ON": Start of program processing

Re-entry point for "Reconfiguration of AS-i bus" by CFM 38

Fig. 6/31: Start-up characteristics of AS-i bus after Power ON
PLEASE NOTE
– The SETPOINT configuration must be created with the AS-i bus configurator of the FST 200 before commissioning.
– Evaluation of the SETPOINT-ACTUAL comparison must be performed in the program. This is the only way for appropriate error messages or other measures to be activated.

Recommendation:
For this reason, always work with a SETPOINT list.
Characteristics of valve terminal with AS-i master during operation

Characteristics in the absence of operating voltage

If there is no operating voltage on an AS-i line (Power Failure), this is reported by set input bit I16.1. In this case, the status of the input bit 16.0 (collective message for individual errors of AS-i slaves) is not relevant. With function module CFM 37, the error characteristics during Power Failure of the AS-i line are defined as follows:

Hard characteristics:  
- Stop all programs  
- Reset the local and the AS-i I/Os  
- Error LED on, error message 13

Soft characteristics:  
- Handling the errors in application program.

PLEASE NOTE
The fault characteristics set with CFM 37 are stored in remanent manner, i.e. they are not affected by mains power failure.
What to do when a slave error occurs

Errors can affect the AS-i slaves during operation. Typical station errors are:

- AS-i operating voltage of slave below tolerance.
- Short circuit on an AS-i output.
- Hardware error (slave failed).

If a fault occurs in an AS-i slave, the collective error bit I16.0 is set.

The faulty slave can be localised with:

- CFM 35 (localisation of individual errors; refer to Appendix B). This is only possible with "soft" error treatment (factory setting, can be set with CFM 37).
- FST 200 in online operation. All failed AS-i slaves are identified with an "E" after the AS-i address.
In the event of a power failure or a fault in the AS-i slaves, the following response is possible:

**Flowchart:**
- **I16.1 set?**
  - yes
  - no
- **I16.0 set?**
  - yes
  - no
- **Error response “hard”**
  - yes
  - no (default)
  - SF 3 in stop status
  - Reset all local and AS-i I/Os
  - ERROR-LED ON, error message 12
  - with CFM 35
  - Localise individual errors in AS-i slaves
  - SF 3 in stop status
  - Reset all local and AS-i I/Os
  - ERROR-LED ON, error message 13
- **Standard program processing**
- **Respond to Power Failure by user program**

**Fig. 6/32: Response to Power Failure or slave error on AS-i bus**
6.5 DIAGNOSIS AND
ERROR HANDLING
6.5 Diagnosis

Contents

6.5 DIAGNOSIS AND ERROR HANDLING ................................... 6-83

6.5.1 Summary of diagnosis options .... 6-85

6.5.2 Diagnosis on location ............... 6-86
  LED displays on AS-i master .......... 6-86
  LED displays on the AS-i slaves .... 6-87
  FST 200 .................................. 6-88

6.5.3 AS-i-specific error handling .... 6-91
  Localisation of defective slaves ... 6-91
  Addressing by means of the
  AS-i addressing device ............... 6-92
  Addressing with the FST 200 ......... 6-93
  Addressing by means of
  autoprogramming ....................... 6-94

6.5.4 Error handling with CFM 35 ........ 6-96
6.5.1 Summary of diagnosis options

For diagnosis on the AS-i bus and to localise defective slaves, the system offers you the following options:

- LED displays on location
- Diagnosis in online operation of FST 200
- Diagnosis via AS-i status bits
- Diagnosis via function module 35 (CFM 35)

These options are described in more detail in the following section.
6.5.2 Diagnosis on location

LED displays on AS-i master

<table>
<thead>
<tr>
<th>LED</th>
<th>Meaning</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONF (yellow LED)</td>
<td>LED is always dark/without significance in conjunction with SF 3</td>
<td>None</td>
</tr>
<tr>
<td>BUS ¹ (green LED)</td>
<td>Lit: AS-i voltage connected, AS-i power pack correctly connected to AS-i bus (yellow cable)¹.</td>
<td>None</td>
</tr>
<tr>
<td></td>
<td>Dark: No AS-i voltage to AS-i master and/or AS-i bus¹.</td>
<td>Connect AS-i power pack to AS-i master and switch on.</td>
</tr>
<tr>
<td></td>
<td>If LED is still dark: Hardware error in AS-i master</td>
<td>Service required</td>
</tr>
</tbody>
</table>

¹ This LED does not monitor the presence of a 24 V auxiliary power supply

Fig. 6/33: LEDs on AS-i master
LED displays on the AS-i slaves

All AS-i slaves from Festo have a green BUS LED and, depending on the number of inputs/outputs, also have the following additional LEDs (status indicators) in the colours:
- Green (status indicator for digit. inputs)
- Yellow (status indicator for digit. outputs)

The yellow and green LEDs display the prevailing signal on the input and output.

<table>
<thead>
<tr>
<th>LED</th>
<th>Meaning</th>
<th>Error handling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input/output</td>
<td><img src="#" alt="Table" /></td>
<td></td>
</tr>
<tr>
<td>yellow and/or green</td>
<td><img src="#" alt="Table" /></td>
<td></td>
</tr>
<tr>
<td>Lit</td>
<td>Logic 1 (signal present)</td>
<td>If output/actuator does not switch:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- check the 24 V auxiliary power supply</td>
</tr>
<tr>
<td></td>
<td></td>
<td>otherwise: Replace defective slave</td>
</tr>
<tr>
<td>Dark</td>
<td>Logic 0 (signal not present)</td>
<td>If output was set:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- check the addressing</td>
</tr>
<tr>
<td></td>
<td></td>
<td>otherwise: Replace defective slave</td>
</tr>
<tr>
<td>BUS ¹ (green LED)</td>
<td><img src="#" alt="Table" /></td>
<td></td>
</tr>
<tr>
<td>Lit</td>
<td>AS-i power connected. AS-i slave correctly connected to power pack (yellow cable) ¹.</td>
<td>None</td>
</tr>
<tr>
<td>Dark</td>
<td>No AS-i power to slave and/or AS-i bus ¹.</td>
<td>Connect AS-i power pack to slave and switch on.</td>
</tr>
<tr>
<td></td>
<td>If LED is still dark: Hardware error in slave.</td>
<td>Replace defective slave</td>
</tr>
</tbody>
</table>

¹ This LED does not monitor the presence of a 24 V auxiliary power supply.

Fig. 6/34: LEDs on the Festo AS-i slaves
FST 200

The FST 200 offers a few auxiliary functions which make rapid and comfortable diagnosis of all connected AS-i slaves possible. Proceed as follows:

1. Depending on the safety concept of your unit/machine:
   Set SF 3 control block to STOP

   Recommendation:
   If the function "SETPOINT-ACTUAL comparison" is called, the control unit is stopped automatically and all outputs are switched off.

2. Connect PC to the diagnosis interface of the SF 3 control block

3. Start FST 200

4. For diagnosis, select either the menu for
   - SF 3 online operation or
   - Project planning AS-i slaves in the AS-i bus configurator.
The following diagnosis aids are provided in the menu "SF 3 Online Mode":

<table>
<thead>
<tr>
<th>Diagnosis aids for &quot;SF 3 Online Mode&quot;</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Displays the AS-i I/Os</td>
<td>For localisation of defective inputs/outputs of an AS-i slave (enter an &quot;E&quot; after the AS-i address).</td>
</tr>
<tr>
<td>Set/delete individual AS-i outputs (toggle)</td>
<td>Key F1...F3</td>
</tr>
<tr>
<td>Transmit slave parameters</td>
<td>Reading/writing slave parameters with CFM 31/32 as macro or in terminal mode (only for slaves with variable parameters).</td>
</tr>
</tbody>
</table>

Fig. 6/35: Diagnosis aids in menu "AS-i Online"

These functions are described in detail in Chapter 6.4.3 of this description and in Chapter 7.4 of the FST description.
The menu "Project planning AS-i slaves" provides you with the following diagnosis aids:

<table>
<thead>
<tr>
<th>Diagnosis aids for &quot;Configuration&quot;</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Display of all configured slaves</td>
<td>For localisation of configuration errors (IO-/ID code incorrect)</td>
</tr>
<tr>
<td>SETPOINT-ACTUAL comparison</td>
<td>For localisation of installation errors or failed slaves</td>
</tr>
<tr>
<td></td>
<td>Note: SETPOINT (in EEPROM) &lt;&gt; ACTUAL bit is set, if a slave is removed/changed during operation, or if it is no longer recognised. Bit is not set, if new slaves are added during operation.</td>
</tr>
<tr>
<td>Processing/resetting</td>
<td>Correcting identified errors in the SETPOINT list</td>
</tr>
</tbody>
</table>

Fig. 6/36: Diagnosis aids in menu "Project planning AS-i slaves"

These functions are described in detail in Chapter 6.4.3 of this manual and in Chapter 8.2 of the FST description.
6.5.3 AS-i-specific error handling

Localisation of defective slaves
To localise defective slaves, you are provided with the following options:
- LED displays on location.
- Diagnosis in online operation of FST 200.
- Diagnosis via AS-i status bits and CFM 35.

Once you have localised a defective slave and need to replace it, the new slave is usually assigned the address 0 (factory setting). To address the new slave, you have the following possibilities:
- Re-addressing by means of the AS-i addressing device
- Re-addressing with the FST 200
- Auto-programming using slave address 00
Addressing by means of the AS-i addressing device

You can always use this method if one or more slaves on the AS-i bus are defective.

WARNING
When replacing defective slaves, please take due account of the following:
• Before conversion, switch off all operating voltages on the AS-i bus system.
• Use only AS-i slaves with identical I/O and ID codes as a substitute for a defective slave.
This is the only way of ensuring that your AS-i bus system functions safely and correctly.

Proceed as follows:
1. Switch off the operating voltages on AS-i bus system.
2. Remove the defective slave from AS-i bus and note its slave address.
3. Set the address of the defective slave on the new slave using the AS-i addressing device.
4. Perform complete installation of the new slave on the AS-i bus.
5. Switch the operating voltages for AS-i bus system back on.
Addressing with the FST 200

You can use this method step by step if one or more slaves on the AS-i bus are defective.

**WARNING**

When replacing defective slaves, always note the following:

- This work must be performed with the operating voltage (terminal + AS-i bus) switched on. Ensure that there is no risk of injury or damage to human beings or equipment when readdressing the new slave.
- When replacing a defective slave, only use AS-i slaves with identical ID and I/O codes.

This is the only way of ensuring that your AS-i bus system functions safely and correctly.

Proceed as follows:

1. Remove the defective slave from the AS-i bus and note down its slave address.
2. Perform complete installation of new the slave on the AS-i bus.
3. Start the FST 200 and select the menu for "Assign/modify AS-i slave address" in the AS-i bus configurator.
   - The new slave appears with the AS-i address 0
- P appears at the location of the defective slave (if a configured SETPOINT list is available).
  Proceed as follows:
  • Highlight slave 0
  • Select with F2
  • Highlight address of defective slave
  • Transfer new slave with F3.

4. Only at this point should you install another new slave and repeat the steps.

Addressing by means of autoprogramming

Autoprogramming constitutes a special feature of the AS-i bus system. This involves replacing a defective slave with another slave with identical ID and I/O code (slave address 00). When the operating voltage is switched back on, the AS-i master recognises the new slave and automatically assigns it the slave address of the defective slave.

You can only use this method if no more than one slave on the AS-i bus is defective.

WARNING

When replacing defective slaves, please note the following:
• Before changeover, switch off all operating voltages on the AS-i bus system.
• When replacing a defective slave, only use AS-i slaves with identical ID and I/O codes. This is the only way of ensuring that your AS-i bus system will operate safely and correctly. During autoprogramming, any slave with a different ID or I/O code is ignored; i.e. the I/Os of the (old) slave address are not processed when the unit is switched back on.
Proceed as follows for autoprogramming:

1. Switch off the operating voltages on the AS-i bus system.

2. Perform complete installation of the new slave with slave address 00 on the AS-i bus.

3. Switch the operating voltages of the AS-i bus system back on. The AS-i master now recognises the new slave by its address 00 and assigns it the address of the old slave - if ID and I/O codes are identical.

**WARNING**

– If you have not installed an identical slave, the new slave is ignored, i.e. the I/Os for the (old) slave address are not processed.
– The AS-i bus system starts up despite this.
– However, status bit I16.2 (SETPOINT <> ACTUAL) is set to enable the program to respond to this error.

Recommendation:
The (incorrect) slave is not recognised or displayed until you use the FST 200 to call up the menu "Assign/modify AS-i slave address".
6.5.4 Error handling with CFM 35

Function modules form part of the operating system. To diagnose the AS-i slaves, the following function module is implemented in the SF 3:

- CFM 35: Diagnosis of all AS-i slaves

CFM 35 is described in more detail in the following section. All other AS-i function modules are described in Appendix B.
When using this module, you receive diagnosis information from all configured AS-i slaves in your AS-i bus system.

The diagnosis result is prepared in special functional units FU33 to FU34. With appropriate programming techniques, you can either receive collective information or information about individual AS-i slaves.

**Input format**

THEN CFM 35

**Parameters**

none  

or P1 = any value (optional)

**Return code parameters:**

**Case 1:**

P1 (FU32) = {-1} processing successful  
P2 (FU33) = error in AS-i slave 1...15  
P3 (FU34) = error in AS-i slave 16...31  
P4 (FU35) = (optional, only when accessed with transfer parameter)  
Address of lower value faulty AS-i slave (decimal)

**Case 2:**

P1 (FU32) = {0} processing faulty  
P2 (FU33) = (error number)  
possible error number 14, 100
The diagnosis result described in the following section only occurs if function module CFM 35 signals successful processing (FU32 = -1).

Diagnosis result in FU33 ... FU34 (FU32 = -1)

**FU33:**

<table>
<thead>
<tr>
<th>AS-i slave address</th>
<th>Data bit number</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
<td>D14</td>
</tr>
<tr>
<td>15</td>
<td>14</td>
</tr>
</tbody>
</table>

**FU34:**

<table>
<thead>
<tr>
<th>Log. value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Not a slave error</td>
</tr>
<tr>
<td>1</td>
<td>AS-i slave failed</td>
</tr>
<tr>
<td>x</td>
<td>Bit content not relevant</td>
</tr>
</tbody>
</table>

Data bits from AS-i slaves which are not available supply the logic value "0". Only configured slaves can cause an error.

If **any** desired parameter is transferred when function module 35 is accessed, the address of the first faulty AS-i slave will be shown as a decimal value in return parameter P4. This will be zero if the registered AS-i slave does not have any functional faults.

The term "first faulty AS-i slave" means the lowest value slave which does **at present** have any functional fault.
6.6 TECHNICAL DATA
Contents

6.6 TECHNICAL DATA ............................. 6-99
  AS-i master ................................. 6-101
  AS-i mains power assembly .......... 6-102
  Electromagnetic compatibility .... 6-102
  General technical data ............. 6-103
6.6 TECHNICAL DATA

<table>
<thead>
<tr>
<th>AS-i master</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of protection (acc. to DIN 40050)</td>
<td>IP65 (compl. fitted)</td>
</tr>
<tr>
<td>Ambient temperature</td>
<td>-5 °C...+50 °C</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>-20 °C...+60 °C</td>
</tr>
</tbody>
</table>

**Electromagnetic compatibility**
- Interference emitted
- Immunity against interference
  - Tested as per DIN EN 61000-6-4 (industry)\(^1\)
  - Tested as per DIN EN 61000-6-2 (industry)

**Connection for AS-i bus**
- (AS-i +, brown)
  - Connection only via specified AS-i power pack
  - max. current consumption (AS-i –, light blue)

**AS-i bus**
- e.g. Festo mains power pack, parts no. 18 949
  - max. 65 mA DC 0 V

**Connection for terminal**
- (internal power supply for terminal; Pin 1 operating voltage connection)
  - additional internal current consumption of AS-i master module at 24 V
  - Comment: The AS-i slaves draw their voltage/ power supply from the external AS-i power pack

**Diagnosis interface**
- Design
- Type of transmission
- Synchronous procedure
  - RS 232, floating serial, asynchronous, Full-duplex software handshake
    - (1 start bit, 8 data bits, 1 stop bit)
    - 9600 Baud

\(^1\) The component is intended for industrial use.

Please refer to Chapter 2 in this description for all other technical data.
### AS-i mains power assembly

<table>
<thead>
<tr>
<th>Output</th>
<th>Both outputs are resistant to overload, sustained short-circuit and idling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voltage $V_{out1}$</td>
<td>30.55 V fixed setting</td>
</tr>
<tr>
<td>Voltage $V_{out2}$</td>
<td>24.0 V fixed setting</td>
</tr>
<tr>
<td>Total tolerance</td>
<td>$-3% + 15%$ contains factory check, load</td>
</tr>
<tr>
<td>Base load</td>
<td>not necessary</td>
</tr>
<tr>
<td>Power output $P_{out}$</td>
<td>max. 240 W Assembly without lateral offset</td>
</tr>
<tr>
<td>Residual ripple</td>
<td>max. 50 mVpp 0...20 MHz constant current or R load</td>
</tr>
<tr>
<td>Safety</td>
<td>PELV IEC/DIN EN 60204-1)</td>
</tr>
</tbody>
</table>

### Input

| Nominal voltage 1 | 100...127 VAC Switch position 115 V |
| Range | 88...132 VAC Full data contents |
| Nominal voltage 2 | 220...240 VAC Switch position 230 V |
| Range | 187...264 VAC Full data contents |
| Nominal frequency | 47...63 Hz DC and/or 400 Hz |
| Input current | max. 6 $A_{eff}$/2.8 $A_{eff}$ at 115/230 VAC |
| Mains power failure bridging time | min. 20 ms at 187 V/100 % load |
| Startup peak current | max. 25 A for cold starts (+ 25 °C) |

### Electromagnetic compatibility

- Interference emitted Tested as per DIN EN 61000-6-4 (industry)1)
- Immunity against interference Tested as per DIN EN 61000-6-2 (industry)1)

Immunity to interference acc. to NAMUR
- NAMUR specifications are complied with

1) The component is intended for industrial use.
### General technical data

#### Safety

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insulation resistance</td>
<td>min. 5 MΩ</td>
</tr>
<tr>
<td>Earth cable resistor</td>
<td>&lt; 0.1 Ω</td>
</tr>
<tr>
<td>Leakage current</td>
<td>max. 0.75 mA</td>
</tr>
<tr>
<td>Overvoltage cat.</td>
<td></td>
</tr>
</tbody>
</table>

#### Operating and environmental data

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating temperature</td>
<td>max. -10 ° C...+70 ° C</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>typ. -20 ° C...+70 ° C</td>
</tr>
<tr>
<td>Air humidity</td>
<td>max. 95 %</td>
</tr>
<tr>
<td>Fitting position</td>
<td>vertical</td>
</tr>
<tr>
<td>Ventilation</td>
<td>natural convection</td>
</tr>
<tr>
<td>Vibration</td>
<td>0.075 mm</td>
</tr>
<tr>
<td>Shock</td>
<td>11 ms/15 g</td>
</tr>
</tbody>
</table>

Tu (at 1 cm distance)  
Tu  
Without dewing  
Secure air throughput  
IEC 68-2-6(10...60 Hz)  
IEC 68-2-27 (3x)
Programmable valve terminal
with
control unit SF 3

Chapter 7: Description of CP interface

CP interface only
in conjunction with
type 03...05
Contents

7.1 USER INSTRUCTIONS AND SYSTEM SUMMARY .............. 7-1
   Information on this description ............ 7-3
   System summary .................. 7-5
   Description of components ............ 7-6

7.2 FITTING AND INSTALLATION ............. 7-9
   General instructions ................ 7-9

7.3 COMMISSIONING .................. 7-13
   Preparing the CP system .............. 7-13
   FST 200 in on-line mode .............. 7-15
   Switch-on reaction of the CP system with SF 3 ............ 7-17
   Operating reaction of the CP system with SF 3 ............. 7-19
   Address range of the CP system .......... 7-21
   Basic rules for addressing the CP system ............ 7-23
   Address assignment after extension or conversion ........ 7-25

7.4 DIAGNOSIS AND ERROR TREATMENT .......... 7-29
   LED display on the CP interface .......... 7-29
   Reaction of the CP system to faults .... 7-30
   Diagnosis of the CP system with FST 200 .......... 7-32
   Function modules of the CP system .... 7-33
   The meaning of the diagnostic bytes of strings 0...3 .......... 7-35

PLEASE NOTE
This description supplements the documentation on your valve terminal with the necessary information on the CP interface.
7.1 USER INSTRUCTIONS
AND
SYSTEM SUMMARY
Contents

7.1 USER INSTRUCTIONS AND SYSTEM SUMMARY

Information on this description ........ 7-3
System summary .......................... 7-5
Description of components ............. 7-6
### 7.1 USER INSTRUCTIONS AND SYSTEM SUMMARY

#### Information on this description

The following product-specific terms and abbreviations are used in this description.

<table>
<thead>
<tr>
<th>Term/abbreviation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>CP system</td>
<td>Complete system consisting of node, CP interface and CP modules.</td>
</tr>
<tr>
<td>CP modules</td>
<td>Common term for various modules which can be incorporated in a CP system.</td>
</tr>
<tr>
<td>CP connection</td>
<td>Socket or plug for connecting the CP modules with the CP cable.</td>
</tr>
<tr>
<td>CP cable</td>
<td>Special cable for coupling the various CP modules.</td>
</tr>
<tr>
<td>SAVE (button)</td>
<td>Saves the current string assignment (connected I/Os); when the CP system is restarted, the last save string will be compared with the current string assignment. Deviations are indicated by a flashing LED.</td>
</tr>
<tr>
<td>String</td>
<td>Total number of I/O modules connected to one CP connection of the CP interface.</td>
</tr>
<tr>
<td>String assignment</td>
<td>Total number of all I/O modules connected via strings to CP interface.</td>
</tr>
</tbody>
</table>

Information on further CP interfaces can be found in the description for the relevant module.
### Descriptions for the CP system

<table>
<thead>
<tr>
<th>Description</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;CP system, installation and commissioning&quot;</td>
<td>General basic information on operating, fitting, installing and commissioning CP systems</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Description</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Programmable valve terminal type 03 with SF 3&quot;</td>
<td>Information on fitting, installing and commissioning CP valve terminals</td>
</tr>
<tr>
<td>&quot;CP valve terminal, pneumatics&quot;</td>
<td>Information on fitting, installing and commissioning CP-I/O modules</td>
</tr>
<tr>
<td>&quot;CP modules, electronics&quot;</td>
<td>Information on fitting, installing and commissioning CP-I/O modules</td>
</tr>
</tbody>
</table>

### Fig. 7/1: Descriptions for the CP system
System summary

CP systems consist of the following modules:

<table>
<thead>
<tr>
<th>CP modules</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node with CP interface</td>
<td>There is a CP interface for the various fieldbuses.</td>
</tr>
<tr>
<td></td>
<td>- provides the connection to certain nodes (here: SF 3)</td>
</tr>
<tr>
<td></td>
<td>- offers connections for up to 4 strings to which CP modules and CP valve terminals can be connected</td>
</tr>
<tr>
<td></td>
<td>- transmits control signals to the connected modules and monitors their functioning</td>
</tr>
<tr>
<td>CP valve terminals</td>
<td>- provide various valve functions via valve plates for controlling pneumatic actuators</td>
</tr>
<tr>
<td></td>
<td>- relay plates, pressure isolating plates and blanking plates can be used here as well</td>
</tr>
<tr>
<td>Input modules</td>
<td>- There are various special designs for various types of connections; these enable e.g. cylinder positions to be scanned</td>
</tr>
<tr>
<td>Output modules</td>
<td>- provide universally usable electrical outputs for actuating low current consuming devices (further valves, bulbs, etc.)</td>
</tr>
</tbody>
</table>

Fig. 7/2: System summary of the CP modules
Description of components

The following diagrams show the general operating, display and connecting elements of the CP interface.

**PLEASE NOTE**
Special information on the structure of the node you are using can be found in Chapter 2.

---

**Fig. 7/3: General operating, display and connecting elements**

1. SAVE button
2. String error LEDs
3. Node SF 3, see Chapter 2
4. Label fields
5. CP connections for up to 4 strings (0...3)
7.2 FITTING AND INSTALLATION
Contents

7.2 FITTING AND INSTALLATION

General instructions .................. 7-9
7.2 FITTING AND INSTALLATION

General instructions

WARNING
Before carrying out installation and maintenance work, switch off the following:

- the compressed air supply
- the operating voltage supply to the node (pins 1 and 2).
- the operating voltage supply to the CP output modules

You thereby avoid:

- uncontrolled movements of loose tubing.
- unintentional movements of the connected actuators
- undefined switching states of the electronic components

PLEASE NOTE
- The CP interface must always be fitted directly to the left of the node.
- Only one CP interface may be fitted per node.
- All further information on fitting and installation of a CP system can be found in the manual for the CP system “Installation and commissioning”.

7.3 COMMISSIONING
7.3 COMMISSIONING

Preparing the CP system ............ 7-13
FST 200 in on-line mode............. 7-15
Switch-on reaction of the
CP system with SF 3................. 7-17
Operating reaction of the
CP system with SF 3................. 7-19
Address range of the CP system ... 7-21
Basic rules for addressing the
CP system.......................... 7-23
Address assignment after
extension or conversion............. 7-25
7.3 COMMISSIONING

Preparing the CP system

PLEASE NOTE
Prepare the CP system for commissioning.
(see "CP system" manual)

Before commissioning the CP system, proceed as follows:

1. Connect the operating voltage for the node.
2. Connect the CP modules.
3. Switch on the operating voltage.
4. Save the string assignment. To do this you have the following 3 possibilities:
4a. by pressing the SAVE button on the CP interface.

4b. in on-line mode with FST 200. Save the string assignment in on-line mode as follows:

- Start the on-line mode of the FST 200
- Change to static display of the CP I/Os (F1 "SF3-Info", then F1 "On/off")
- Press button F7 "Register CP new"

4c. Set soft error reaction by program. Access CFM 28 "New configuration of CP system".
FST 200 in on-line mode

When you activate functions F1 "SF3-Info" and F1 "On/off", the current values of the connected local digital inputs and outputs will be shown on the screen. CP inputs/outputs which are unused are shown in grey.

When a CP interface is installed, the operands (I/O 8.x...15.x) will be firmly assigned to the CP system. With the buttons F1 and F2 you can display the CP inputs and outputs.

![Diagram](image)

Fig. 7/4: Display of inputs and outputs in on-line mode
Remarks:

- When CP inputs and outputs (Fig. 7/4) are displayed, the SAVE button on the CP interface has no function.
- With button F7 you can register and save the string assignment of the CP system. (This function is identical to the function of the SAVE button on the CP interface).

If a CP output is already set, the following question will appear:

*Caution!*
*Outputs already set will be reset briefly.*
*New outputs can be set.*

Do you wish to continue (Yes/No)?

You can avoid this question if you first:
- stop all programs
- reset all CP outputs.
Switch-on reaction of the CP system with SF 3

When the CP system is switched on, there is a starting phase during which the string assignment is determined. If there is no difference between this and the saved string assignment, the system will start operating immediately. The run-up bit input I0.1.0 defined for this will be reset.

If there is a difference (e.g. during the first commissioning), the relevant string LEDs on the CP interface and the operating LED on the CP modules will flash. The run-up bit input I0.1.0 for the CP system remains set until:

- the string assignment is corrected (manual error elimination) or
- the SAVE button of the CP interface is pressed or
- button F7 "New configuration CP system" is pressed in FST on-line for CP or
- you carry out a new configuration of the CP system by user program by accessing CFM 28 (only possible with soft set error reaction, see CFM 27; default: hard error reaction)

CAUTION
When the CP system is reconfigured, the following must be observed:
- outputs already set will be reset briefly
- new outputs can be set
An appropriate warning will be shown in FST on-line.
Switch on CP system ready to operate

Register current string assignment

Current string assignment = saved string assignment?

Yes

CP system ready to operate

SF 3: I 0.1.0 = 0
Programs are processed further

No

LED on appropriate string flashes

SF 3: E 0.1.0 = 1

SAVE button or FST on-line button "F7" pressed or CFM 28 accessed?

Yes

Current string assignment is saved

Manual error elimination possible

No

Fig. 7/5: Switch-on reaction of the CP system

1 When commissioning is done the first time the string assignment must be saved here.
Operating reaction of the CP system with SF 3

**PLEASE NOTE**
Observe here also the information in the manual for your CP system - Installation and commissioning, section "Replacing modules during operation".

With CFM 27 you can determine the operating/error reaction when one or more CP modules fail.

- **Hard reaction:**
  Stop all programs. ERROR-LED on, run time error message 21.

- **Soft reaction:**
  Treat errors in user program.

When soft error reaction is selected, one or more modules can be temporarily disconnected from the CP system during operation (failure of a module) and/or replaced by one or more modules of the same type. In this case:

- the relevant string LED will flash
- the common error bit input I0.1.1 will be set
- in FST on-line for CP the message "CP connection interrupted" will appear for the relevant CP string.
- CFM 25 will indicate an error status when accessed.
The relevant module will no longer be operated, but data exchange with the other functioning modules will continue, which means that CP inputs and outputs can be processed as follows:

- in on-line mode.
- in user program, if "soft error reaction" has been set by using CFM 27.

When the module is reconnected, it will operate again and its design/type will be stored in the SF 3.

If one or more modules of different types are replaced, normal operation will only continue if:

• you press button F7 in FST on-line for CP
• you press the SAVE button on the CP interface
• you access CFM 28 (only if soft error reaction is set).

Further diagnostic messages of the modules are shown with common error bit input I0.1.2 and can be determined further with CFM 25 (see section Diagnosis).
Address range of the CP system

The available address range is determined by the hardware fitted on the modular valve terminals.

The diagram below shows the address range for the operating mode "Stand alone" with the CP interface. The structure of the valve terminal is shown schematically and the address range is listed in a table.

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>Operating mode master</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW0...7</td>
<td>local I/Os</td>
</tr>
<tr>
<td>OW0...7</td>
<td>local inputs</td>
</tr>
<tr>
<td></td>
<td>local outputs/valves</td>
</tr>
<tr>
<td>IW8...15</td>
<td>CP system (string 0...3) for:</td>
</tr>
<tr>
<td></td>
<td>CP inputs</td>
</tr>
<tr>
<td></td>
<td>CP outputs/CP valves</td>
</tr>
</tbody>
</table>

Fig. 7/6: Operands (I/O addresses) of the CP system
Contrary to the information in the general CP manual "Installation and commissioning", fixed addresses are reserved for the CP system in the address range of the SF 3.

These addresses (IW/OW8...15) are firmly assigned to the CP system and therefore reduce the address range of the local I/Os to 64I and 64O (IW/OW0...7):
- as soon as the CP interface is fitted
- irrespective of whether a CP string is assigned or not

**CAUTION**

Note when assigning addresses as well as in eventually existing programs:

- If a CP interface is fitted or at a later stage, the address range IW/OW8...15 is firmly assigned to the CP system and therefore can not be used for the local I/Os.
- If a CP interface is removed at a later stage, the address range IW/OW8...15 is again available for the local I/Os.
Basic rules for addressing the CP system

- The CP interface provides four strings with a total of 64 input addresses and 64 output addresses
- One string occupies 16 input addresses and 16 output addresses
- The address assignment in the individual strings is firmly assigned in ascending order.

<table>
<thead>
<tr>
<th>String number</th>
<th>Input addresses</th>
<th>Output addresses</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>I8.0...9.7</td>
<td>O8.0...9.7</td>
</tr>
<tr>
<td>1</td>
<td>I10.0...11.7</td>
<td>O10.0...11.7</td>
</tr>
<tr>
<td>2</td>
<td>I12.0...13.7</td>
<td>O12.0...13.7</td>
</tr>
<tr>
<td>3</td>
<td>I14.0...15.7</td>
<td>O14.0...15.7</td>
</tr>
</tbody>
</table>

The address assignment of the individual CP modules is determined by the string to which the modules are connected.
The diagram below shows as an example the address assignment of a CP system with SF 3.

Fig. 7/7: Address assignment of a CP system with SF 3
Address assignment after extension or conversion

CAUTION
Observe the following if the string assignment of your CP system is modified at a later stage.

The input and output addresses of the modules will be modified if the modules are connected to a different string.

A special feature of the CP system is its flexibility. If the demands placed on your machine change, you can quickly replace, remove or add further modules.

The input and output addresses of already used modules will not be modified, providing these remain connected to the same string.

The following diagram shows as an example the new address assignment after modification of the string assignment shown in Fig. 7/7.

Compared with Fig. 7/7, strings 0 and 2 have been extended by the addition of further modules. On string 0 a CP valve terminal with 4 valve locations has been replaced by a valve terminal with 8 valve locations. Please note that the assignment of the input and output addresses on the strings has not changed.
Fig. 7/8: Extended add. assignment of a CP system with SF 3

1. Address range occupied by CP system
2. Address range used by module
3. ( ) = Reserved address range

O = Output  I = Input
7.4 Diagnosis and error treatment

7.4 DIAGNOSIS AND ERROR TREATMENT
Contents

7.4 DIAGNOSIS AND ERROR TREATMENT

LED display on the CP interface . . . . . . . . 7-29
Reaction of the CP system to faults . . 7-30
Diagnosis of the CP system with FST 200 . . . . . . . . . 7-32
Function modules of the CP system . . 7-33
The meaning of the diagnostic bytes of strings 0...3 . . . . . . . . 7-35
7.4 DIAGNOSIS AND ERROR TREATMENT

LED display on the CP interface

The LEDs on the CP interface permit fast on-the-spot diagnosis.

Meaning of the string LEDs

- In the starting phase:
  Flashes if the string assignment has been modified since the last operation
- During operation:
  Lights up if a CP connection is interrupted.
Reaction of the CP system to faults

If there is a fault on a CP module during operation (e.g. cable fracture), the appropriate string LED will light up. The status LED on the relevant CP module will go out. All faultlessly working modules remain ready for operation.

Faults in the CP system can also be analysed with software. The following possibilities exist:

- Common message “Failure of CP components” I0.1.1 = 1 (e.g. cable fracture)
- Common message “Individual error of CP components” I0.1.2 = 1 (e.g. short circuit)

Both common errors can be localized more accurately in on-line mode with the FST 200 or via CFM 25. These two possibilities are described in detail below.

You can restore the defective connection during operation, or replace the defective module without affecting operation of the remaining modules in the other strings. When the connection has been restored or after replacement, the appropriate module will automatically be ready to operate again.
String error LED lights up; common error "Failure CP"
I0.1.1 = 1
(see also CFM 25), non-faulty modules are still active

Has error been eliminated manually?

No

Have not more than 1 I-module and 1 O-module/valve terminal been replaced? (max. 2 modules)

Yes

Failed CP modules will not be operated.

No

String error LED is switched off. Failed CP modules are operated again.

Yes

1) Only modules of the same type on a single string can be replaced at the same time during operation.

2) To restore operation:
   - you must press "F7" in FST on-line mode or
   - you must access CFM 28 or
   - the operating voltage must be switched on again and pressed the SAVE button

Fig. 7/10: Reaction to faults in the system
Diagnosis of the CP system with FST 200

The status displays of the CP inputs and outputs, as well as diagnostic information for the CP system will be displayed completely on one page in on-line mode (FST on-line for CP).

- Non-existant inputs and outputs are shown on a grey background
- Diagnostic information is shown in clear text

List of diagnostic information:
- CP connection interrupted
- Last (load) voltage failure valve terminal
- Undervoltage valves
- Short circuit sensor supply
- Common message short circuit
- Voltage failure output modules

*) (of a CP output)
Function modules of the CP system

Function modules are part of the operating system. The following function module is particularly suitable for diagnosing the CP system:

- CFM 25: Diagnosis of CP module, input and output modules

CFM 25 is described in detail below. All other CP function modules can be found in Appendix B.
The diagnostic bytes always occupy bits 0...7 of the return parameter (bits 8...15 always 0). The meaning of the diagnostic bytes is explained in the following diagram:

**Return parameter**

**Case 1:**
- \( P1 \) (FU32) = {-1} Processing successful
- \( P2 \) (FU33) = Diagnostic byte of string 0
- \( P3 \) (FU34) = Diagnostic byte of string 1
- \( P4 \) (FU35) = Diagnostic byte of string 2
- \( P5 \) (FU36) = Diagnostic byte of string 3

**Case 2:**
- \( P1 \) (FU32) = {0} Processing faulty
- \( P2 \) (FU33) = \{error number\} possible error numbers 19, 100

**Remarks:**
The diagnostic bytes always occupy bits 0...7 of the return parameter (bits 8...15 always 0). The meaning of the diagnostic bytes is explained in the following diagram:
The meaning of the diagnostic bytes of strings 0...3

<table>
<thead>
<tr>
<th>FU-no.</th>
<th>String Type</th>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>CP modules</td>
<td>IX</td>
<td>V_val</td>
<td>V_sen</td>
<td>V_out</td>
<td>SC/O</td>
<td>I_on</td>
<td>I_off</td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td>IX</td>
<td>V_val</td>
<td>V_sen</td>
<td>V_out</td>
<td>SC/O</td>
<td>I_on</td>
<td>I_off</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Remarks:

- The common message "Failure of CP component(s)" I0.1.1 is composed of bits 0, 1, 7 of the diagnostic bytes of strings 0...3.
- The common message "Individual errors of CP component(s)" I0.1.2 is composed of bits 2...5 of the diagnostic bytes of strings 0...3 (see also Chapter 3.4 "Diagnosis and error treatment").
Programmable valve terminal
with
control block SF 3

APPENDIX A:
General connection principles,
cable length and earthing
Contents

A.1 GENERAL CONNECTION PRINCIPLES . . . A-3
   Connection of cables to the plug/sockets . . . A-4

A.2 CABLE LENGTH
   AND WIRE CROSS SECTION . . . . . . . . . . . . . . . . . . A-9
   Determining with the use of graphs . . . . . A-10
   Determining with the use of formulae . . . . . A-12

A.3 EARTHING . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . A-17
   Example of connection . . . . . . . . . . . . . . . . . A-17
A.1 GENERAL CONNECTION PRINCIPLES
A.1 GENERAL CONNECTION PRINCIPLES

Connection of cables to the plug/sockets . . . A-4
A.1 GENERAL CONNECTION PRINCIPLES

**WARNING**
Switch off the following items before installation and maintenance operations:
- Compressed air supply.
- Operating voltage supply electronics (Pin 1).
- Operating voltage supply outputs/valves (Pin 2).

By doing this, you prevent:

- uncontrolled movements from loosened flexible tubes.
- accidental movements from connected actuator units.
- undefined circuit conditions of the electronics.
Connection of cables to the plug/sockets

**CAUTION**
The position of the pins on the plugs differs from that on the sockets!

- The connections on the input and output modules are designed as sockets.
- The connections on the diagnosis interface and the operating voltage connection are designed as plugs.
For pin assignment, please consult subsequent chapters.

Once you have selected appropriate cables, connect them to the plug/sockets in accordance with the subsequent steps 1...7.

1. Open the plug/socket as follows (refer to diagram):
   - Mains socket:
     Insert the mains socket into the operating voltage connection on the valve terminal.
     Unscrew the housing from the socket.
     Remove the connection part from the socket, located in the operating voltage connection.
   - Sensor connector/diagnosis socket:
     Unfasten the middle knurled nut.
2. Open the strain relief on the rear part of the housing. Then guide your cable through as follows (refer to diagram).

Outer diameter (OD) of cable
PG7:  4.0...6.0 mm
PG9:  6.0...8.0 mm
PG13.5: 10.0...12.0 mm

Plug/socket (straight or angled):
Mains socket:  PG7, 9 or 13.5
Sensor socket:  PG7
Bus cable socket:  PG7, 9 or 13.5

Fig. A/1: Socket/plug components and cable routing
3. Remove 5 mm of the insulation from the ends of the conductors.

4. Fit cable strands with cable end sleeve.

5. Connect up the ends of the cables.

6. Reconnect the mains section to the housing on the plug/socket and screw the two parts together. Then retract the cable until there are no cable loops in the housing.

7. Tighten the strain relief.
A.2 CABLE LENGTH AND WIRE CROSS SECTION
A.2 CABLE LENGTH AND WIRE CROSS SECTION

Determining with the use of graphs . . . . . A-10
Determining with the use of formulae . . . . . A-12
A.2 CABLE LENGTH AND WIRE CROSS SECTION

PLEASE NOTE
The following information assumes a knowledge of the Chapters entitled "Installation" in this Manual and is aimed exclusively at specialist staff with a training in electrical engineering.

A load-dependent voltage drop occurs on all three cores in the operating voltage supply to a valve terminal. This can lead to the voltage on pin 1 or pin 2 of the operating voltage connection falling outside the permissible tolerance range.

Recommendation:
• Avoid long distances between power pack and valve terminal.
• Calculate the appropriate cable length and wire cross section in accordance with the following graphs or formulae. Bear in mind that the graphs
  – supply approximate values for cross sections 1.5 and 2.5 mm\(^2\).
  – The formulae supply precise values for any cross section.

PLEASE NOTE
The following graphs and formulae assume that the core cross sections on the operating voltage supply (pins 1, 2 and 3) are identical.
Determining with the use of graphs

Proceed as follows:

1. Calculate the maximum current consumption for outputs/valves ($I_2$).

2. Calculate the lowest voltage to be anticipated during operation ($V_{O\text{min}}$) on the power supply unit. Take due account when doing this of:
   - the load-dependency of the power supply unit.
   - the fluctuation in primary mains power voltage.

3. Read off the permissible cable length from the relevant table for your cross section.
   Example for 1.5 mm$^2$:
   $V_{O\text{min}} = 22.8$ V,
   $I_2 = 2$ A,
   $L_{\text{max}} = 25$ m
A.2 Cable length...

**Cross section 1.5 mm²**
(AWG 16)

**Cross section 2.5 mm²**
(AWG 14)
Determining with the use of formulae

Proceed as follows:

1. Calculate the maximum current consumption of the inputs and electronics ($I_1$) as well as of outputs/valves ($I_2$).

2. Calculate the lowest voltage anticipated during operation ($V_{Omin}$) on the mains power device. Take account of the following when doing so:
   - The load-dependency of the power supply unit.
   - The fluctuation in primary mains voltage.

3. Enter the values in the corresponding formulae. The substitute circuit diagram and the example explain the relationship.

![Substitute circuit diagram]

**Fig. A/3: Cable length ($L$) and line resistor ($R_L$)**
Formula for max. cable length:

\[ L \leq \frac{(V_{\text{Omin}} - V_{\text{TERMINALmin}}) \cdot A \cdot \kappa_{\text{Cu}}}{2 \cdot I_2 + h} \]

Key to terms:
- \( V_{\text{TERMINAL}} = 24\, V \pm 10\% \), minimum: \( V_{\text{TERMINALmin}} \geq 21.6\, V \)
- \( V_{\text{Omin}} \) = minimum operating voltage power supply (at power supply unit)
- Current \( I_1 \) = current for electronic components and inputs
- Current \( I_2 \) = current for outputs/valves
- \( A \) = wire cross-section (uniform size e.g. 1.5 mm\(^2\))
- \( \kappa \) = conductance of the wires
  (uniform size e.g. \( \kappa_{\text{Cu}} = 56 \frac{m}{mm^2 \cdot \Omega} \))

Example:
- \( I_1 = 1\, A; \)
- \( I_2 = 5\, A; \)
- \( V_o = 24\, V; \)
- \( V_{\text{TERMINALmin}} = 21.6\, V; \)
- \( \kappa_{\text{Cu}} = 56 \frac{m}{mm^2 \cdot \Omega}; \)

result of example:
\[ L \leq 18\, m \text{ for } A = 1.5\, mm^2 \]
\[ L \leq 30\, m \text{ for } A = 2.5\, mm^2 \]
A.3 EARTHING
Contents

A.3 EARTHING

Example of connection . . . . . . . . . . . . . . . . . . A-17
A.3 EARTHING

Example of connection

The following diagram shows the connection of a common 24 V power supply for pin 1 and pin 2. Please note that:

- the power supply for outputs/valves has to be protected against short circuits (s.c.)/overloads externally with a maximum fuse rating of 10 A,
- the power supply for electronic components and inputs has to be protected against short circuits (s.c.)/overloads externally with a maximum fuse rating of 3.15 A,
- the common tolerance 24 V DC ± 10 % must be observed,
- both mains earth cables must be connected for potential compensation and compensation currents must be prevented.
Fig. A/4: Example for connection of a common 24 V power supply and both mains earth cables.
Programmable valve terminal
with
control block SF 3

APPENDIX B:
Summary of operands (I/Os)
Function modules (CFM)
Error messages
Contents

B.1 SUMMARY OF OPERANDS (I/Os) . . . . . . . B-3
   I/O Addresses for control block SF 3 . . . . . . B-4
   Diagnosis I/Os . . . . . . . . . . . . . . . . . . . . . . . . B-5
   Field bus I/Os . . . . . . . . . . . . . . . . . . . . . . . . B-6

B.2 SUMMARY OF
   FUNCTION MODULES (CFM) . . . . . . . . . . . B-9

B.3 SUMMARY OF
   THE ERROR MESSAGES . . . . . . . . B-77
B.1 SUMMARY OF OPERANDS (I/Os)
Contents

B.1 SUMMARY OF OPERANDS (I/Os)
   I/O Addresses for control block SF 3 . . . . . . . . . . . . B-4
   Diagnosis I/Os . . . . . . . . . . . . . . . . . . . . . . . . . . . B-5
   Field bus I/Os . . . . . . . . . . . . . . . . . . . . . . . . . . . . B-6
B.1 SUMMARY OF OPERANDS (I/Os)

The next section illustrates the maximum extension for type 03 valve terminals with AS-i master and CP module in master operating mode.

### Table: Summary of Operands

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>Master operating mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW0...7, OW0...7</td>
<td><strong>Local</strong> Inputs, outputs, valves</td>
</tr>
<tr>
<td>IW8...15, OW8...15</td>
<td><strong>CP</strong> Inputs, outputs, valves</td>
</tr>
<tr>
<td>IW16...31, OW16...31</td>
<td><strong>AS-i master</strong> AS-i inputs, AS-i outputs</td>
</tr>
<tr>
<td>IW0.0...0.3</td>
<td><strong>Diagnosis</strong> Local I/O</td>
</tr>
<tr>
<td>IW0.4...0.15</td>
<td><strong>Diagnosis</strong> Field bus I/O</td>
</tr>
<tr>
<td>IW1.0...1.15, OW1.0...1.15</td>
<td>Field bus station slave 1</td>
</tr>
<tr>
<td>IW2.0...2.15, OW2.0...2.15</td>
<td>Field bus station slave 2</td>
</tr>
<tr>
<td>IW31.0...31.15, OW31.0...31.15</td>
<td>Field bus station slave 31</td>
</tr>
</tbody>
</table>

**Fig. B/1: Summary of operands and I/O addresses**
I/O Addresses for control block SF 3

Distribution of local I/Os, AS-i I/Os and CP I/Os on a valve terminal with control block SF 3.

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>without CP/AS-i</th>
<th>with AS-i</th>
<th>with CP</th>
<th>with CP/AS-i</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Local inputs and outputs</td>
<td>Local inputs and outputs</td>
<td>Local inputs and outputs</td>
<td>Local inputs and outputs</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. B/2a: Address distribution for local I/Os*
Diagnosis I/Os

Distribution and definition of diagnosis I/Os depending on the operating mode of control block SF 3.

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>Word width [bit]</th>
<th>Master operation</th>
<th>Slave operation</th>
<th>Standalone (without field bus)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>8</td>
<td>I/O diagnosis bytes</td>
<td>I/O diagnosis bytes</td>
<td>I/O diagnosis bytes</td>
</tr>
<tr>
<td>0.1</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.2</td>
<td>8</td>
<td>Short circuit (s.c.) Bit no.</td>
<td>Short circuit (s.c.) Bit no.</td>
<td>Short circuit (s.c.) Bit no.</td>
</tr>
<tr>
<td>0.3</td>
<td>8</td>
<td>Short circuit (s.c.) Byte no.</td>
<td>Short circuit (s.c.) Byte no.</td>
<td>Short circuit (s.c.) Byte no.</td>
</tr>
<tr>
<td>0.4</td>
<td>8</td>
<td>Transmission errors in field bus stations 1...7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>8</td>
<td>Transmission errors in field bus stations 8...15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.6</td>
<td>8</td>
<td>Transmission errors in field bus stations 16...23</td>
<td>Via field bus cyclically exchanged data range.</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>8</td>
<td>Transmission errors in field bus stations 24...31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>8</td>
<td>Diagnosis error in field bus stations 1...7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>8</td>
<td>Diagnosis error in field bus stations 8...15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.10</td>
<td>8</td>
<td>Diagnosis error in field bus stations 16...23</td>
<td>Size configurable 0...12 bytes Default: 2 bytes</td>
<td></td>
</tr>
<tr>
<td>0.11</td>
<td>8</td>
<td>Diagnosis error in field bus stations 24...31</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.12</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.13</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.14</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.15</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Fig. B/2b: Address distribution for diagnosis I/Os*
Field bus I/Os

Distribution of field bus I/Os depending on word width of field bus station (8 or 16 bit organisation).

<table>
<thead>
<tr>
<th>IW/OW</th>
<th>Word width [bit]</th>
<th>Field bus station slave 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>8/16</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>8/16</td>
<td>0...8 words for inputs</td>
</tr>
<tr>
<td>1.2</td>
<td>8/16</td>
<td>0...8 words for outputs</td>
</tr>
<tr>
<td>1.3</td>
<td>8/16</td>
<td>or</td>
</tr>
<tr>
<td>1.4</td>
<td>8/16</td>
<td>0...16 bytes for inputs</td>
</tr>
<tr>
<td>1.5</td>
<td>8/16</td>
<td>0...16 bytes for outputs</td>
</tr>
<tr>
<td>1.6</td>
<td>8/16</td>
<td></td>
</tr>
<tr>
<td>1.7</td>
<td>8/16</td>
<td>End with 16 bit organisation</td>
</tr>
<tr>
<td>1.8</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.9</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.10</td>
<td>8</td>
<td>End with 8 bit organisation</td>
</tr>
<tr>
<td>1.11</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.12</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.13</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.14</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>1.15</td>
<td>8</td>
<td>End with 8 bit organisation</td>
</tr>
<tr>
<td>2.0</td>
<td>8/16</td>
<td>Field bus station slave 2</td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>31.0</td>
<td>8/16</td>
<td>Field bus station slave 31</td>
</tr>
<tr>
<td>31.1</td>
<td>8/16</td>
<td>0...8 words for inputs</td>
</tr>
<tr>
<td>31.2</td>
<td>8/16</td>
<td>0...8 words for outputs</td>
</tr>
<tr>
<td>31.3</td>
<td>8/16</td>
<td>or</td>
</tr>
<tr>
<td>31.4</td>
<td>8/16</td>
<td>0...16 bytes for inputs</td>
</tr>
<tr>
<td>31.5</td>
<td>8/16</td>
<td>0...16 bytes for outputs</td>
</tr>
<tr>
<td>31.6</td>
<td>8/16</td>
<td>End with 16 bit organisation</td>
</tr>
<tr>
<td>31.7</td>
<td>8/16</td>
<td></td>
</tr>
<tr>
<td>31.8</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.9</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.10</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.11</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.12</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.13</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.14</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>31.15</td>
<td>8</td>
<td>End with 8 bit organisation</td>
</tr>
</tbody>
</table>

*Fig. B/3: Address distribution of field bus I/Os*
B.2 SUMMARY OF FUNCTION MODULES (CFM)
Contents

B.2 SUMMARY OF FUNCTION MODULES (CFM) . . . . . . . . . . . . . . . . . B-9
### B.2 SUMMARY OF FUNCTION MODULES (CFM)

<table>
<thead>
<tr>
<th>CFM no.</th>
<th>Application</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td><strong>SF 3</strong></td>
<td>Reset internal operands</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>Localise short circuit (s.c.)</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Indirect setting/resetting of local outputs</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Indirect access to data on FU0 to FU4095</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Measurement of program run time</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Read remanent data words</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>Write remanent data words</td>
</tr>
<tr>
<td>10</td>
<td></td>
<td>Parametrize/read out interrupt-controlled counters/timers</td>
</tr>
<tr>
<td>11</td>
<td></td>
<td>Block/release interrupt-controlled counters/timers</td>
</tr>
<tr>
<td>21</td>
<td>CP module</td>
<td>Read and write data of extra CP module</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td>Reset all outputs which can be accessed by CP</td>
</tr>
<tr>
<td>25</td>
<td></td>
<td>Diagnosis of CP valve terminal, input and output modules</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>Set parameters for reaction to CP errors</td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>Record CP configuration</td>
</tr>
<tr>
<td>31</td>
<td><strong>AS-i master/AS-i bus system</strong></td>
<td>Read AS-i slave parameter</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>Write AS-i slave parameter</td>
</tr>
<tr>
<td>33</td>
<td></td>
<td>Reset all outputs which can be accessed via AS-i bus</td>
</tr>
<tr>
<td>35</td>
<td></td>
<td>Diagnosis of all AS-i slaves</td>
</tr>
<tr>
<td>37</td>
<td></td>
<td>Set parameters for reaction of SF 3 to AS-i errors</td>
</tr>
<tr>
<td>38</td>
<td></td>
<td>Record actual configuration of AS-i bus</td>
</tr>
<tr>
<td>40</td>
<td><strong>SF 3 field bus</strong></td>
<td>Interrogate field bus configuration</td>
</tr>
<tr>
<td>41</td>
<td></td>
<td>Master/slave mode: Read parameters of a field bus station</td>
</tr>
<tr>
<td>42</td>
<td></td>
<td>Master/slave mode: Write parameters of a field bus station</td>
</tr>
<tr>
<td>43</td>
<td></td>
<td>Reset all outputs which can be accessed by field bus</td>
</tr>
<tr>
<td>44</td>
<td></td>
<td>Read status of field bus station</td>
</tr>
<tr>
<td>47</td>
<td></td>
<td>Set parameters for reaction to field bus errors</td>
</tr>
<tr>
<td>48</td>
<td></td>
<td>Record the ACTUAL configuration of field bus</td>
</tr>
<tr>
<td>49</td>
<td></td>
<td>Compare the ACTUAL list with the SET POINT list</td>
</tr>
<tr>
<td>50</td>
<td></td>
<td>Read out information of a field bus station</td>
</tr>
<tr>
<td>51</td>
<td></td>
<td>Reset field bus station</td>
</tr>
</tbody>
</table>

*Fig. B/4a: Summary of function modules*
<table>
<thead>
<tr>
<th>CFM no.</th>
<th>Application</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>Analogue modules</td>
<td>Read analogue values</td>
</tr>
<tr>
<td>61</td>
<td></td>
<td>Output analogue values</td>
</tr>
<tr>
<td>63</td>
<td></td>
<td>Diagnosis of analogue modules/channels</td>
</tr>
<tr>
<td>90</td>
<td>SF 3</td>
<td>Call Assembler programs (function modules)</td>
</tr>
<tr>
<td>91</td>
<td></td>
<td></td>
</tr>
<tr>
<td>92</td>
<td></td>
<td></td>
</tr>
<tr>
<td>93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>94</td>
<td></td>
<td></td>
</tr>
<tr>
<td>95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>96</td>
<td></td>
<td></td>
</tr>
<tr>
<td>97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>98</td>
<td></td>
<td></td>
</tr>
<tr>
<td>99</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. B/4b: Summary of function modules
P1 has the following definition:

0: Reset all registers, flags, timers and counters
1: Reset all registers
2: Reset all flags
3: Reset all timers and counters
4: Reset FU48...FU4095

Parameter P1 can be a constant (e.g. K3) or a variable (e.g. R33).

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number 100
Comments:
In a call, the first short-circuited electrical output is reported back.

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful
P2 (FU33) = {-1} no short circuits (s.c.) occurred or no outputs available
P3 (FU34) = without definition

Case 2:
P1 (FU32) = {-1} processing successful
P2 (FU33) = word number of first short circuited output
P3 (FU34) = bit number

Case 3:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number} possible error number 100, 110

The function modules 1 and 2 are suitable for diagnosis and error handling. For further details, consult Chapter 3.4 “Diagnosis and error handling”.
The parameters have the following definition:
P1 = 0: Reset output  
     = 1: Set output
P2 = Word number of output
P3 = Bit number of output

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number
100, 101, 102, 103

The function modules 1 and 2 are suitable for diagnosis and error handling. For further details, consult Chapter 3.4 "Diagnosis and error handling".
The parameter has the following definition:
P1 = Number of special operands FU0...4095 (read access)

**Input format**
THEN
WITH <P1>

**Parameter**
P1 = {0...4095}

**Return parameters**

**Case 1:**
P1 (FU32) = {-1} processing successful (read access)
P2 (FU33) = value of the selected special operand (FU)

**Case 2:**
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number 100, 101
The parameter has the following definition:

\[
\begin{align*}
P1 &= \text{Number of special operands FU0...4095} \\
    &\quad \text{(write access)} \\
P2 &= \text{New contents of selected special operand (FU)}
\end{align*}
\]

**Input format**

\[
\text{THEN} \quad \text{CFM 3} \\
\text{WITH <P1>} \\
\text{WITH <P2>}
\]

**Parameters**

\[
P1 = \{0...4095\}
\]

**Return parameters**

**Case 1:**

\[
P1 (FU32) = \{-1\} \quad \text{processing successful} \\
\quad \text{(write access)}
\]

**Case 2:**

\[
P1 (FU32) = \{0\} \quad \text{processing faulty} \\
P2 (FU33) = \{\text{error number}\} \\
    \text{possible error number} \\
    100, 101
\]
P1 has the following definition:
0...15 Starts measurement of cycle time for programs 0...15
16 Starts measurement of a program section
17 Stop measurement

The result of the time measurement is stored in special operand FU3 (time in ms) and FU4 (time in µs).

Also note the comments on the following page.

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful
FU3 = result in ms
FU4 = result in µs

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number
100, 101
Note on CFM 4:
Measurement of cycle time:
With cycle times of programs 0...15, the maximum value is determined from the total measuring time.

Measurement of a program section:
Due to interrupt-related events during program processing, repeated measurement of the relevant time window (program section) can give rise to differing measurement results.

PLEASE NOTE
Please note that the program cycle time is increased during the run-time of a program if the FST 200 is used in Online Mode at the same time.
Function

Read one or several remanent data words (maximum 15), depending on the parameters transferred.

Input format

THEN  
CFM 5  
WITH <P1>  
WITH <P2> (optional)

Parameters

P1 = {0...511}  
Address of remanent data word n

P2 = {1...15}  
(optional)  
Number of remanent data words to be read out as from address selected with P1

Return parameters

Case 1:

P1 (FU32) = {-1} processing successful  
P2 (FU33) = Contents of selected address n  
Optional (only when module is accessed with appropriate parameter P2):

P3 (FU34) = Contents of address n + 1
...

P16 (FU47) = Contents of address n + 14

Case 2:

P1 (FU32) = {0} processing faulty  
P2 (FU33) = {error number}  
possible error number  
100, 101, 106 (hardware status < 1097)
Function

Write one or several remanent data words (maximum 15), depending on the parameters transferred.

Input format

THEN CFM 6
WITH <P1>
WITH <P2>
WITH <P3> (optional)
... (optional)
WITH <P16> (optional)

Parameters

P1 = {0...511} Address of remanent data word n
P2 = {0...65535} New contents of selected address n
P3 = {0...65535} New contents of selected address n + 1 (optional)
...
P16 = {0...65535} New contents of selected address n + 14 (optional)

Return parameters

Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number} possible error number
100, 101, 106 (hardware status < 1097)

The remanent data written with function module 6 are not read by the FST 200 utility "upload".
Function

Parametrizing or reading out interrupt-controlled counters/timers. The counters/timers are started and stopped with CFM 11.

Input format

THEN CFM 10
WITH <P1>
WITH <P2> (optional)
WITH <P3> (optional)
WITH <P4> (optional)
WITH <P5> (optional)

Parameters

Variant 1: Definition of counters/timers

Number of transfer parameters = 5

P1: Number and mode of counter/timer
P2: Trigger source and function
P3: Trigger destination output
P4: Trigger destination flag
P5: Starting value for counter/timer

Variant 2: Read out current counter/timer value

Number of transfer parameters = 1

P1: Number of counter/timer

Return parameters

Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number
100, 101, 102*, 103*, 104*

*) Only with variant 1
Explanation of the parameters.

**PLEASE NOTE**
The values last parametrized with CFM 10 are retained. The parametrization of the counters/timers needs therefore to be carried out only once, if it does not change.
This applies especially to counters/timers in normal mode, i.e. they can be reactivated by means of CFM 11.

With function modules 10 and 11, you can register fast counting procedures (operating mode counter) or time-dependent events (operating mode timer), irrespective of the cycle time required for processing user programs. Parametrizing is carried out with function module 10; the interrupt-controlled counters/timers are started or stopped with function module 11.

**Counter operating mode**
In the counter operating mode, you can implement counter inputs up to 300 Hz together with the "fast" input module (input signal delay of 1 ms) by triggering on the positive and/or negative edge.

**Timer operating mode**
In the timer operating mode, time-dependent events with a resolution of ± 1 ms can be controlled, either with or without gate function. The gate function is determined as an input or flag by means of the trigger source ascertained in transfer parameter P2.
The user can decide how to divide the interrupt functions (max. 4) into the operating modes timer or counter. The user can also define the direction of counting (upwards/downwards). When the event is triggered (underrun or overrun of counter/timer), an output and/or a flag can be set, reset or reactuated (toggle function) as desired.

**PLEASE NOTE**

If an output is specified as the trigger destination, not only will this output be written in direct access when the trigger event occurs, but the other outputs on this module will also be written with the current values in the process image. This procedure differs from the usual method of operation of the cyclic update of the I/O peripherals after a task change.

In the Reload mode, the parametrized starting value is loaded again automatically into the current counter value or timer value when the event is triggered. The interrupt remains released until blocking is explicitly made by means of function module 11. In normal mode (operation without reload function), the interrupt is blocked with the trigger event. The counter/timer can be triggered later in the user program if function module 11 is accessed.
If only one parameter, which corresponds to the counter/timer number, is transferred to function module 10, the return parameter will contain the current counter/timer value. This enables e.g. the impulse duration of an input signal to be measured.

The use of interrupt-controlled timers/counters has hardly any influence on the time reaction of the SF 3.

**Variant 1: Definition of the counter/timer**

**Number of transfer parameters = 5**

Transfer parameter P1:
number and mode of counter/timer

<table>
<thead>
<tr>
<th>High byte P1</th>
<th>Low byte P1</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
<tr>
<td>X X X 0 0 0 0 0</td>
<td>Number 1...4</td>
</tr>
</tbody>
</table>

- 0 Operating mode Timer
- 1 Operating mode counter

- 0 Normal mode: interrupt is blocked during trigger event. Counter/timer is stopped.
- 1 Reload mode: interrupt is not blocked during trigger event. Counter/timer is loaded with starting value and continues to run.

- 0 Counting direction backwards (trigger at underrun 1 \(\rightarrow\) 0)
- 1 Counting direction forwards (trigger at overrun 65535 \(\rightarrow\) 0)
Transfer parameter P2: 
trigger source and function

<table>
<thead>
<tr>
<th>High byte P2</th>
<th>Low byte P2</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
<tr>
<td>X X X Bit number</td>
<td>Word number</td>
</tr>
</tbody>
</table>

---

**Input as trigger source**

- **Meaning with operating mode timer:**
  - 0 0 0 without gate function
  - 0 1 0 with gate function input low level
  - 1 0 0 with gate function input high level

- **Meaning with operating mode counter:**
  - 0 0 0 fast input on positive edge
  - 0 1 0 fast input on negative edge
  - 1 0 0 fast input on positive and negative edges

---

**Flag as trigger source**

- **Meaning with operating mode timer:**
  - 0 0 1 without gate function
  - 0 1 1 with gate function flag low level
  - 1 0 1 with gate function flag high level

- **Meaning with operating mode counter:**
  - 0 0 1 flag on positive edge
  - 0 1 1 flag on negative edge
  - 1 0 1 flag on positive and negative edges

States not represented are not permitted
### Transfer parameter P3: trigger destination output

<table>
<thead>
<tr>
<th>Bit number</th>
<th>Word number</th>
</tr>
</thead>
<tbody>
<tr>
<td>High byte P3</td>
<td>Low byte P3</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

0 0  Output is not influenced  
0 1  Output is set during trigger event  
1 0  Output is reset during trigger event  
1 1  Output is reversed during trigger event

### Transfer parameter P4: trigger destination flag

<table>
<thead>
<tr>
<th>Bit number</th>
<th>Word number</th>
</tr>
</thead>
<tbody>
<tr>
<td>High byte P4</td>
<td>Low byte P4</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

0 0  Flag is not influenced  
0 1  Flag is set during trigger event  
1 0  Flag is reset during trigger event  
1 1  Flag is reversed during trigger event

### Transfer parameter P5: starting value for counter/timer

**Variant 2: Read out current counter/timer value**  
**Number of transfer parameters = 1**

### Transfer parameter P1: number of counter/timer

<table>
<thead>
<tr>
<th>Bit number</th>
<th>Word number</th>
</tr>
</thead>
<tbody>
<tr>
<td>High byte P1</td>
<td>Low byte P1</td>
</tr>
<tr>
<td>7</td>
<td>6</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Function
Blocking and releasing interrupt-controlled counters/timers.
The counters/timers are parametrized or read out with CFM 10.

Input format
THEN CFM 11
WITH <P1>

Parameter
P1 = Number and state of counter/timer

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = possible error number
100, 101

Explanation of parameter see below.
Transfer parameter P1:
Number and status of the counter/timer

<table>
<thead>
<tr>
<th>High byte P1</th>
<th>Low byte P1</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 6 5 4 3 2 1 0</td>
<td>7 6 5 4 3 2 1 0</td>
</tr>
<tr>
<td>0 0 X X 0 0 X X 0 0 X X</td>
<td></td>
</tr>
</tbody>
</table>

Counter/timer 4
- Interrupt no change: X 0
- Block interrupt (disable): 0 1
- Release interrupt (enable): 1 1

Counter/timer 3
- Interrupt no change: X 0
- Block interrupt (disable): 0 1
- Release interrupt (enable): 1 1

Counter/timer 2
- Interrupt no change: X 0
- Block interrupt (disable): 0 1
- Release interrupt (enable): 1 1

Counter/timer 1
- Interrupt no change: X 0
- Block interrupt (disable): 0 1
- Release interrupt (enable): 1 1
Function
Simultaneous call of read and write functions on the X modules\(^1\) of the CP system.
Parameter P2...P5 can be specified with the call.
Depending on the relevant X module, up to four parameters are returned (refer to description of relevant X module).

Return parameters
Case 1:
- P1 (FU32) = {-1} processing successful (read and write access)
- P2 (FU33) = 1st data word (read access)
- P3 (FU34) = 2nd data word (read access)
- P4 (FU35) = 3rd data word (read access)
- P5 (FU36) = 4th data word (read access)

Case 2:
- P1 (FU32) = {0} processing faulty
- P2 (FU33) = {error number} possible error number
  - 19, 20, 21, 100

Input format
THEN CFM 21
WITH <P1>
WITH <P2>
...
WITH <P5>

Parameters
- P1 = {0...3}
- P2 = write 1st data value (16 bit)
- P3 = write 2nd data value (16 bit)
- P4 = write 3rd data value (16 bit)
- P5 = write 4th data value (16 bit)

1) X modules in preparation
**Return parameters**

**Case 1:**
- P1 (FU32) = {-1} processing successful

**Case 2:**
- P2 (FU32) = {0} processing faulty
- P3 (FU33) = {error number}
  - possible error number
  - 19, 100

<table>
<thead>
<tr>
<th>CFM 23</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input format</td>
</tr>
<tr>
<td>THEN CFM 23</td>
</tr>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>no parameter</td>
</tr>
</tbody>
</table>

Reset all outputs which can be accessed by CP
**Recommendation:**
The diagnosis bytes always occupy 0...7 of the return code parameter (bit 8...15 always 0). The following diagram provides a key to these diagnosis bytes:

**Input format**
THEN CFM 25

**Parameter**
no parameter

**Return parameters**

**Case 1:**
P1 (FU32) = {-1} processing successful
P2 (FU33) = diagnosis byte of string 0
P3 (FU34) = diagnosis byte of string 1
P4 (FU35) = diagnosis byte of string 2
P5 (FU36) = diagnosis byte of string 3

**Case 2:**
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number
19, 100

**Diagnosis of CP valve terminal, input and output modules**

**Input format**
THEN CFM 25

**Parameter**
no parameter
Significance of diagnosis bytes from branch 0...3

<table>
<thead>
<tr>
<th>FU-no.</th>
<th>Branch Type</th>
<th>Bit no.</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>33</td>
<td>CP modules</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

IX CP connection interrupted X module
(* X module in preparation)
Iinp CP connection interrupted input module
Iout CP connection interrupted output module
(valve terminal or electric output module)
V Val dropped below voltage tolerance for valves
V Sen short circuit (s.c.) for sensor supply
V Out load voltage failure of output module CP
sc/oshort circuit (s.c.)/overload output module CP

Recommendation:
- The collective message "failure of CP component(s)" I0.1.1 comprises bits 0, 1, 7 of diagnosis bytes of branch 0...3.
- The collective message "single error, CP component(s)" I0.1.2 comprises bits 2...5 of diagnosis bytes of branch 0...3 (also refer to Chapter 3.4 "Diagnosis and error handling").
P1 has the following definition:
0: CP error stops all programs (hard reaction).
1: CP error does not trigger STOP. Treatment of error possible in user program (soft reaction).

Return parameters
Case 1:
P1 (FU32) = {-1} processing successful
Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
   possible error number
   19, 100, 101
P1 has the following definition:
0: Recording configuration without storage
   (SETPOINT/ACTUAL-value comparison)
1: Recording configuration with storage
   (compare SAVE key)

Return parameters
Case 1:
P1 (FU32) = {-1} configuration started
Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
   possible error number
   19, 100

Input format
THEN CFM 28
WITH <P1>

Parameter
P1: = {0, 1}
Function

The current parameter (0...F) of AS-i slaves is read and returned in hexadecimal form to parameter P2 (FU33). The definition of the parameter is listed in the relevant description of the AS-i slave. If the AS-i slave called does not have a parameter processing feature, FH is always returned. No error message is issued.

Input format

THEN

CFM 31

WITH <P1>

Parameter

P1: = {0...31} (AS-i slave address)

Return parameters

Case 1:

P1 (FU32)= {-1} processing successful
P2 (FU33)= {0...F} current AS-i slave parameter

Case 2:

P1 (FU32)= {0} processing faulty
P2 (FU33)= {error number} possible error number
14, 15, 100, 101
Function

Writes a value 0...F (hexadecimal) in the addressed AS-i slave.

Input format

THEN CFM 32
WITH <P1>, <P2>

Parameter

P1 = {0...31}
P2 = {0...F}

The parameters have the following definition:
P1 = AS-i slave address
P2 = slave parameters for AS-i slaves on which parameters can be set

Return parameters

Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number} possible error number 15, 100, 101, 102

Recommendation:
Consult the manual for your AS-i slave to find details about setting parameters of AS-i slaves and the effects of this operation.

CAUTION
- Only alter parameters if you know what effect this will have.
- If the system is switched on, the slaves will respond immediately to your entry.
- Ensure that modification of the slave parameters does not create a risk of injury to people or damage to equipment.
Return parameters

Case 1:
P1 (FU32) = {-1} processing successful

Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number}
possible error number
14, 100
Function

When using this module, you receive the diagnosis information of all configured AS-i slaves on your AS-i bus system.

The diagnosis result is prepared in special function units FU33 to FU34. With appropriate programming techniques, you can either receive collective information or information about individual AS-i slaves.

Input format

THEN CFM 35

Parameter

no parameter

Only the configured slaves can cause errors.

Return parameters

Case 1:

P1 (FU32) = {-1} processing successful
P2 (FU33) = error AS-i slave 1...15
P3 (FU34) = error AS-i slave 16...31

Case 2:

P1 (FU32) = {0} processing faulty
P2 (FU33) = (error number)
possible error number
14, 100
The diagnosis result described in the following section will only occur if function module CFM 35 reports successful processing (FU32 = -1).

**Diagnosis result in FU33...FU34 (FU32 = -1)**

**FU33:**

- **AS-i slave address**
- **Data bit number**

<table>
<thead>
<tr>
<th>D15</th>
<th>D14</th>
<th>D13</th>
<th>D12</th>
<th>D11</th>
<th>D10</th>
<th>D9</th>
<th>D8</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>X</td>
</tr>
</tbody>
</table>

**FU34:**

<table>
<thead>
<tr>
<th>D15</th>
<th>D14</th>
<th>D13</th>
<th>D12</th>
<th>D11</th>
<th>D10</th>
<th>D9</th>
<th>D8</th>
<th>D7</th>
<th>D6</th>
<th>D5</th>
<th>D4</th>
<th>D3</th>
<th>D2</th>
<th>D1</th>
<th>D0</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>30</td>
<td>29</td>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>24</td>
<td>23</td>
<td>22</td>
<td>21</td>
<td>20</td>
<td>19</td>
<td>18</td>
<td>17</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No error on AS-i slave</td>
</tr>
<tr>
<td>1</td>
<td>AS-i slave failed</td>
</tr>
<tr>
<td>x</td>
<td>Bit value not important</td>
</tr>
</tbody>
</table>

Databits of non-existing AS-i slaves have the log. value "0". Only figured slaves can cause an error.
If any parameter is transferred when function module 35 is accessed, the address of the first faulty AS-i slave will be shown as a decimal value in return parameter P4. This value is zero if the registered AS-i slave is functioning correctly. The first faulty AS-i slave is the lowest-value slave which at present has a fault.
P1 has the following definition:
0: AS-i error stops all programs (hard reaction).
1: AS-i error does not trigger STOP. Treatment of error possible in application program (soft reaction).

Input format
THEN CFM 37
WITH <P1>

Parameter
P1: = {0, 1}

Return code parameters
Case 1:
P1 (FU32) = {-1} processing successful
Case 2:
P1 (FU32) = {0} processing faulty
P2 (FU33) = {error number} possible error number
14, 100, 101
Function

Recording the ACTUAL configuration, compiling an ACTUAL list.

The functional module performs the following function:
• User-controlled starting of a re-configuration phase of the AS-i bus system.

CAUTION
During the configuration run, the SF 3 controller stops. All AS-i outputs are switched off. AS-i inputs are not updated for approx. 2 seconds.

Input format
THEN CFM 38

Parameter
no parameter

Return parameters
Case 1:
P1 (FU32)={-1} configuration performed
The result of SET POINT-ACTUAL comparison is indicated in AS-i status bit I16.2 and I16.3

Case 2:
P1 (FU32)={0} processing faulty
P2 (FU33)={error number}
possible error number
14, 100, 115
Function
After POWER ON, the SF 3 checks the configuration of slave modules installed. At this point, SETPOINT and ACTUAL data are compared with each another. Function module 40 interrogates the ACTUAL data and indicates these values in the return code parameter.

Input format
THEN CFM 40
WITH K <P1>

Parameter
P1: address of field bus station

Return parameters
Case 1:
FU32: {-1} SETPOINT and ACTUAL data correspond
FU32: {+1} SETPOINT and ACTUAL data are different
FU33: ACTUAL type of field bus slave (refer to table)
FU34: ACTUAL number of inputs of field bus slave (in bytes)
FU35: ACTUAL number of outputs of field bus slave (in bytes)

Case 2:
FU32 = 0 processing faulty
FU33: error number
FU33: possible error number

Example
THEN CFM 40
WITH K11

In SF 3, the configuration of fieldbus slave 11 is interrogated.
**ACTUAL type of the field bus slave:**

<table>
<thead>
<tr>
<th>FU33 (dec)</th>
<th>ACTUAL type of the field bus slave</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Valve terminal type 01* up to 8 valves (*type to be discontinued)</td>
</tr>
<tr>
<td>34</td>
<td>Valve terminal type 01* more than 8 valves (*type to be discontinued)</td>
</tr>
<tr>
<td>35</td>
<td>Valve terminal type 02 up to 8 valves</td>
</tr>
<tr>
<td>36</td>
<td>Valve terminal type 02 more than 8 valves</td>
</tr>
<tr>
<td>37</td>
<td>Valve/sensor terminal up to 6 valves</td>
</tr>
<tr>
<td>38</td>
<td>Valve/sensor terminal more than 6 valves</td>
</tr>
<tr>
<td>5</td>
<td>Modular valve terminal 03/04/05</td>
</tr>
<tr>
<td>129</td>
<td>Programmable valve terminal SF type 02 up to type 05 (slave)</td>
</tr>
<tr>
<td>4</td>
<td>FSI (FB-interface 19)</td>
</tr>
<tr>
<td>3</td>
<td>FB-202-I/O-Byte (regular addressing byte-by-byte)</td>
</tr>
<tr>
<td>33</td>
<td>FB-202-I/O-Word (exception addressing word oriented)</td>
</tr>
<tr>
<td>81</td>
<td>FB-405-I/O (only analogue I/Os)</td>
</tr>
<tr>
<td>161</td>
<td>FB-405-I/O (digital and/or analogue I/Os)</td>
</tr>
<tr>
<td>65</td>
<td>Open interface Jumo</td>
</tr>
<tr>
<td>66</td>
<td>Open interface RS 232</td>
</tr>
<tr>
<td>80</td>
<td>Open interface Hartmann &amp; Braun</td>
</tr>
<tr>
<td>70</td>
<td>Text display FD-2/40-F</td>
</tr>
<tr>
<td>166</td>
<td>FB-ADA</td>
</tr>
<tr>
<td>. . .</td>
<td>Status 4/96, others in preparation</td>
</tr>
</tbody>
</table>
SF 3 in "master" operating mode

Function
As a master, the SF 3 is able to read one parameter field per field bus slave of size 256 words of 16 bits. Function module 41 is able to read a selectable parameter field area in an "intelligent" field bus slave (e.g. SF-202 as slave). Each time the field bus module 41 is called, a 16-bit word addressed by parameter P2 in the parameter field is read, together with the four subsequent words.

Example
parameter field of a field bus slave:
256 words of 16 bits
Input format (master)
THEN CFM 41
WITH K <P1>
WITH K <P2>

Parameters
P1: address of field bus slave 1,..., 31
P2: address of parameter word n = 0,..., 255

Return parameters (master)
Case 1:
FU32 = {-1} command was processed successfully
FU33: status of field bus slave
FU34: parameter word of selected address n
FU35: parameter word of address n+1
FU36: parameter word of address n+2
FU37: parameter word of address n+3
FU38: parameter word of address n+4

Case 2:
FU32 = {0} processing faulty
FU33: error number
possible error number
100, 101, 102, 110, 111, 113
Example (master)

THEN CFM 41
WITH K9
WITH K11

In field bus slave 9, the 16-bit words 11, 12, 13, 14 and 15 are read, and the command success and slave status are indicated.

FU32: -1 or 0
FU33: Status
FU34: Word 11
FU35: Word 12
FU36: Word 13
FU37: Word 14
FU38: Word 15

Graphic representation:

**SF 3 in "slave" operating mode**

![Diagram](image)

**Fig. B/5: Example CFM 41 (master)**
Function

Every SF 3 as slave contains a parameter field of size 256 words of 16 bits. Function module 41 reads its own parameter field area in SF 3 as a slave. Each time field bus module 41 is called, a 16 bit word in the parameter field addressed by parameter P1 is read together with the five subsequent words.

Example

Parameter field of SF 3 as slave:
256 words of 16 bits

<table>
<thead>
<tr>
<th>WORD ADDRESS</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>...</th>
<th>n</th>
<th>n+1</th>
<th>n+2</th>
<th>...</th>
<th>n+5</th>
<th>255</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Addressed parameter word  \[ \text{Read parameter words} \]
### Input format (slave)

**THEN**
- CFM 41
- WITH K \(<P1>\)

**Parameter**
- \(P1\): address of parameter word
- \(n=0,\ldots, 255\)

### Return parameters (slave)

**Case 1:**
- \(FU32 = \{-1\}\) processing successful
- \(FU33\): parameter word of selected address \(n\)
- \(FU34\): parameter word of address \(n+1\)
- \(FU35\): parameter word of address \(n+2\)
- \(FU36\): parameter word of address \(n+3\)
- \(FU37\): parameter word of address \(n+4\)

**Case 2:**
- \(FU32 = \{0\}\) processing faulty
- \(FU33\): error number
- Possible error numbers:
  - 100, 101, 113
Example (slave)

THEN CFM 41
WITH K11

In the slave station, the words 11 to 15 will be read and command success will be indicated.

- FU32: -1 or 0
- FU33: Word 11
- FU34: Word 12
- FU35: Word 13
- FU36: Word 14
- FU37: Word 15

Graphic representation:

Fig. B/6: Example CFM 41 (slave)
SF 3 in *master* operating mode

**Function**

Function module 42 writes a selectable parameter field in a field bus slave (for parameter field, refer to CFM 41).

**Input format (master)**

THEN CFM 42
WITH K <P1>
WITH K <P2>
WITH K <P3>
WITH K <P4>
WITH K <P5>
WITH K <P6>
WITH K <P7>

**Parameter**

P1: address of field bus slave 1,..., 31
P2: address of parameter word n= 0,..., 255
P3: new value of parameter word with address n selected in P2
P4: new value of parameter word of address n+1 (optional)
P5: new value of parameter word of address n+2 (optional)
P6: new value of parameter word of address n+3 (optional)
P7: new value of parameter word of address n+4 (optional)

**Return parameters (master)**

**Case 1:**

FU32 = {-1} processing successful
FU33: status of field bus slave

**Case 2:**

FU32 = {0} processing faulty
FU33: error number
   possible error number
   100, 101, 102, 110, 111, 113,
Example (master)

THEN CFM 42
WITH K5
WITH K8
WITH K1000
WITH K33

In field bus station 5, word 8 (= n) will be written with value 1000 and word 9 (= n + 1) will be written with value 33.

Graphic representation:

CFM call (master)
THEN CFM 42
WITH K5
WITH K8
WITH K1000
WITH K33

Return code parameters
FU32 – order ok?
FU33 – status of FB slave

SF 3 parameter field in slave 5

Fig. B/7: Example CFM 42 (master)
SF 3 in "slave" operating mode

Function

Function module 42 writes a parameter field in SF 3 as slave (for parameter field, refer to CFM 41).

Input format (slave)

```
THEN CFM 42
  WITH K <P1>
  WITH K <P2>
  WITH K <P3>
  WITH K <P4>
  WITH K <P5>
  WITH K <P6>
```

Parameters

P1: address of parameter word n= 0,.., 255
P2: new value of parameter word with address n selected in P2
P3: new value of parameter word of address n+1 (optional)
P4: new value of parameter word of address n+2 (optional)
P5: new value of parameter word of address n+3 (optional)
P6: new value of parameter word of address n+4 (optional)

Return parameters (slave)

Case 1:

```
FU32 = {-1} processing successful
```

Case 2:

```
FU32 = {0} processing faulty
FU33: error number possible error number 100, 101, 113
```
Example (slave)

THEN CFM 42
WITH K8
WITH K1000
WITH K33

In the slave station, word 8 (= n) from the parameter field will be written with value 1000 and word 9 (= n+1) will be written with value 33.

Graphic representation:

Fig. B/8: Example CFM 42 (slave)
Function

Function module 43 has the function of resetting all outputs which can be accessed by the field bus in the SF 3 memory. This means that all cyclically activated outputs on the field bus have logic 0 state until they are reset by the program or by a direct command.

Input format

THEN   CFM 43

Return parameters

Case 1:

FU32 = \{-1\} processing successful

Case 2:

FU32 = \{0\} processing faulty
FU33: error number
possible error number
100, 113

Example

All SF 3 outputs accessible by the field bus are reset.

THEN   CFM 43
Function module 44 interrogates the status and error status of addressed field bus slaves.

If no parameter is transferred when function module 44 is accessed, the address of the first faulty field bus slave, as well as that of the first failed field bus slave are shown as decimal values in the return parameters. These values are zero if the registered field bus slaves are functioning correctly.

The term "first faulty or failed field bus slave" means the lowest value slave which at present has a functional fault.
Return parameters (access without transfer parameter P1)
Case 1:
FU32 = {-1} processing successful
FU33: address of the first failed field bus slave as decimal value
FU34: address of the first faulty field bus slave as decimal value
Case 2:
FU32 = {0} processing faulty
FU33: possible error number 100, 113

Return parameters (access with transfer parameter P1)
Case 1:
FU32 = {-1} processing successful
FU33: status of field bus slave
Case 2:
FU32 = {0} processing faulty
FU33: error number possible error number 100, 101, 113

Example
Read status of field bus slave 5.
THEN CFM 44
WITH K5
Function

This function module parametrizes the reaction of the SF 3 to field bus errors. The following error responses can be set:

**Input format**

THEN CFM 47
WITH K<P1>

**Parameter**

**Case 1:**

P1 = {0} Hard error reaction (default)

**Case 2:**

P1 = {1} Soft error reaction (error handling via user program possible)

**Return parameters**

**Case 1:**

FU32 = {-1} processing successful

**Case 2:**

FU32 = {0} processing faulty
FU33: {error number}
possible error number
100, 101, 113
Effects of preset characteristics

<table>
<thead>
<tr>
<th>Settings</th>
<th>Control unit characteristics</th>
<th>Error message</th>
<th>User characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFM 47 (Default): P1 = 0</td>
<td>Control unit stopped</td>
<td>Error 4</td>
<td>Remove the error</td>
</tr>
<tr>
<td>CFM 47: P1 = 1</td>
<td>Error will be corrected by error handling</td>
<td>Collective error bit I0.0.0. = 1</td>
<td>Previously: programming of error handling program is required</td>
</tr>
</tbody>
</table>

Fig. B/9: Characteristics for transmission errors during the operating phase

If I0.0.0. displays a "1" in master operating mode, the failed slave can be localised via IW0.4...0.7 (slaves 1...31) or FU1 (slaves 1...16).

Diagnosis byte IW0.4...0.7

<table>
<thead>
<tr>
<th>Bit no.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IW 7</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

Errors in transmission - station is present in bit-code format

Fig. B/10a: Possibility of diagnosis and error handling (IW0.4...0.7)

FU1, assignment:

<table>
<thead>
<tr>
<th>bit FU1 *)</th>
<th>15</th>
<th>14</th>
<th>13</th>
<th>12</th>
<th>11</th>
<th>10</th>
<th>9</th>
<th>8</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slave address</td>
<td>16</td>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
<td>9</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

*) '1' slave 1...16 failed
'0' no transmission fault

Fig. B/10b: Possibility of diagnosis and error handling (FU1)
Function

Recording the ACTUAL configuration, compiling the ACTUAL list.

The function module performs the following function:

- User-controlled initialisation of a re-configuration phase on the field bus system.

CAUTION

During the configuration run, all field bus outputs are switched off for approx. 2 seconds. Field bus inputs are not updated for approx. 2 seconds.

Input format

THEN CFM 48

Parameter

no parameter

Return parameters

Case 1:

FU32 = {-1} ACTUAL list = SETPOINT list
FU32 = {1} ACTUAL list <> SETPOINT list
FU32 = {2} Only the ACTUAL list is available

Case 2:

FU32 = {0} processing faulty
FU32: possible error number 100

Result is also stored in FU0

FU0 = {0} error occurred
FU0 = {2} ACTUAL list = SETPOINT list
FU0 = {3} ACTUAL list <> SETPOINT list
FU0 = {4} only the ACTUAL list is available
Function

Comparison of ACTUAL list with SETPOINT list.

The function module performs the following function:
- compares the ACTUAL list with the SETPOINT list, without re-configuration.

Input format
THEN CFM 49

Parameter
no parameter

Return parameters

Case 1:
- FU32 = {-1} ACTUAL list = SETPOINT list
- FU32 = {1} ACTUAL list <> SETPOINT list
- FU32 = {2} Only the ACTUAL list is available

Case 2:
- FU32 = {0} comparison not possible
- FU33: possible error number 100

Comment:
result is also stored in FU0
- FU0 = {0} error occurred
- FU0 = {2} ACTUAL list = SETPOINT list
- FU0 = {3} ACTUAL list <> SETPOINT list
- FU0 = {4} Only the ACTUAL list is available
Function

If a registered slave is to be clearly identified as the Festo field bus master by the SF 3 control block, the information as to which slave it concerns, can be read out as a clear text with the aid of function module 50 and stored as a string in the field for special operands FU48...FU4095 as follows:

Length of string with address n (FUn)
Contents of string as from address n+1 (FUn+1)

The address of the special operand will be specified as transfer parameter when it is accessed.

Input format

THEN CFM 50
  WITH <P1>
  WITH <P2>

Parameters

P1 = {1...31} Address of field bus station
P2 = {48...4095} Address n of special operand from which the information (string) is stored as follows:

Length of string with address n (= FUn)
Contents of string as from address n+1 (= FUn+1)
Example of a clear text message from an IFB5 bus node with a valve terminal type 03 as field bus slave "Terminal 03/FB5," the string would be stored as follows:

Return parameters

Case 1:
- FU32 = {-1} processing successful
- FU33 = status of field bus station

Case 2:
- FU32 = {0} processing faulty
- FU33 = possible error number
  - 100, 101, 102, 110, 111, 113

Example of a clear text message from an IFB5 bus node with a valve terminal type 03 as field bus slave "Terminal 03/FB5," the string would be stored as follows:

FU...FU+15 = 15,84,110,114,109,110,97,108,32,48,51,47,70,66,53

1st. ASCII character of string ...
Number of subsequent received ASCII characters
**Input format**

THEN

CFM 51

WITH <P1>

**Parameter**

P1: \{1...31\} address of field bus station

**Return parameters**

Case 1:

FU32 = \{-1\} processing successful

Case 2:

FU32 = \{0\} processing faulty

FU33: possible error number

100, 101, 110, 111, 113
Example for current/voltage values

<table>
<thead>
<tr>
<th>Input current 11 bit resolution no missing codes</th>
<th>Numerical value (input current = 16 mA x numerical value / 4096) + 4 mA</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 19.992 mA</td>
<td>4094</td>
</tr>
<tr>
<td>12.000 mA</td>
<td>2048</td>
</tr>
<tr>
<td>..</td>
<td>2 (smallest resolution)</td>
</tr>
<tr>
<td>4.0078 mA</td>
<td></td>
</tr>
<tr>
<td>&lt; 4.000 mA</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input voltage 12 bit resolution no missing codes</th>
<th>Numerical value (input voltage = 10 V x numerical value / 4096)</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 9.9975 V</td>
<td>4095</td>
</tr>
<tr>
<td>5.000 V</td>
<td>2048</td>
</tr>
<tr>
<td>..</td>
<td>1 (smallest resolution)</td>
</tr>
<tr>
<td>0.00244 V</td>
<td>0</td>
</tr>
<tr>
<td>&lt; 0.000 V</td>
<td>0</td>
</tr>
</tbody>
</table>
Example for current/voltage values

<table>
<thead>
<tr>
<th>Output current</th>
<th>Numerical value (numerical value = 4096 x (output current-4 mA)/16 mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 19.996 mA</td>
<td>4095</td>
</tr>
<tr>
<td>12.000 mA</td>
<td>2048</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>4.0039 mA</td>
<td>1 (smallest resolution)</td>
</tr>
<tr>
<td>&lt; 4.000 mA</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Output voltage</th>
<th>Numerical value (numerical value = 4096 x (output voltage)/10 V)</th>
</tr>
</thead>
<tbody>
<tr>
<td>≥ 9.9975 V</td>
<td>4095</td>
</tr>
<tr>
<td>5.000 V</td>
<td>2048</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>0.00244 V</td>
<td>1 (smallest resolution)</td>
</tr>
<tr>
<td>&lt; 0.000 V</td>
<td>0</td>
</tr>
</tbody>
</table>
When this module is used, you receive diagnosis information about the analogue I/O modules of your valve terminal.

The module provides 6 diagnosis functions. The diagnosis result, depending on the diagnosis function used, is prepared in special function units FU33 to FU35. By using appropriate programming techniques, you can either receive collective information or information about individual channels.

<table>
<thead>
<tr>
<th>Input format</th>
<th>CFM 63</th>
</tr>
</thead>
<tbody>
<tr>
<td>THEN</td>
<td>WITH &lt;P1&gt;</td>
</tr>
<tr>
<td></td>
<td>WITH &lt;P2&gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>channel number</td>
</tr>
<tr>
<td></td>
<td>(0...11) for output channels</td>
</tr>
<tr>
<td></td>
<td>(0...35) for input channels</td>
</tr>
<tr>
<td></td>
<td>or -1 for all available channels</td>
</tr>
<tr>
<td>P2</td>
<td>diagnosis function (0...5)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Return parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>P1 (FU32)</td>
<td>(-1) processing successful</td>
</tr>
<tr>
<td></td>
<td>(0) processing faulty</td>
</tr>
<tr>
<td>P2 (FU33)</td>
<td>diagnosis result</td>
</tr>
<tr>
<td></td>
<td>or error number</td>
</tr>
<tr>
<td></td>
<td>(100, 101, 102, 112), if P1 = 0</td>
</tr>
</tbody>
</table>

With diagnosis function 2, 4, 5

| P3 (FU34) | diagnosis result |
| P4 (FU35) | diagnosis result |
The diagnosis result described in the following section only occurs if the function module CFM 63 reports successful processing (FU32 = -1).

**Diagnosis function 0**

Overload/short circuit (s.c.) in analogue voltage outputs; representation by channel

**Diagnosis result in FU33 (FU32 = -1)**

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
<td>D14</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value O0...O11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Voltage output O... not overloaded/not short-circuited</td>
</tr>
<tr>
<td>1</td>
<td>Voltage output O... overloaded, output value cannot be issued</td>
</tr>
</tbody>
</table>

Data bits from channels which are not available and from channels which have not been selected supply the value "logic 0".

Corresponding data bits greater than D11 also always return the logic value "0". These are not relevant because no more than a maximum of 12 analogue output channels can be available at any one time (O0...O11).
Diagnosis function 1

Overloading / short circuit (s.c.) / undervoltage of DC 24 V actuator supply voltage for available analogue I/O modules; representation by modules.

Diagnosis result in FU 33 (FU 32 = -1)

<table>
<thead>
<tr>
<th>Module numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td>V0 V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value V0...V11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>24 V actuator power supply of module V... not overloaded, not short-circuited, no undervoltage</td>
</tr>
<tr>
<td>1</td>
<td>Overloading, short circuit (s.c.) or undervoltage in DC 24 V actuator supply voltage of module V...</td>
</tr>
</tbody>
</table>

Data bits from unavailable modules and unselected modules supply the logic value "0".

Corresponding data bits greater than D11 also always return the logic value "0". They are not relevant, since a maximum of 12 analogue I/O modules can be available (V0...V11).
Diagnosis function 2

Wire break diagnosis of analogue current inputs, input current < 2 mA; representation by channel

Diagnosis result in FU33...FU35 (FU32 =-1)

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15, D14, D13, D12, D11, D10, D9, D8, D7, D6, D5, D4, D3, D2, D1, D0</td>
<td></td>
</tr>
</tbody>
</table>

FU34:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15, D14, D13, D12, D11, D10, D9, D8, D7, D6, D5, D4, D3, D2, D1, D0</td>
<td></td>
</tr>
</tbody>
</table>

FU35:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>000000000000 I 3 5 I 3 4 I 3 3 I 3 2</td>
<td>Log. value I0...I35</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0".

Corresponding channel numbers greater than I35 also always return the logic value "0". These are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
Diagnosis function 3

Wire break diagnosis of analogue current outputs:
Idling / excessive burden resistance; representation by channel

**PLEASE NOTE**
Idling is not recognised until after one set of analogue values has been issued.

Diagnosis result in FU33 (FU32 = -1)

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers (FU33)</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td></td>
</tr>
<tr>
<td>0 0 0 0 O11 O10 O9 O8 O7 O6 O5 O4 O3 O2 O1 O0</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log. value O0...O11</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No idling</td>
</tr>
<tr>
<td>1</td>
<td>Idling of current output; output value cannot be issued</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0".

Corresponding data bits greater than D11 also always return the logic value "0". They are not relevant because no more than a maximum of 12 analogue output channels can be available (O0...O11).
Diagnosis function 4

Determining the analogue voltage input; representation by channel. This diagnosis function makes it possible to check the valve terminal configuration with regard to the fitted voltage inputs.

**Diagnosis result in FU33...FU35 (FU32 = -1)**

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0</td>
<td></td>
</tr>
<tr>
<td>I15 I14 I13 I12 I11 I10 I9 I8 I7 I6 I5 I4 I3 I2 I1 I0</td>
<td></td>
</tr>
</tbody>
</table>

FU34:

| D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0 |
| I31 I30 I29 I28 I27 I26 I25 I24 I23 I22 I21 I20 I19 I18 I17 I16 |

FU35:

| D15 D14 D13 D12 D11 D10 D9 D8 D7 D6 D5 D4 D3 D2 D1 D0 |
| 0 0 0 0 0 0 0 0 0 0 0 0 0 0 I35 I34 I33 I32 |

| Log. value | Description |
| I0...I35 | |
| 0 | No voltage input |
| 1 | Voltage input |

Data bits from unavailable channels and from unselected channels supply the logic value "0". Corresponding channel numbers greater than I35 also always return the logic value "0". They are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
Diagnosis function 5

Determining the analogue current inputs; representation by channel. This diagnosis function makes it possible to test the valve terminal configuration with regard to the fitted current inputs.

**Diagnosis result in FU33...FU35 (FU32 =-1)**

FU33:

<table>
<thead>
<tr>
<th>Channel numbers</th>
<th>Data bit numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>D15</td>
<td>D14</td>
</tr>
<tr>
<td>I15</td>
<td>I14</td>
</tr>
</tbody>
</table>

FU34:

| D15  | D14  | D13  | D12  | D11  | D10  | D9   | D8   | D7   | D6   | D5   | D4   | D3   | D2   | D1   | D0   |
| I31  | I30  | I29  | I28  | I27  | I26  | I25  | I24  | I23  | I22  | I21  | I20  | I19  | I18  | I17  | I16  |

FU35:

| D15  | D14  | D13  | D12  | D11  | D10  | D9   | D8   | D7   | D6   | D5   | D4   | D3   | D2   | D1   | D0   |
| 0    | 0    | 0    | 0    | 0    | 0    | 0    | 0    | 0    | 0    | 0    | 0    | I35  | I34  | I33  | I32  |

<table>
<thead>
<tr>
<th>Log. value I0...I35</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No current input</td>
</tr>
<tr>
<td>1</td>
<td>Current input</td>
</tr>
</tbody>
</table>

Data bits from unavailable channels and from unselected channels supply the logic value "0". Corresponding channel numbers greater than I35 also always return the logic value "0". They are not relevant, since no more than a maximum of 36 analogue input channels can be available (I0...I35).
Number and contents of parameters depend on the function module.

Return parameters
number and contents of parameters depend on function module.
B.3 SUMMARY OF THE ERROR MESSAGES
Contents

B.3 SUMMARY OF THE ERROR MESSAGES .............. B-77
B.3 SUMMARY OF THE ERROR MESSAGES

The red error display (ERROR) lights up as soon as the error status in error bit F is no longer equal to 0. The following error numbers are entered in the error word (errors when switching on, during initialisation and during run time):

<table>
<thead>
<tr>
<th>Error number</th>
<th>Explanation</th>
<th>Cause, possible remedies, effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>02 02</td>
<td>Hardware errors in node or I/O module</td>
<td>Defective hardware in node or I/O module. Control unit starts up.</td>
</tr>
<tr>
<td>04 04</td>
<td>Station (Slave) failed</td>
<td>Check...:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Field bus for defects or interruptions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Power supply of slave</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Otherwise...: servicing required</td>
</tr>
<tr>
<td>07 07</td>
<td>Hardware error in P module</td>
<td>Defective hardware in P module; Control unit starts up.</td>
</tr>
<tr>
<td>08 08</td>
<td>EEPROM defective or missing</td>
<td>Number of operating voltage interruptions reached or user program invalid. Repair required. Control unit moves into stop status.</td>
</tr>
<tr>
<td>09 09</td>
<td>User memory defective</td>
<td>Defective hardware, servicing required.</td>
</tr>
<tr>
<td>A 10</td>
<td>User memory initialised</td>
<td>Error message does not occur, when boot mode set to EEPROM.</td>
</tr>
<tr>
<td>B 11</td>
<td>Operating system memory defective</td>
<td>Defective hardware, servicing required.</td>
</tr>
<tr>
<td>C 12</td>
<td>Collective error AS-i bus</td>
<td>Individual error on an AS-i slave; refer to description of AS-i slave.</td>
</tr>
<tr>
<td>D 13</td>
<td>Voltage supply AS-i defective</td>
<td>Voltage supply of AS-i power pack or of yellow flat cable impaired (power failure).</td>
</tr>
<tr>
<td>E 14</td>
<td>AS-i master not available</td>
<td>CFM call made to AS-i master, although no AS-i master was available.</td>
</tr>
<tr>
<td>F 15</td>
<td>AS-i slave not available</td>
<td>Check installation on AS-i bus system.</td>
</tr>
<tr>
<td>13 19</td>
<td>CP module not available</td>
<td>CFM call made to CP modules, although no CP modules were available.</td>
</tr>
<tr>
<td>14 20</td>
<td>CP component not available</td>
<td>Check installation on CP system.</td>
</tr>
<tr>
<td>15 21</td>
<td>CP component failed</td>
<td>CP cable interrupted or CP module defective.</td>
</tr>
<tr>
<td>17 23</td>
<td>Memory full</td>
<td>Reduce size of project / program to suit available storage capacity. Then repeat the loading process. Control unit moves into stop status.</td>
</tr>
</tbody>
</table>

---

Fig. B/11a: Error messages of the operating system
## Error messages

<table>
<thead>
<tr>
<th>Error number</th>
<th>Explanation</th>
<th>Cause, possible remedies, effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>19 25</td>
<td>Watchdog overflow</td>
<td>Incorrect function (deadlock) in program sequence (software reset carried out)</td>
</tr>
<tr>
<td>30 48</td>
<td>Type of module not permitted</td>
<td>Only fit approved I/O or P modules. Control unit starts up.</td>
</tr>
<tr>
<td>31 49</td>
<td>More than 12 I/O modules fitted.</td>
<td>Reduce number of I/O modules. Control unit starts up.</td>
</tr>
<tr>
<td>33 51</td>
<td>I/O address space exceeded</td>
<td>Reduce number of I/O modules or valves. Control unit starts up.</td>
</tr>
<tr>
<td>35 53</td>
<td>CP module not first module</td>
<td>Refer to CP Handbook</td>
</tr>
<tr>
<td>64 100</td>
<td>Number of parameters incorrect</td>
<td></td>
</tr>
<tr>
<td>65 101</td>
<td>Error in parameter 1</td>
<td>Check transfer parameters. Program processing is continued.</td>
</tr>
<tr>
<td>66 102</td>
<td>Error in parameter 2</td>
<td></td>
</tr>
<tr>
<td>67 103</td>
<td>Error in parameter 3</td>
<td></td>
</tr>
<tr>
<td>68 104</td>
<td>Error in parameter 4</td>
<td></td>
</tr>
<tr>
<td>69 105</td>
<td>Error in other parameters</td>
<td></td>
</tr>
<tr>
<td>6A 106</td>
<td>Error in module call</td>
<td>Module with specified module number is not available. Program processing is continued.</td>
</tr>
<tr>
<td>6B 107</td>
<td>Error in program call</td>
<td>Program with specified program number is not available. Program processing is continued.</td>
</tr>
<tr>
<td>6C 108</td>
<td>Division faulty</td>
<td>Division by zero or Divisor ≤ -32768. Program processing is continued.</td>
</tr>
</tbody>
</table>

...continued next page...
<table>
<thead>
<tr>
<th>Error number</th>
<th>Explanation</th>
<th>Cause, possible remedies, effects</th>
</tr>
</thead>
<tbody>
<tr>
<td>6D 109</td>
<td>Multiplication faulty</td>
<td>Range of values during multiplication exceeded ($\pm 32767$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Program processing is continued.</td>
</tr>
<tr>
<td>6E 110</td>
<td>Access to program-controlled parameter not possible</td>
<td>Processing of function module was faulty or addressed slave was incorrect type.</td>
</tr>
<tr>
<td>6F 111</td>
<td>Timeout</td>
<td>Timeout when processing a program-controlled data transfer (e.g. slave defective).</td>
</tr>
<tr>
<td>70 112</td>
<td>I/O module error</td>
<td>Addressed module not available or faulty.</td>
</tr>
<tr>
<td>71 113</td>
<td>Incorrect operating mode selected</td>
<td>The called function module is not supported in the set operating mode. If necessary, alter the operating mode.</td>
</tr>
<tr>
<td>72 114</td>
<td>Field bus configuration error</td>
<td>Number of max. permissible field bus I/Os exceeded.</td>
</tr>
<tr>
<td>73 115</td>
<td>AS-i configuration error</td>
<td>AS-i master does not respond (time out).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Switch valve terminal on/off.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check assembly of AS-i master module.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(VG-strip, bent contact).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Otherwise: Hardware error, servicing required.</td>
</tr>
</tbody>
</table>

*Fig. B/11c: Error messages of the operating system*
Programmable valve terminals
with
control block SF 3

APPENDIX C:
Command interpreter
Contents

C. COMMAND INTERPRETER .......................... C-1
   Summary ........................................... C-1
   Connection to a dialogue device ............. C-3
   Calling the command interpreter ............. C-4
   Exiting the command interpreter ............ C-5
   Command structure .............................. C-5
   Command recognition ........................... C-6
   Definition of parameters ....................... C-7
   Loading format ................................. C-8

Command description .............................. C-9

   HEXDUMP ........................................ C-9
   Display memory contents (H-command) .......... C-9
   Display memory .................................. C-9

   DISPLAY .......................................... C-10
   Display operands (D-command) ................ C-10
      One-bit operand ............................... C-10
      Multi-bit operands ............................ C-11
      Special operands .............................. C-12
         A) Display format .......................... C-12
         B) Handshake format ....................... C-12
         C) Program ................................. C-12
         D) Module .................................. C-15
         E) Library .................................. C-16
         F) Vacant user memory ..................... C-16
         G) EEPROM ................................ C-17
         H) Runmode ................................ C-17
         I) DKS command
            (refer to MODIFY) ....................... C-17
CHECKSUM .......................... C-18
Create check sum (C command) ...... C-18
Check sum and check byte .......... C-18

MODIFY ............................. C-19
Modify ............................. C-19
Modifying operands (M command) ... C-19
One-bit operands .................. C-20
Multi-bit operands ............... C-21
Special operands ................ C-22
  A) Display format .............. C-22
  B) Handshake format .......... C-22
  C) Run mode ................ C-22
  D) MKF system configuration of field bus SETPOINT.. .... C-22

ALLOCATION .......................... C-23
Memory management .............. C-23
Memory allocation (Z command) .... C-23
  Create a program/module ...... C-24
  Create a library ............ C-24

INITIALIZE .......................... C-25
Delete user memory (Y command) ... C-25
  A) Delete all files .......... C-25

NULLIFY ............................. C-25
  B) Delete single programs ... C-25
  C) Delete individual modules .... C-26
  D) Delete individual libraries .. C-26
Delete operands .................. C-27
  A) Delete all flags .......... C-27
  B) Delete all registers ....... C-27
  C) Delete all timers .......... C-27
  D) Delete all counters ....... C-27
PROM .................................................. C-28
Data backup ................................. C-28
EEPROM data backup (P command) .... C-28
E) Select RAM/EEPROM .......... C-28

WRITE .............................................. C-29
Read out data (W command) ........ C-29

LOAD ............................................... C-30
Read in (load) data ................. C-30

RUN ................................................. C-31
Program start (R command) .... C-31

STOP ............................................... C-31
Program stop (S command) .... C-31

FUNCTION MODULE ..................... C-32
Call function modules (F command) C-32
C. COMMAND INTERPRETER

Summary

The command interpreter is part of the operating system of a programmable valve terminal. It permits easy operation of a programmable valve terminal by means of a PC and provides an interface to the FST 200 in online-operation (comfortable PC function).

Fig. C/1: Summary of command interpreter
Operation options with the terminal:

Program
• Start.
• Stop.
• Load.
• Delete (single or all).
• Modify.
• Create check sum.

Function unit
• Modify.
• Display.

Memory areas
• Display (individual areas or total programming area).
• Display directory.
• Display all data stored in memory.

PLEASE NOTE
In the following section, the terms "command" and "instruction" are used as synonyms.
Connection to a dialogue device

To operate the command interpreter, the programmable valve terminal has to be connected to an appropriate dialogue device.

You can select between:
• PC with RS 232 interface.
• terminal with RS 232 interface.

The diagnosis interface acts as the interface for a programmable valve terminal. This is an RS 232 interface (V.24) with electrical insulation provided by an optocoupler (specification: refer to Chapter 2, Technical Data).

Proceed as follows when connecting this unit:
• Switch off the operating voltage of the dialogue device and programmable valve terminal.
• Adjust RS 232 interface parameters (at hardware level).
• Connect up the interface.
• Switch on the operating voltages in the following sequence:
  1. Programmable valve terminal.
  2. Dialogue device.
• If necessary, adjust RS 232 interface parameters at software level.
The utility program "Online operation" is a fixed component of the FST 200. It is used to display or modify the system status of current processes. It also serves to define the system configuration of the SF 3. When called, the command interpreter of the SF 3 becomes active.

Calling the command interpreter

WARNING

- The command interpreter contains instructions which reorganise or delete parts of the memory. Data in the memory are then lost.
- Use an instruction only if you know the effects it will have.

Call with FST 200

The command interpreter is convenient to operate using the function key.
- To do this, select "Online operation" in the Utilities program menu.

In addition, a terminal can be simulated inside the FST 200.
- To do this, select F3 terminal mode in online operation.
Call with terminal

The second option is to access data via a terminal. For this, you require the following commands.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL T</td>
<td>SF 3 V1.1 &gt;_</td>
</tr>
</tbody>
</table>

Recommendation:
The command interpreter can be restarted at any time by pressing CTRL+T.

Exiting the command interpreter

You can exit the command interpreter in two ways, either with the control unit or with the dialogue device:

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>X &lt;CR&gt;</td>
<td>No signal from command interpreter</td>
</tr>
</tbody>
</table>

Command structure

Every instruction has a defined entry format. This includes:

- one letter (for command recognition).
- one parameter (letter or number in accordance with parameter definition).
- one memory address (not always required).

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;command letter&gt;, [&lt;parameter&gt;], [&lt;memory address&gt;]</td>
<td>Dependent on command</td>
</tr>
</tbody>
</table>
Recommendation:
- Entry is possible in upper or lower case letters.
- Conclude entries by pressing the <CR>-key.
- Corrections are possible with DEL, Backspace and CRTL H.

Command recognition

The following letters represent commands:

<table>
<thead>
<tr>
<th>Command letter and command</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>H = HEX-DUMP</td>
<td>(= Hexadec. memory excerpt)</td>
</tr>
<tr>
<td>D = DISPLAY</td>
<td>(= Display of operands)</td>
</tr>
<tr>
<td>C = CHECKSUM</td>
<td>(= Check sum)</td>
</tr>
<tr>
<td>M = MODIFY</td>
<td>(= Modify operands)</td>
</tr>
<tr>
<td>Z = ALLOCATION</td>
<td>(= Assignment of memory location)</td>
</tr>
<tr>
<td>N = NULLIFY</td>
<td>(= Delete programs and operands)</td>
</tr>
<tr>
<td>Y = INITIALISE</td>
<td>(= Initialise the directory)</td>
</tr>
<tr>
<td>P = PROM</td>
<td>(= EEPROM)</td>
</tr>
<tr>
<td>W = WRITE</td>
<td>(= Data output)</td>
</tr>
<tr>
<td>L = LOAD</td>
<td>(= Data entry)</td>
</tr>
<tr>
<td>R = RUN</td>
<td>(= Start)</td>
</tr>
<tr>
<td>S = STOP</td>
<td>(= Stop)</td>
</tr>
<tr>
<td>F = FUNCTION</td>
<td>(= Function modules)</td>
</tr>
<tr>
<td>X = EXIT</td>
<td>(= Exit)</td>
</tr>
</tbody>
</table>

Display of memory contents/operands.
Create check sum.
Modify.
Memory management.
EEPROM program, Adjust boot mode.
Data backup.
Start program.
Stop program.
Calling function modules.
Exiting the command interpreter.
Definition of parameters

Depending on which operand, file or memory area a command is to be applied, different parameters have to be entered.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Total memory contents without vacant user memory or system status.</td>
</tr>
<tr>
<td>S</td>
<td>Vacant available user memory or system interrogation (dependent on previous command).</td>
</tr>
<tr>
<td>D</td>
<td>Directory (program directory) or the display format (dependent on previous command).</td>
</tr>
<tr>
<td>K</td>
<td>Configuration</td>
</tr>
<tr>
<td>L&lt;xx&gt;</td>
<td>Library (=library)</td>
</tr>
<tr>
<td>H</td>
<td>Handshake format</td>
</tr>
<tr>
<td>P&lt;xx&gt;</td>
<td>Program no. x</td>
</tr>
<tr>
<td>B&lt;xx&gt;</td>
<td>Program module no. x</td>
</tr>
<tr>
<td>E&lt;y&gt;.&lt;xx&gt;</td>
<td>Input no. y.x (*)</td>
</tr>
<tr>
<td>EW&lt;y&gt;</td>
<td>Input word no. y *)</td>
</tr>
<tr>
<td>A&lt;y&gt;.&lt;xx&gt;</td>
<td>Output no. y.x *)</td>
</tr>
<tr>
<td>AW&lt;y&gt;</td>
<td>Output word y *)</td>
</tr>
<tr>
<td>M&lt;y&gt;.&lt;xx&gt;</td>
<td>Flag no. y.x</td>
</tr>
<tr>
<td>MW&lt;y&gt;</td>
<td>Flag word no. y</td>
</tr>
<tr>
<td>Z&lt;y&gt;</td>
<td>Counter no. y</td>
</tr>
<tr>
<td>ZW&lt;y&gt;</td>
<td>Counter word no. y</td>
</tr>
<tr>
<td>ZV&lt;y&gt;</td>
<td>Counter preselect no. y</td>
</tr>
<tr>
<td>T&lt;y&gt;</td>
<td>Timer no. y</td>
</tr>
<tr>
<td>TW&lt;y&gt;</td>
<td>Timer word no. y</td>
</tr>
<tr>
<td>TV&lt;y&gt;</td>
<td>Timer preselection no. y</td>
</tr>
<tr>
<td>R&lt;z&gt;</td>
<td>Register no. z</td>
</tr>
<tr>
<td>O&lt;u&gt;</td>
<td>Special operand no. u</td>
</tr>
</tbody>
</table>

- u = (0...4095)
- x = (0,..., 15)
- y = (0,..., 31)
- z = (0,..., 127)

*) dependent on size of valve terminal
Loading format

Data are read into or out of "INTELHEX format" by the command interpreter using the diagnostic port. This data format possesses a special feature: every line contains its memory address as well as its check sum. This increases reliability of transmission by means of serial data transmission.

In the "INTELHEX format", a loaded file is displayed in the following manner:

<Length> <AA> <Type> <Data><Check byte>

Length: number (two-digit in hex form) of data bytes per line.
AA: initial address of first data byte (four-digit).
Type: 00 = the line contains information
01 = the line is the last transmission line and does not contain any information. This record is an "end record".
Data: data-bytes (always two-digit)
Check byte: check byte for all bytes in the line.

Example:

<table>
<thead>
<tr>
<th>Relative address</th>
<th>Data-bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>: 10 0230 00</td>
<td>0640170017C0188019401A001AC01B801</td>
</tr>
</tbody>
</table>

Line contains 10-hex=16-dez data bytes
Line contains information
Check byte
Command description

In the next section, you will find an explanation of individual commands and, where applicable, suitable examples.

Display memory contents (H-command)

To display the memory contents, you can use the following commands:
- HEXDUMP instruction: Display of memory.
- DISPLAY instruction: Display of operands.

Display memory

The H-command provides the option of printing memory areas on the display in hexadecimal form.

You can cancel the H-command by pressing the <CR>-key.

The following displays are possible:
- total memory area.
- directory.
- single files.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>HD</td>
<td>Display the directory</td>
</tr>
<tr>
<td>HP (0, ..., 15)</td>
<td>Display programs no. 0 up to no. 15</td>
</tr>
<tr>
<td>HB (0, ..., 15)</td>
<td>Display program modules no. 0 up to no. 15</td>
</tr>
<tr>
<td>HL (0, ..., 15)</td>
<td>Display SETPOINT list</td>
</tr>
<tr>
<td>HI (0, ..., 15)</td>
<td>Display ACTUAL list</td>
</tr>
<tr>
<td>HK</td>
<td>Display system configuration</td>
</tr>
</tbody>
</table>
**Display operands (D-command)**

With the help of the DISPLAY instruction, you can display the modes and contents of operands as well as the current program status. The answer from the command interpreter always appears on the entry line.

One-bit operand

Display status 0 signal or 1 signal of the selected operands.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timers D T $&lt;y&gt;$</td>
<td>$DT&lt;y&gt; = {0/1}$</td>
</tr>
<tr>
<td>Counters D Z $&lt;y&gt;$</td>
<td>$DZ&lt;y&gt; = {0/1}$</td>
</tr>
<tr>
<td>Inputs D E $&lt;y&gt;.&lt;x&gt;$</td>
<td>$DE&lt;y.x&gt; = {0/1}$</td>
</tr>
<tr>
<td>Outputs D A $&lt;y&gt;.&lt;x&gt;$</td>
<td>$DA&lt;y.x&gt; = {0/1}$</td>
</tr>
<tr>
<td>Flags M $M&lt;y&gt;.&lt;y&gt;$</td>
<td>$DM&lt;y.x&gt; = {0/1}$</td>
</tr>
</tbody>
</table>

**Example 1**

Display status of one-bit operand, output 0.6
DA0.6=1
>_<

**Example 2**

Display status of one-bit operand, input 1.5
DE1.5=0
>_<
Multi-bit operands

The contents of multibit function units can be interrogated as follows:

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input words: D E W &lt;y&gt;</td>
<td>DEW&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Output words: D A W &lt;y&gt;</td>
<td>DAW&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Flag words: D M W &lt;y&gt;</td>
<td>DMW&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Timer words: D T W &lt;y&gt;</td>
<td>DTW&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Counter words: D Z W &lt;y&gt;</td>
<td>DZW&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Timer preselect: D T V &lt;y&gt;</td>
<td>DTV&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Counter preselect: D Z V &lt;y&gt;</td>
<td>DZV&lt;y&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Error word: D F</td>
<td>DF = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Register: D R &lt;z&gt;</td>
<td>DR&lt;z&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
<tr>
<td>Special operand: DO &lt;u&gt;</td>
<td>DO&lt;u&gt; = ($HEXADEC./SIGNED DEC./DEC./BIN)</td>
</tr>
</tbody>
</table>

x = {0,..., 15},  u = {0,..., 4095},  y = {0,..., 31},  z = {0,..., 127}
Special operands

A) Display format

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>D D</td>
<td>&gt;DD=(S/D/H/B)</td>
</tr>
</tbody>
</table>

S = signed decimal (default),  D = decimal,
H = hexadecimal,  B = binary

B) Handshake format

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>D H</td>
<td>&gt;DH=(0/1)</td>
</tr>
</tbody>
</table>

0 = without handshake, 1 = with handshake (default)

C) Program

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>D P &lt;x&gt;</td>
<td>&gt;DPx = &lt;program type&gt;,</td>
</tr>
<tr>
<td></td>
<td>&lt;program length&gt;,</td>
</tr>
<tr>
<td></td>
<td>&lt;program status&gt;,</td>
</tr>
<tr>
<td></td>
<td>[&lt;current program step no.&gt;],</td>
</tr>
<tr>
<td></td>
<td>[&lt;module number&gt;],</td>
</tr>
<tr>
<td></td>
<td>[current module step no.&gt;]</td>
</tr>
</tbody>
</table>

x = {0,..., 15}

* These signals are explained in more detail at a later point
Program type

The program type is indicated in code form in accordance with the following bit map image (in decimal or hexadecimal form):

<table>
<thead>
<tr>
<th>Bit</th>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>File name:</td>
<td>000 = program</td>
<td>001 = module</td>
<td>010 = library</td>
<td>011 = external file</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Storage type:</td>
<td>0 = relocatable</td>
<td>1 = absolute</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Language:</td>
<td>000 = LDR (ladder diagram)</td>
<td>001 = STL (statement list)</td>
<td>010 = assembler</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Check sum bit:</td>
<td>0 OR 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. C/2: Coding of program type
Example:

Relocatable ("variable") STL module with check sum bit.

<table>
<thead>
<tr>
<th>Coding:</th>
<th>1 0 0 1 0 0 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representation: &gt;DP3=145,... decimal</td>
<td></td>
</tr>
<tr>
<td>&gt;DP3=$0091,... hexadecimal</td>
<td></td>
</tr>
</tbody>
</table>

Fig. C/3: Example of program type

Program length
The program length indicates the number of data bytes in a program.

Program status
Program status = 0 means "program is inactive"
Program status = 1 means "program is active"

Program step-no.
The current program step number (0 to 255) is only displayed if the program is active (status = 1):
LDR (ladder diagram) progr.: no. 0
STL (statement list) progr.: no. 0...255

Module-no. and module step no.
If an active program processes a module, the module no. (0 to 15) and the module step no. (0..255) are displayed.
D) Module

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>D B &lt;x&gt;</td>
<td>&gt;DB&lt;x&gt; = &lt;module type&gt;,&lt;module length&gt;</td>
</tr>
<tr>
<td>x = {0,..., 15}</td>
<td></td>
</tr>
</tbody>
</table>

Module type
See program type

Module length
The module length indicates the number of data bytes in a module.
### E) Library

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>DL{0...15}</td>
<td>DL{0...15} = &lt;Nr.&gt;&lt;Library length&gt;&lt;Name&gt;</td>
</tr>
</tbody>
</table>

**Library length**

The library length indicates the number of data bytes in a library.

### F) Vacant user memory

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>DS</td>
<td>&gt;DS = &lt;available user memory&gt;</td>
</tr>
</tbody>
</table>

**Available user memory**

With the DS command, the size of the vacant user memory can be displayed in three optional ways: in decimal, hexadecimal or signed decimal.
G) EEPROM

The number of completed program cycles for programs, output of current boot mode.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC</td>
<td>( \text{DC} = \langle \text{Number of programming cycles} \rangle, \langle \text{Boot mode} \rangle )</td>
</tr>
</tbody>
</table>

Number of programming cycles = \(0...999\)
Boot mode = \{80, 82\}
80 = EEPROM
82 = RAM

H) Runmode

<table>
<thead>
<tr>
<th>Entry format</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>DKR</td>
<td>{0/1}</td>
</tr>
</tbody>
</table>

0 = automatic start off  1 = automatic start on

I) DKS command (refer to MODIFY)
Create check sum (C command)

Using the command interpreter, you have the option of checking any area in the user memory.

Check sum and check byte

With the C command, the check sum and the check byte of a memory area are formed and are shown on the display.

The check sum is shown as six digits and the check byte is shown as two digits in hexadecimal form.

Similarly, the appropriate check sum can be formed for the entire memory area, for the directory alone or for each file individually.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>Check sum for all programs, program modules, libraries</td>
</tr>
<tr>
<td>CD</td>
<td>Check sum for the directory</td>
</tr>
<tr>
<td>CP (0,...,15)</td>
<td>Check sum for program no. 0 to 15</td>
</tr>
<tr>
<td>CB (0,...,15)</td>
<td>Check sum for module no. 0 to 15</td>
</tr>
<tr>
<td>CL (0,...,15)</td>
<td>Check sum for library</td>
</tr>
</tbody>
</table>

Example 1
CP2 = 001500:00

Example 2
CL2 = 02F300:00

The check byte for the entire memory area, the directory and each individual file must always be 00. If it is not equal to 00, the interrogated storage area is faulty.
Modify

In the command interpreter, you have the option of modifying operands.

Modifying operands (M command)

With the M command, you can modify the contents and/or the status of the operands. You then have the choice of displaying the operands or not.

- In order to modify an operand directly without prior display, enter the desired value after the command and complete the operation by pressing the <CR> key.

Example:
>MAW1=255

- If you wish to view the contents and/or the status of the operands beforehand, press the <CR> key after entering the command. The command interpreter will supply current values. After the colon, you can enter the new value, then press the <CR> key.

Example:
>MAW1=255:126

The values can be entered in decimal, hexadecimal and signed decimal notation (refer to display format).
One-bit operands

To modify the status of operands, enter the following:

<table>
<thead>
<tr>
<th>Entry format:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Outputs: M A &lt;y&gt;.&lt;x&gt;</td>
<td></td>
</tr>
<tr>
<td>Flag status: M M &lt;y&gt;.&lt;y&gt;</td>
<td></td>
</tr>
<tr>
<td>Timer status: M T &lt;y&gt;</td>
<td></td>
</tr>
<tr>
<td>Counter status: M Z &lt;y&gt;</td>
<td></td>
</tr>
</tbody>
</table>

x = {0,..., 15}, y = {0,..., 31}

Example

>MA1.6 = 1 : 0

Before the alteration

After the alteration
Multi-bit operands

To modify the contents of multi-bit operands, the following entries are required:

<table>
<thead>
<tr>
<th>Entry format:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output word: M A W &lt;y&gt;</td>
</tr>
<tr>
<td>Flag word: M M W &lt;y&gt;</td>
</tr>
<tr>
<td>Timer word: M T W &lt;y&gt;</td>
</tr>
<tr>
<td>Timer preselect: M T V &lt;y&gt;</td>
</tr>
<tr>
<td>Counter word: M Z W &lt;y&gt;</td>
</tr>
<tr>
<td>Counter preselect: M Z V &lt;y&gt;</td>
</tr>
<tr>
<td>Register: M R &lt;z&gt;</td>
</tr>
<tr>
<td>Error: M F</td>
</tr>
<tr>
<td>Special operand: M O &lt;u&gt;</td>
</tr>
</tbody>
</table>

\[ x = \{0,\ldots, 15\}, \quad y = \{0,\ldots, 31\}, \quad z = \{0,\ldots, 127\}, \quad u = \{0\ldots 4095\} \]

Example

> MZW0=9662:xxxxx

Current counter value Counter value to be selected
Special operands

A) Display format

<table>
<thead>
<tr>
<th>Entry format</th>
</tr>
</thead>
<tbody>
<tr>
<td>M D = (D/S/H/B)</td>
</tr>
<tr>
<td>D = decimal, S = signed decimal, H = hexadecimal</td>
</tr>
<tr>
<td>B = binary</td>
</tr>
</tbody>
</table>

B) Handshake format

<table>
<thead>
<tr>
<th>Entry format</th>
</tr>
</thead>
<tbody>
<tr>
<td>M H = (0/1)</td>
</tr>
<tr>
<td>0 = without handshake, 1 = with handshake</td>
</tr>
</tbody>
</table>

C) Run mode

<table>
<thead>
<tr>
<th>Entry format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MkR = {0,1}</td>
</tr>
<tr>
<td>0 = auto mode off, 1 = auto mode on</td>
</tr>
</tbody>
</table>

DKS system configuration field bus ACTUAL. Display field bus settings currently in use.

DKF system configuration of field bus SETPOINT. Display field bus settings which are used with the next system start (Power On).

D) MKF system configuration of field bus SETPOINT.

No field bus: MkS = K
Master: MkS = M, FB-baud = bus
Slave: MkS = S, FB-baud, bus, address, number I, number O
Mode: K (0), M (1), S (2)
FB-baud: 0 = 375 kB, 1 = 187.5 kB, 2 = 62.5 kB, 3 = 31.25 kB
Bus: 0 = bus terminator off, 1 = bus terminator on
Address: FB-address for slaves (1...31)
Number-I: Number of field bus input bytes (0...12)
Number-O: Number of field bus output bytes (0...12)
Memory management

As soon as you wish to erase programs or load them into the control unit, you will require instructions which organise the memory.

Memory allocation (Z command)

The Z command must be entered before loading
- programs
- modules
- libraries
(Also refer to "Read in data"). This organises the memory and takes charge of the following functions:
- checks whether there is sufficient vacant memory capacity.
- opens new memory areas for programs, modules and libraries.
- moves existing programs and modules to the last location in the user memory in case a program or module is to be loaded with the same number.
- checks the version number if a library is already available.
Create a program/module

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZP &lt;program number&gt;</td>
<td>$&quot;address between 0000 and 7FFE&quot; OR E9: MEMORY FULL *) &gt;_</td>
</tr>
<tr>
<td>&lt;program length&gt;</td>
<td></td>
</tr>
<tr>
<td>ZB &lt;module number&gt;,</td>
<td>$&quot;address between 0000 and 7FFE&quot; OR E9: MEMORY FULL *) &gt;_</td>
</tr>
<tr>
<td>&lt;module length&gt;</td>
<td></td>
</tr>
</tbody>
</table>

Program and/or module number = \{0...7\}
Program and/or module length = length of file in byte
*) The program which is to be loaded is now too large to fit in the user memory

Create a library

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZL &lt;x&gt;,</td>
<td>$&quot;address between 0000 and 7FFF&quot; OR $&quot;address between 0000 and 7FFE&quot; OR E8: ACCESS ERROR OR E9: MEMORY FULL</td>
<td>Library is no longer available and can be loaded using the LL command. Library is already available, therefore no longer has to be loaded into the control unit. Available library has incorrect version or max. number of libraries is already loaded. Library too large to fit in user memory.</td>
</tr>
<tr>
<td>&lt;library length&gt;</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

x = \{0...15\}
Delete user memory (Y command)

Depending on whether you wish to delete all files (programs, program modules, libraries) at the same time or just one at a time, a different command may have to be used.

⚠️ CAUTION
The Y command deletes the complete user memory (incl. directory). It is then not possible to reconstruct these data.

A) Delete all files

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>&gt;DEL ALL ? [Y/N]:</td>
</tr>
</tbody>
</table>

Entry format

Y <CR> if all are to be deleted
<CR> if the Y command is not to be used

B) Delete single programs

<table>
<thead>
<tr>
<th>Entry format</th>
</tr>
</thead>
<tbody>
<tr>
<td>N P &lt;Program number&gt;</td>
</tr>
</tbody>
</table>

Program number = {0, ..., 15}
C) Delete individual modules

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>N B &lt;Module number&gt;</td>
<td>DEL ALL ? [Y/N]:_</td>
</tr>
<tr>
<td>Module number = {0,..., 15}</td>
<td></td>
</tr>
</tbody>
</table>

D) Delete individual libraries

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>NL&lt;Library type&gt;</td>
<td>DEL ALL ? [Y/N]:_</td>
</tr>
<tr>
<td>Library type = {A/K/F}</td>
<td></td>
</tr>
<tr>
<td>Input format</td>
<td></td>
</tr>
<tr>
<td>Y &lt;CR&gt; if all are to be deleted</td>
<td></td>
</tr>
<tr>
<td>&lt;CR&gt; if the Y command is not to be used</td>
<td></td>
</tr>
</tbody>
</table>
Delete operands

A) Delete all flags

Entry format
N M

B) Delete all registers

Entry format
N R

C) Delete all timers

Entry format
N T

D) Delete all counters

Entry format
N Z
**Data backup**

The command interpreter offers various commands for data backup. This means that data:

- can be copied from RAM into EEPROM (safety backup in the event of a power failure).
- can be read out of the control unit for external storage (redundancy).
- can be read into the control unit (from external backup).

**EEPROM data backup (P command)**

E) Select RAM/EEPROM

There is a choice of three commands for copying the program from RAM to EEPROM. These commands are protected and cannot be accessed until you have entered **CTRL A**.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTRL A</td>
<td>&gt;(^*)Release protected command</td>
</tr>
<tr>
<td>PE copy contents from RAM into EEPROM</td>
<td>&gt;PE = {-1} Processing successful &gt;PE = {0, error number}</td>
</tr>
<tr>
<td>PP EEPROM boot mode</td>
<td>&gt;PP = {-1} Processing successful &gt;PP = {0, error number}</td>
</tr>
<tr>
<td>PR Boot mode RAM</td>
<td>&gt;PR = {-1} Processing successful &gt;PR = {0, error number}</td>
</tr>
</tbody>
</table>

*) The commands remain unprotected, until
- CTRL A is entered again or
- until the command interpreter is exited
**Read out data (W command)**

Using the W command, you can read a specified area of user memory in load format (= INTELHEX format) via the diagnostic port. The address of the data lines read with the W command is relative, i.e. it refers to the initial address of the relevant area of memory.

You can also display the entire memory area, the directory by itself or each file individually.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>WC</td>
<td>Read all programs, program modules and libraries</td>
</tr>
<tr>
<td>WD</td>
<td>Read the directory</td>
</tr>
<tr>
<td>WP {0,..., 15}</td>
<td>Read program nos. 0 to 15</td>
</tr>
<tr>
<td>WB {0,..., 15}</td>
<td>Read module nos. 0 to 15</td>
</tr>
<tr>
<td>WL {0...15}</td>
<td>Read libraries</td>
</tr>
</tbody>
</table>
Read in (load) data

With the load command (read in data), you have the option of loading files (programs, modules, libraries) from the dialogue device to the memory of the programmable valve terminal.

**PLEASE NOTE**
- Before using the L command, you must use the Z command to create the new program in the memory or move it to the end of the valid user memory area.
- The data read into the memory are stored in the memory location indicated by the Z command.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>LC</td>
<td>Load the entire memory area</td>
</tr>
<tr>
<td>LD</td>
<td>Load the directory</td>
</tr>
<tr>
<td>LP {0,..., 15}</td>
<td>Load program nos. 0 to 15</td>
</tr>
<tr>
<td>LB {0,..., 15}</td>
<td>Load module nos. 0 to 15</td>
</tr>
<tr>
<td>LL {0,..., 15}</td>
<td>Load library nos. 0 to 15</td>
</tr>
</tbody>
</table>

Loading is concluded with the end record (otherwise when the operating voltage is switched on/off). You must then enter <DC1> or <CR> (either with or without handshake).

The command interpreter transfers data in INTELHEX format.
Program start (R command)

If an executable program is to be activated in the control unit, use the R command for this purpose.

Entry format

```
R
```

Only the R command can start the first runnable program entered in the directory.

Recommendation:
Allocate program number 0 to the control program.

If a specific program is to be started, this must be entered directly.

Entry format

```
R P <Program number>
```

Program number = \{0,..., 15\}

Program stop (S command)

The S command stops all running programs.

Entry format

```
S
```
Call function modules (F command)

With the F command, function modules are called. After a successful module call, up to six parameters are returned.

<table>
<thead>
<tr>
<th>Entry format</th>
<th>Signal from command interpreter</th>
</tr>
</thead>
<tbody>
<tr>
<td>F &lt;FBN&gt; [.P1 [.P2 [.P3 [.P4 [.P5 [.P6 ]]]]]]</td>
<td></td>
</tr>
</tbody>
</table>

FBN: = functional module number {0..255}
Refer to Appendix B, summary of function modules
Programmable valve terminal
with
control block SF 3

APPENDIX D:
Glossary, index
Contents

D.1 GLOSSARY . . . . . . . . . . . . . . . . . . . . . . . . . . . D-1

D.2 INDEX . . . . . . . . . . . . . . . . . . . . . . . . . . . D-13
D.1 GLOSSARY

Automode

If this mode is selected for a programmable valve terminal, the program with the lowest program number is started, when the operating voltage is switched on. Otherwise, a program has to be started explicitly (e.g. with terminal/command interpreter, PC/FST 200).

If programmable valve terminals with field bus modules are used as independent slaves on a Festo field bus, the automode function should be set for each of these independent slaves. Do however ensure that automatic program starts on the field bus do not trigger any accidental procedures on connected actuator units.

Baud

Unit for the speed of data transmission.

1 baud = 1 bit/s

Baud rate

The baud rate indicates the speed of data transmission. Transmitter and receiver must be set to the same baud rate, since otherwise errors can occur during transmission.
Boot mode

“Booting” refers to the process of starting up/running up programmable devices (PCs, programmable logic controllers, programmable valve terminals). The boot mode defines what happens during a start-up phase. Programmable valve terminals feature the following boot mode options:

- **RAM boot mode**
  When operating voltage is switched on, the system accesses the user program in RAM.

- **EEPROM boot mode**
  When operating voltage is switched on, the entire contents of EEPROM are copied into RAM. After this, the program is processed in RAM.

Bus station/slave

Field bus station/device which can send, transmit or amplify data via the bus, e.g. master station, slave station, repeater, etc.

Command set

This includes all instructions which can be used to program programmable valve terminals in the programming languages STL and LDR. The command set has different peripherals for each of the different languages.
Control block

The control block is the "intelligent part" of a programmable valve terminal and contains the electronics (PLC) which make a standard valve terminal programmable. The control block is designed to comply with protection class IP65. This means that the integrated PLC can be used directly on a machine.

Centrally-controlled

A field bus system is only centrally-controlled if a master manages all information on the field bus and the actions of all inputs and outputs. A centrally-controlled system is deemed to exist whenever valve terminals are used on the Festo field bus.

Diagnosis

Recognition, localisation, classification, display, further evaluation of faults, defects and signals.

Diagnosis offers monitoring functions which run automatically while equipment is in operation.

Diagnosis byte

When a fault occurs, the diagnosis byte provides important information about the status of connected peripheral equipment (e.g. AS-i masters or other field bus stations). Diagnosis bytes are interrogated by the PLC user program via the field bus. Status information obtained in this manner can be evaluated in programs. This means that, for example, errors can be displayed in (normal) text on a text display.
Diagnosis interface
The diagnosis interface is the connection between a programmable valve terminal and the following devices:
- Control unit (text display and control unit ABG/DCU and ABG-2/DCU-2, terminal)
- Programming tool (PC with RS 232 interface)
The diagnosis interface complies with international standard V.24/RS 232.

EEPROM (Electrically Erasable Programmable Read Only Memory)
Read-only memory (ROM), the content of which can be deleted electrically, then reprogrammed with new information. EEPROM units can be programmed up to 100,000 times. With the SF 3 control unit, user programs can be transferred to the EEPROM up to 1000 times.

Emulation, emulate
Copying/representing a function or a device using a computer. The FST 200 contains emulators which copy/emulate display and control devices on the PC. This simplifies the commissioning process.
**Festo Software-Tools**

(FST) - software packages from Festo for menu and mouse-controlled programming of Festo control units.

**Field bus**

Serial bus system for information exchange between spatially distant parts of a process or technical production processes. Remote stations of the process have sensors, actuators and control units with differing levels of complexity. The principal advantages of using a field bus are the reduction in parallel wiring, the reduced workload of higher-order control units due to remote preliminary processing and the elimination of transmission difficulties for analogue values during conversion to digital format in the field bus station.

**Field bus address**

This address is required in order to identify the slaves in the field bus system. The field bus address is set using the relevant field bus station, either with address selector switch or with FST 200 software tools.

**Field bus module**

Field bus module makes it possible to connect an electronic controller (e.g. PLC, PC, programmable valve terminal) to a field bus system. The field bus module must be adapted to suit the field bus being used in order to support the physical and logical functions of the relevant bus system.
Independent/autonomous

Each programmable valve terminal has a built-in PLC and is therefore independent of a higher-order master. Once the commissioning of an SF 3 programmable valve terminal has been completed, it is then able to run without any auxiliary equipment and is therefore independent.

Interrupt

Interruption of a process or programs by an external event. An event caused by an interrupt is always assigned top priority and is processed immediately. Only then is the process or the program continued from the interrupt point.

LDR (ladder diagram)

Graphic description type for link-oriented control problems, based on the electric circuit diagram.

Multi-tasking

Ability of a computer system or control system to process several programs or processes (task = problem, sequential process) at the same time.
Operand

The control instruction of a program for a programmable controller is usually sub-divided into operation and operand: the operation indicates what is to be done and the operand indicates what to use to do it. The operand can contain the address of an I/O module as well as an internal controller address (e.g. a flag, timer, counter or similar).

Example: with the instruction:

\texttt{AND I3.2}

"AND" is the operation (logical AND) and "I3.2" is the operand.

Operating system

The operating system is a component of every computer system. It determines the performance and field of application of a system. The operating system of a programmable valve terminal controls the processor, manages the user and program memory and the peripheral devices connected to the diagnosis interface (PG, terminal, display and operating device).

The SF 3 programmable valve terminal also features a built-in field bus module (FESTO field bus). The operating system also manages all passive field bus stations and controls data traffic via the field bus.

Operation

Part of the command set of a PLC (also refer to operand)
Power failure

Operating voltage failure. For this, more precise differentiation is required on the AS-i bus. The AS-i master always reports "Power failure" when the AS-i master has no AS-i operating voltage. This may be due to the following reasons:
- Interrupt in AS-i flat cable leading to AS-i master.
- Failure of AS-i power pack.

Process image

(PIO = writing process image for outputs, PII = reading process image for inputs).

In order to achieve a defined switching attitude for inputs and outputs in a control unit (local I/Os and field bus I/Os), these are only read/set at a defined time. In the operating system of a programmable valve terminal, this always takes place at the same time as a task change.

A task change occurs whenever

- A program step is completed in a statement list (STL) step program.
- A program has run in a statement list (STL) sequencing program or LDR (ladder diagram) program.

With SF 3 valve terminals, this function also applies to field bus I/Os.
Program module

For ease of programming and to reduce the work-load of the user program, frequent command sequences or text messages are handled by the program module (sub-programs). The text messages (text modules) are produced by a special text display editor and are filed as program modules. Program modules are called up by the current program. SF 3 programmable valve terminals manage up to 16 program modules in user memory. If independent slaves are used, each of these slaves can manage up to 16 program modules.

Programmable valve terminals

The programmable valve terminal is made up of:
- A standard valve terminal
- A control unit (control block)

The control block contains the entire control electronics, i.e. programmable logic controller, memory modules (RAM and EEPROM) and the operating system together with the command interpreter. This enables the programmable valve terminal to perform automation tasks independently on location (also refer to valve terminal and valve/sensor terminal).

Protocol

Agreement which allows two or more items of equipment to communicate with one another. For instance, a protocol establishes which control characters indicate the beginning and end of a text entry.
RAM (Random Access Memory)

Memory with random access to data. The contents of this kind of memory can be read, deleted or modified by a computer. Access to data in the memory is gained by specifying the address. This makes it possible to gain rapid access at all times to data stored in the memory. Since RAM loses its storage ability when power is switched off, its contents have to be transferred to a different storage medium (EEPROM, hard disk) or backed up on battery storage to ensure that no data are lost.

Relocatable

(relocatable = variable location; opposite: absolute)
In general terms, a program written in a high-level language (LDR, STL) can be relocated to any position in the memory (relocatable). If fixed (absolute) addresses are defined for a given process for the translation into the machine language, a program of this kind is only able to run at one memory location.

Remanent

Remanent operands do not change their status when the control unit is switched off, then switched back on.

Some of the operands for the programmable valve terminal are "saved" in non-volatile memory (EEPROM) in the event of a power failure. When the power is switched back on, these data can be called up again unchanged by the user. The non-remanent (unsaved) operands lose their contents as soon as power is switched off.
RS 232

International standard format for transmission of data between different devices (PC, printer, control unit etc.). The transmission of data to the programmable valve/sensor terminal is performed using an RS 232 interface (diagnosis interface) with the following specification: serial, asynchronous, full-duplex and with software-handshake. The transmission rate can be set to 300, 2400, 4800 or 9600 baud.

Status byte

(refer to diagnosis byte)

STL (Statement List)

Programming language for programmable controller and programmable valve terminals: the command set of this language can be used to program all logical links and procedures. A statement list (STL) can contain step-oriented as well as link-oriented components.

Task

(problem, sequential process)

A term used frequently in open-loop control technology whenever a processor/operating system is able to perform several tasks at once. An operating system of this kind can be referred to as a multi-tasking system (e.g. programmable valve terminal).
Task change

If several programs are processed at the same time, the operating system sub-divides each of these programs into tasks. The tasks are then processed in sequence by the processor. Task changes occur whenever:

- An LDR or STL sequencing program is processed.
- Whenever a step is completed in a statement list (STL) program.

At every task change, the process image (PIO/PII) is read/written. This means that a SF 3 valve terminal with appropriate programming can operate very rapidly.

Terminating resistor

Resistance network for the line termination of bus cable; terminating resistors are always required on the ends of cables and/or segments.

Toggle

Continuous switching on/off of an output with FST 200 in online mode (set, reset).

User program

A program written by or for a user for a control system, e.g. a PLC with which specific control problems are solved for the user. In the SF 3 valve terminal, user programs are written in STL or LDR and can be stored in non-volatile form in the EEPROM.
## D.2 INDEX

### A
- Abbreviations .................................................. 1-7
- Actuator voltage supply
  - overload/short circuit. .................................. 5-43
  - Address assignment after extension/conversion .. 2-47
  - Address assignment of type 03 valve terminal .. 2-43
- Addressing
  - analogue channels ....................................... 5-23
  - analogue I/Os basic points ............................. 5-21
  - analogue inputs/outputs ................................. 5-22
  - AS-i slaves ............................................... 4-20
  - field bus I/Os .......................................... 4-4
  - type 03 .................................................... 2-41
  - valves ................................................... 2-43, 5-22
- Analogue I/Os
  - activating with SF 3 ................................... 5-26
  - assembly .................................................. 5-9
  - characteristics after switching on .................. 5-27
  - display and control units .............................. 5-4
  - pin assignment current I/Os ......................... 5-18
  - pin assignment current I/Os (Prop) ................. 5-16
  - pin assignment voltage I/Os ........................... 5-17
- AS-i
  - addressing ................................................. 6-44
  - addressing with FST 200 ................................. 6-47
  - basic variants ............................................ 6-9
  - cable selection .......................................... 6-22
  - commissioning - basic points .......................... 6-49
  - commissioning with FST 200 ............................. 6-57 - 6-58
  - mounting the components .............................. 6-17
  - online operation ........................................ 6-73
  - operands .................................................. 6-60
  - pin allocation ............................................ 6-29
  - startup performance .................................... 6-76
  - user instructions ........................................ 6-3
- Assembly ..................................................... 5-9
- Automode ....................................................... 3-16

### B
- Basic rule for type 03 addressing ....................... 2-44
- Boot mode ..................................................... 3-15
### C

**Cables**
- connection ........................................... A-4
- diagnosis interface ................................. 2-28
- field bus interface ................................. 2-30
- operating voltage .................................. A-9
- select .................................................. 5-13, A-9

**Calculating current consumption for type 03** ........................................ 2-24

**Command set** ................................. 3-18, 3-20, 3-22

**Commissioning**
- SF 3 as master ........................................ 4-11
- SF 3 as slave .......................................... 4-15
- SF 3 as standalone .................................... 3-27

**Connecting up**
- AS-i lines ........................................... 6-24
- AS-i power pack ..................................... 6-30
- field bus interface ................................. 2-30

**Current**
- analogue current I/Os ............................... 5-18
- analogue current I/Os (Prop) ....................... 5-16
- analogue voltage I/Os ............................... 5-17

### D

**Danger categories** ................................. 1-5

**Data transmission** ................................. 4-25
- settings ............................................. 3-3, C-3

**Defects**
- program ................................................ 3-60

**Designation of inputs and outputs** ................................. 2-37

**Diagnosis**
- cable .................................................. 2-28
- error word .......................................... 3-55, 3-62, 4-43
- functional module ................................... 4-43, 4-45
- LED current I/Os ................................... 5-40
- LED voltage I/Os ................................... 5-41
- LEDs .................................................. 3-55 - 3-56, 4-43
- LEDs (Prop) .......................................... 5-39
- status bit .......................................... 3-55, 4-43
- status byte .......................................... 3-55, 3-60, 4-44

**Dismantling input/output modules** ................................. 2-8, 2-10

**Display elements, type 03** ................................. 2-4

**Distances, permissible** ................................. 2-23

**DUO cable** .......................................... 2-36
E
Earthing .................................................. 2-13
Electrical connections type 03 ......................... 2-21
Emulator ................................................ 1-19
End plates ................................................. 2-12
Error
evaluation .................................. 2-32, 3-55, 3-60, 3-62, 4-43 - 4-44, 4-50
LED display ......................................... 3-56
Error diagnosis of AS-i ................................. 6-85
Error messages .................................. 3-60, 3-62, 4-44, 4-50
operating system .................................. 3-62, 4-50
Establishing configuration data for type 03 ........... 2-41
Example
emulator as commissioning support ............... 1-19

F
Festo software tools ..................................... 1-11, 4-3
Field bus
terface .................................................. 2-30
FST 200 ................................................... 3-19
Function
trol block ............................................... 1-11
Function modules .................................. 3-19, 3-61, 4-45
activation of analogue I/Os ......................... 5-27
analogue valve output ................................ B-65
diagnosis of analogue channels ................. B-66
output of analogue values ......................... 5-29
read in analogue values ......................... 5-28, B-64

I
Inputs/output module, type 03/05 ...................... 2-10
Inputs/outputs
activating analogue I/Os ........................... 5-28
Instructions for Manual ............................. 1-7
### Glossary

#### L
- LDR .................................................. 3-19
- LED display
  - AS-i master diagnosis LEDs AS-i .................. 6-86
  - AS-i slaves diagnosis LEDs AS-i .................. 6-87
  - inputs .......................................... 3-59
  - node ........................................... 3-56
  - outputs ....................................... 3-59
  - valves .......................................... 3-58

#### M
- Module ............................................ 4-45, B-11
- Mounting input/output modules ..................... 2-11
- Mounting the type 03 components ................... 2-9

#### O
- Online operation .................................. 3-5, C-1
- Operands
  - error word .................................... 3-25
  - errors ......................................... 3-25
  - flag words .................................... 3-24
  - flags .......................................... 3-24
  - init flags .................................... 3-24
  - input words .................................. 3-24
  - output words .................................. 3-24
  - program ....................................... 3-25
  - program module ................................ 3-25
  - register ....................................... 3-25
  - remanent RAM/EEPROM ......................... 3-11
  - special operands ............................... 3-25
  - timer words .................................. 3-24
- Operating elements, type 03 ....................... 2-4
- Operating mode
  - master ........................................ 4-9, 4-11, 4-20, 4-22
  - slave ......................................... 4-9, 4-15, 4-20
- Operating status
  - inputs ......................................... 3-59
  - outputs ....................................... 3-62
  - valves .......................................... 3-58
<table>
<thead>
<tr>
<th>Operating system</th>
<th>3-19, 4-19</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating voltage, tolerance</td>
<td>2-25, A-17</td>
</tr>
<tr>
<td>Operation and display element</td>
<td>1-11</td>
</tr>
<tr>
<td>Operators</td>
<td>3-18, 3-20, 3-22</td>
</tr>
<tr>
<td>Overload/short circuit</td>
<td></td>
</tr>
<tr>
<td>actuator voltage supply</td>
<td>5-43</td>
</tr>
<tr>
<td>analogue voltage outputs</td>
<td>5-31, B-67</td>
</tr>
</tbody>
</table>

**P**

Pictograms | 1-6 |
PII | 4-25 |
Pin allocation for operating voltage | 2-26 |
Pin assignment | |
| analogue current inputs/outputs | 5-18 |
| analogue voltage inputs/outputs | 5-17 |
AS-i master | 6-29 |
| diagnostic interface | 2-29 |
field bus interface | 2-34 |
for proportional valves | 5-16 |
PIO | 4-25 |
Programmable valve/sensor terminals | 1-8 |
Programmer | 1-21 |
Programming | |
| automode | 3-16 |
| boot mode | 3-15 |
| loading programs | 3-8 |
| programming technique | 3-19, 4-19 |
Project | 3-8 |
Protective grounding type 03 | 2-27 |
Protocol | C-3 |

**R**

RAM | 3-23 |
Relocatable | C-13 |
Return parameters | 3-61, 5-28, 5-30, 6-97, B-9 |
RS 232 interface | 2-28, 3-3 |
| S     | Screening                  | 5-14  |
|      | analogue I/Os              |       |
|      | diagnosis interface        | 2-29  |
|      | field bus interface        | 2-34  |
|      | Short circuit              |       |
|      | rectify                    | 4-45  |
|      | remedy                     | 3-62  |
|      | Status byte                | 4-40, 4-44 |
|      | Structure of type 03 valve terminal | 2-3 |
|      | System structure           | 1-11, 4-3 |
|      | System summary             | 1-11  |

| T     | Technical data             |       |
|      | analogue module            | 5-47  |
|      | AS-i master                | 6-101 |
|      | type 03                    | 2-51  |
|      | Timer preselection         | 3-24  |
|      | Top-hat rail clamping unit | 2-14  |

| V     | V.24 interface             | 2-28, 3-3 |
|      | Voltage                    | A-4     |
|      | connection                 | A-4     |
|      | select cable               | A-9     |

| W     | Wire cross section         | 2-23   |