SIEMENS

SIMATIC S7-300

FM 354 Servo Drive Positioning Module

Manual 04.97 Edition

This manual is intended to accompany the configuration package,
Order No.: 6ES7 354-1 AH01-7AG0.

SIEMENS

SIMATIC S7

FM 354 Servo Drive Positioning Module

Manual

Safety-related Guidelines

This manual contains notices intended to ensure personal safety, as well as to protect the products and connected equipment against damage. These notices are highlighted by the symbols shown below and graded according to severity by the following texts:

Danger

indicates that death, severe personal injury or substantial property damage **will** result if proper precautions are not taken.

Warning

indicates that death, severe personal injury or substantial property damage **can** result if proper precautions are not taken.

Caution

indicates that minor personal injury or property damage can result if proper precautions are not taken.

Note

contains important information about the product, its operation or a part of the document to which special attention is drawn.

Qualified Personnel

A device/system may only be commissioned or operated by **qualified personnel**. Qualified personnel as referred to in safety guidelines in this document are persons authorized to energize, de-energize, clear, ground, and tag circuits, equipment and systems in accordance with established safety practice.

Proper Usage

Please observe the following:

Warning

The equipment/system or the system components may only be used for the applications described in the catalog or the technical description, and only in combination with the equipment, components, and devices of manufacturers as far as this is recommended or permitted by Siemens.

The product will function correctly and safely only if it is transported, stored, set up, and installed as intended, and operated and maintained with care.

Trademarks

SIMATIC® and SINEC® are registered trademarks of SIEMENS AG.

Third parties using for their own purposes any other names in this document which refer to trademarks might infringe upon the rights of the trademark owners.

Siemens AG
Automation Group
Industrial Automation Systems
P.O. Box 4848,D- 90327 Nuremberg

Disclaimer of Liability

We have checked the contents of this manual for agreement with the hardware and software described. However, deviations cannot be precluded entirely. The data in this manual are reviewed regularly and any necessary corrections included in subsequent editions. Suggestions for improvement are welcome.

Preface

Purpose of this document

This manual contains all information about the FM 354 module:

- Hardware and functions
- Parameter definition
- Human-machine interface
- S7 function blocks
- Safe setup

Information blocks in this manual

The following information blocks describe the purpose and uses of this manual:

- Product overview of the module (Chapter 1)

  This section explains the purpose and possible applications of the module. It provides introductory information about the FM 354 and its functions.

- Basic principles of positioning (Chapter 2)

  Here you will find introductory information on positioning methods and associated definitions of terms.

- Installing and removing the FM 354 (Chapter 3)

  Explains the installation and removal of the FM 354.

- Wiring the FM 354 (Chapter 4)

  Describes the connection and wiring of drives, encoders and digital input/output modules.

- Defining parameters of the FM 354 (Chapter 5)

  Describes the parameterization and functions of “Parameterize FM 354.”

- Programming the FM 354 (Chapter 6)

  Describes how to program the FM 354 with STEP 7.

- Starting up the FM 354 (Chapter 7)

  Describes startup procedures for the FM 354.

- Human-machine interface (Chapter 8)

  – Describes the various options for operating and monitoring the FM 354, and which data and signals can be used and monitored.

- Reference information and appendices for finding factual information (module functions, programming guide, interface signals, parameter lists, error handling, technical specifications, standard HMI user interface)
- List of abbreviations and index for looking up information.

User requirements

The present manual describes the hardware and functions of the FM 354

To set up, program and start up a SIMATIC S7-300 with the FM 354, you will need a knowledge of:

- The SIMATIC S7

  Installation manual *S7-400/M7–400 Programmable Controller, Hardware and Installation*
- Your programming device (PG)
- How to perform programming with STEP 7
- How to configure a operator panel interface.

FM 354 users

The structure and presentation of the information in the manual are oriented to the intended uses of the FM 354, and the user's own activity.

It distinguishes among the following:

- Installation

  These activities include installation and wiring of the FM 354.
- Programming

  These activities include parameterizing and programming the FM 354.
- Troubleshooting and diagnostics

  These activities include detecting and correcting faults and errors

  – in the hardware setup of the module and its components

  – and in the programming, handling and control of module functions.
- Operation

  These users operate the FM 354. The operator accordingly deals only with the control of positioning tasks.

CE marking

Our products are in compliance with the EU Guideline 89/336/EEC “Electromagnetic Compatibility” and the harmonized European standards (EN) which it embodies.

The EC Declaration of Conformity in accordance with Article 10 of the EU Guideline referenced above is contained in this manual (see Chapter B).

Contact partners

If you should encounter any problems using this manual, or if you have any questions, please contact the office specified on the query form at the end of this manual.

Hotline

If you have an urgent problem, please contact:

Test Hotline, +49 911 / 895 – 7000

■

Contents

Figures

Tables

■

1 Product Summary

What can the FM 354 do?

The FM 354 is a microprocessor-controlled positioning module for a drive with an analog setpoint interface.

The FM 354 is a high-performance module for tasks in servo-controlled positioning.

The module works autonomously and is controlled by way of the user program in the SIMATIC S7-300 system.

It can operate rotary and linear axes by servo or open-loop control with actual-value tracking.

The FM 354 has a variety of operating modes.

The module has a non-volatile data memory to store parameterization data.

- The FM 354 is low-maintenance (no battery).
- It can be linked and adapted to user circumstances by parameterizing it as required by the system.

Where can the FM 354 be used?

The FM 354 can be used for both simple positioning and complex traversing profiles demanding superior dynamic response, accuracy and speed. It is also suitable for positioning tasks in machinery with high clock-pulse rates.

Typical uses for the positioning module might include:

- Transfer lines
- Assembly lines
- Presses
- Woodworking machines
- Handling equipment
- Loaders
- Auxiliary movements in milling and turning machines
- Packaging machines
- Conveyor equipment

Its range of functions is comparable to that of the WF 721 module in the SIMATIC S5 system, and the FM 353 in the SIMATIC S7 system.

Compatibility list

Please note the following compatibility specifications for the various product releases/order numbers when using the FM 354 with CPUs of the S7-300 system and with OPs.

Table 1-1 Compatibility list for use of the FM 354 with S7-300 CPUs and OPs

Modules are compatible with each other...		FM 354-1AH00-0AE0	FM 354-1AH01-0AE0
S7-300 CPUs			
313	313-1AD**01**-0AB0	**no**	**yes**
314	314-1AE**02**-0AB0		
314 IFM	314-5AE**01**-0AB0		
315	315-1AF**01**-0AB0		
315-2 DP	315-2AF**01**-0AB0		
S7-300 CPUs			
312 IFM	312-5AC**00**-0AB0	**yes**	**yes** **(no module replacement without PG/PC)**
313	313-1AD**00**-0AB0		
314	314-1AE**00**/**01**-0AB0		
314 IFM	314-5AE**00**-0AB0		
315	315-1AF**00**-0AB0		
315-2 DP	315-2AF**00**-0AB0		
CP 342-5	342-5DA**00**-0XE0	**yes**	**no**
OPs... (with Protocol **V 3.x)**			
OP 7/DP	607-1JC20-0XA0	**yes**	**yes**
OP 7/DP-12	607-1JC30-0XA0		
OP 17/DP	617 1JC20-0XA0		
OP 17/DP-12	617 1JC30-0XA0		
OP 25	525-1EA **.**1-0AX0		
OP 35	535-1FA01-0AX0		
OP 37	637-1 **.** L00-0 **.** X0		
OPs (with Protool **< V 3.x**)			
OP 3	503-1DB10	**yes**	**no**
OP 5/A2	505-1FB12		
OP 15/A2	515-1EB32-1AA0		
OP 15/C2	515-1MA22-1AA0		
OP 7/DP	607-1JC20-0XA0		
OP 7/DP-12	607-1JC30-0XA0		
OP 17/DP	617 1JC20-0XA0		
OP 17/DP-12	617 1JC30-0XA0		
OP 25	525-1EA **.**1-0AX0		
OP 35	535-1FA01-0AX0		
OP 37	637-1 **.** L00–0 **.** X0		

Chapter overview

1.1 The FM 354 in the S7-300 programmable controller

How is the FM 354 linked up with the S7-300?

The FM 354 is designed as a function module of the SIMATIC S7-300 controller.

The S7-300 programmable controller consists of a CPU and a variety of peripheral modules mounted on a mounting rail.

The configuration may have one or more racks.

Multi–rack configurations

A SIMATIC S7-300 CPU may run up to four racks with as many as eight bus stations each (see Figure 1-1).

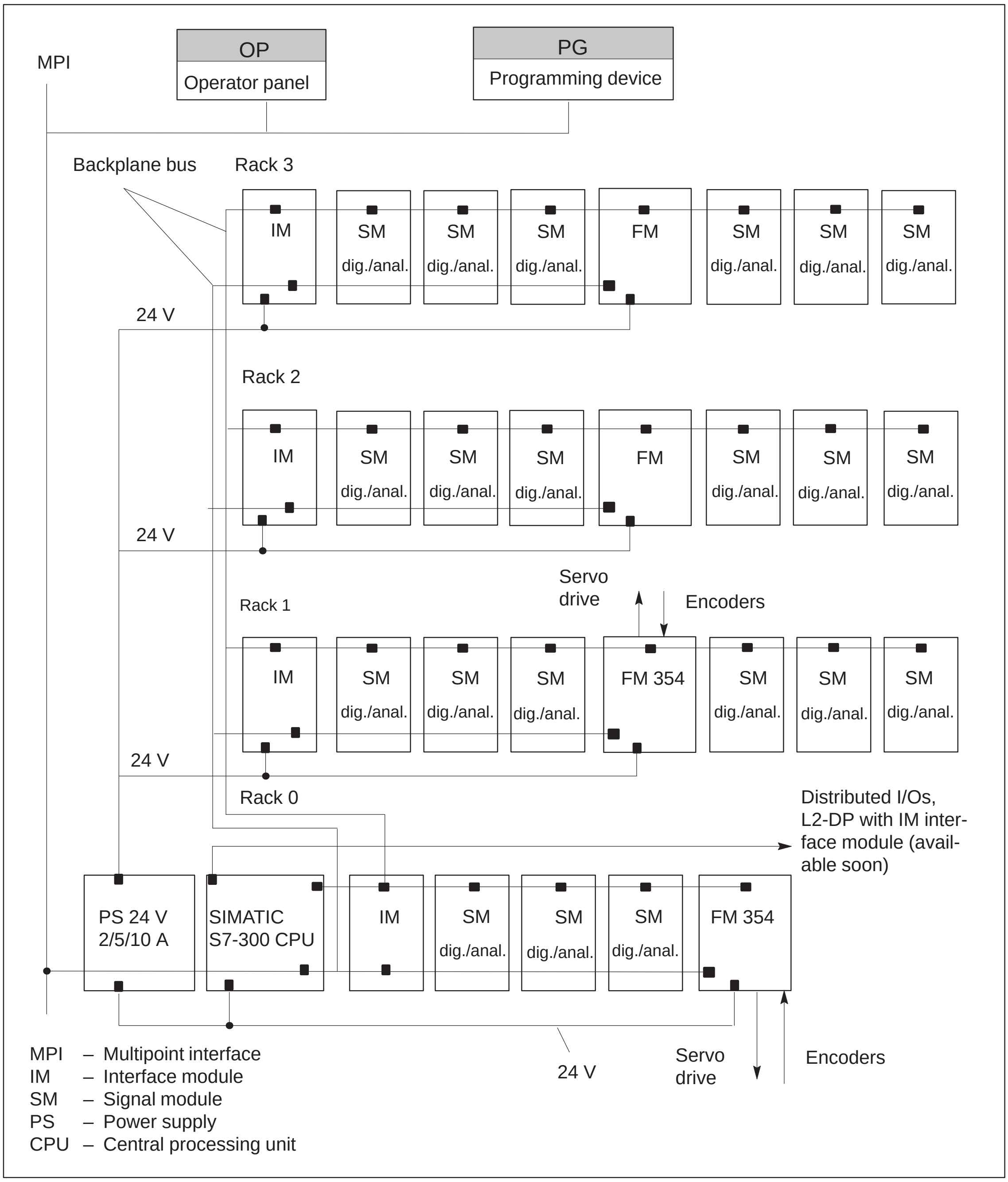


Fig. 1-1 Multi–rack configuration of a SIMATIC S7-300 with FM 354 (example)

Overview of system components

A positioning controller using the FM 354 consists of a variety of individual components. which are shown in Figure 1-2.

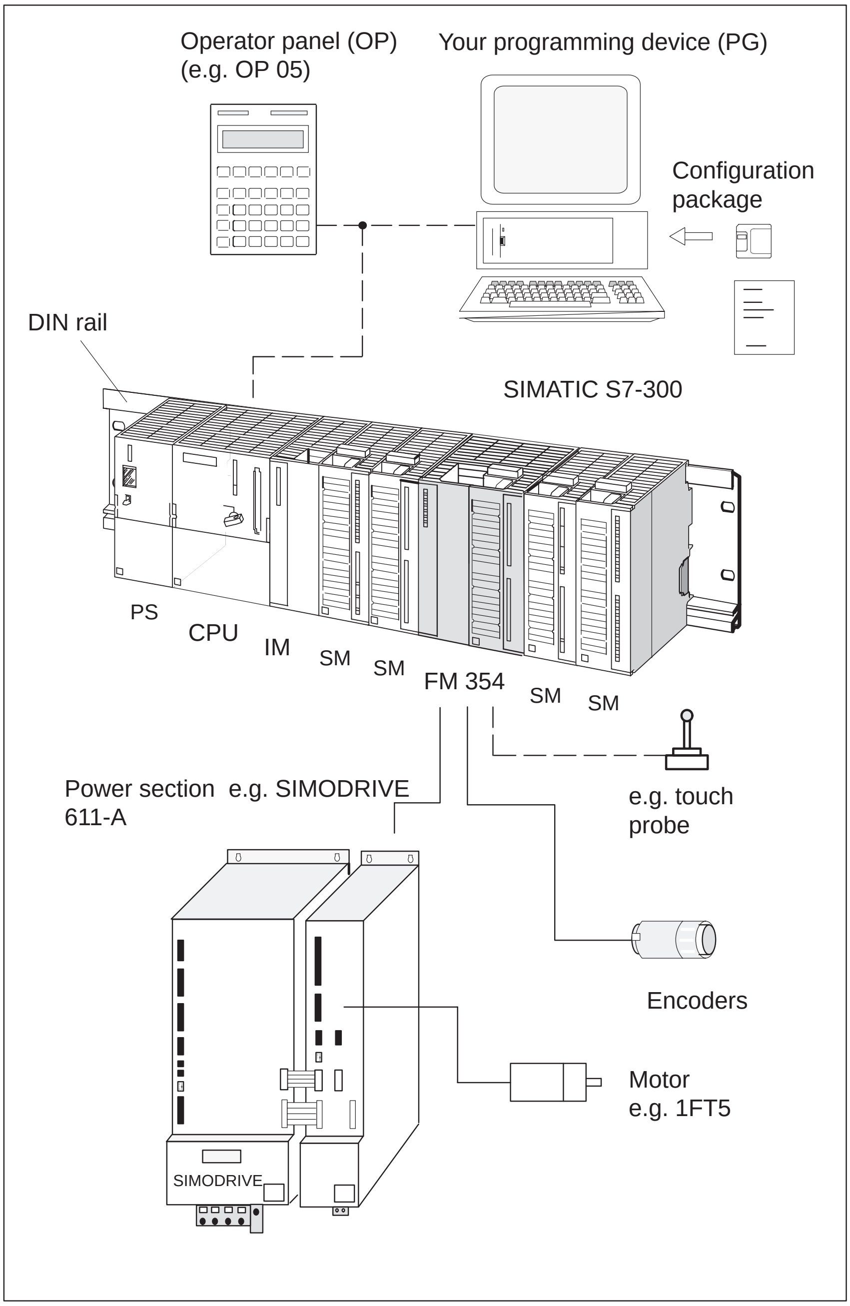


Fig. 1-2 System overview (schematic)

Components

The most important components and their functions are listed in Table 1-2.

Table 1-2 Components of a positioning controller

Component	Function
DIN rail	... the module mounting rack for the S7-300.
FM 354	... the positioning module. It is controlled by the S7-300 CPU.
CPU	... executes the user program; powers the S7-300 backplane bus at 5 V; and communicates with the programming device and the operator panel via the MPI interface.
Power supply (PS)	... converts line voltage (120/230 V AC) to 24 V DC operating voltage to power the S7-300.
Signal modules (SM)	... adapts various process-signal levels to the S7-300
Interface module (IM)	... connects the individual cells of an S7-300 with one another (applies to multi-tier configuration; see Figure 1-1).
Programming device (PG)	... configures, parameterizes, programs and tests the S7-300 and the FM 354.
Operator panel (OP)	... the interface to the machine. It serves for operation and monitoring. It is not an absolute prerequisite for operation of an FM 354.
Power section	... actuates the motor.
Motor	... drives the axis.
Encoders	... the path measurement system that detects the current position of the axis. By comparing the actual position with the applicable setpoint position, the FM 354 immediately detects discrepancies and attempts to compensate for them.
Configuration package	... includes: • A manual • 3 1/2” diskette with: – Function-block package FCs. – The “Parameterize FM 354” parameterization tool. – Preconfigured interface for COROS devices OP 07 and OP 17.

System overview of data handling

The following figure gives you an overview of the data storage concept.

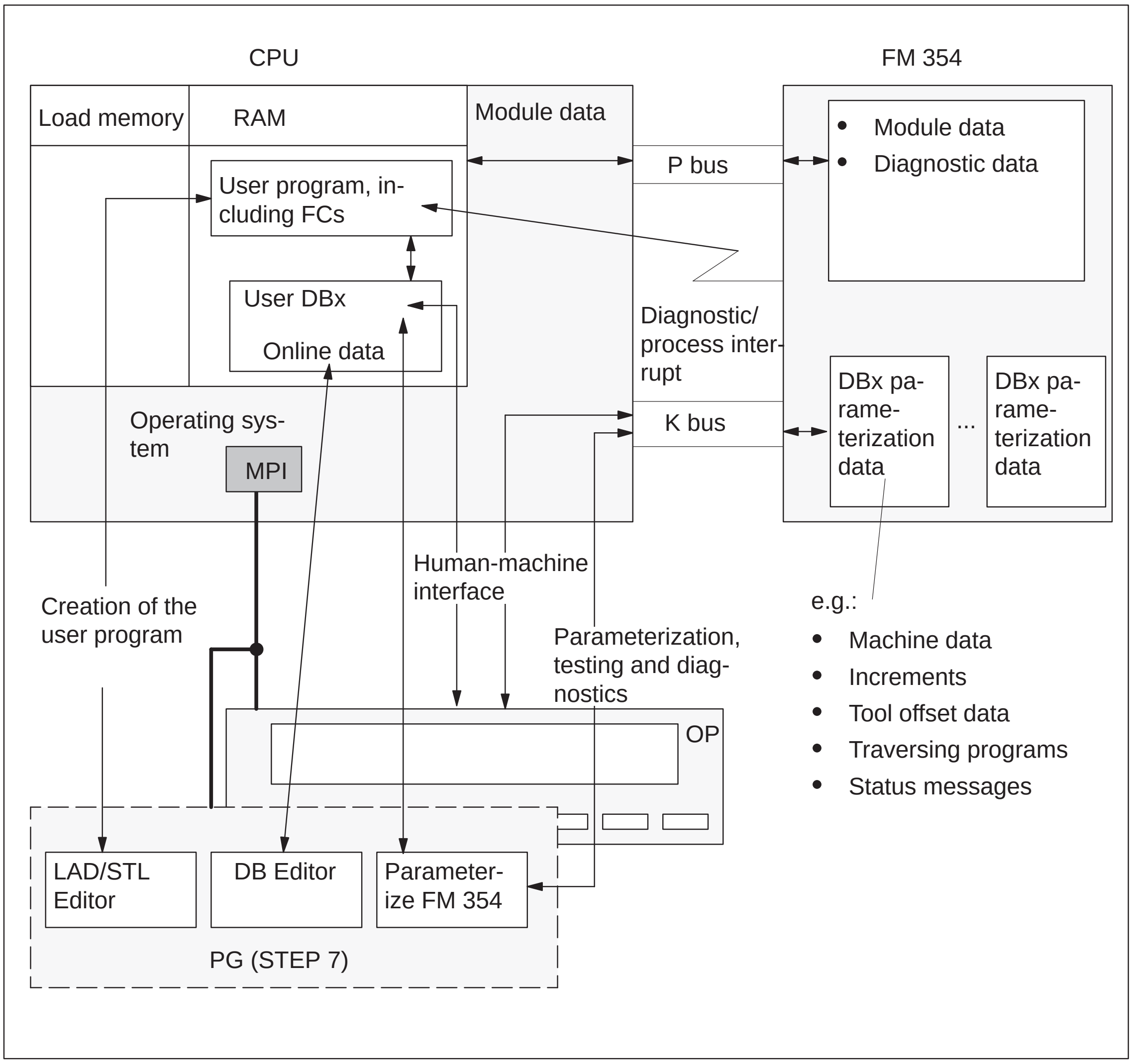


Fig. 1-3 Data storage concept

1.2 Module description

View of the FM 354 Figure 1-4 shows the FM 354 module, its interfaces and front-panel elements (including fault and status displays).

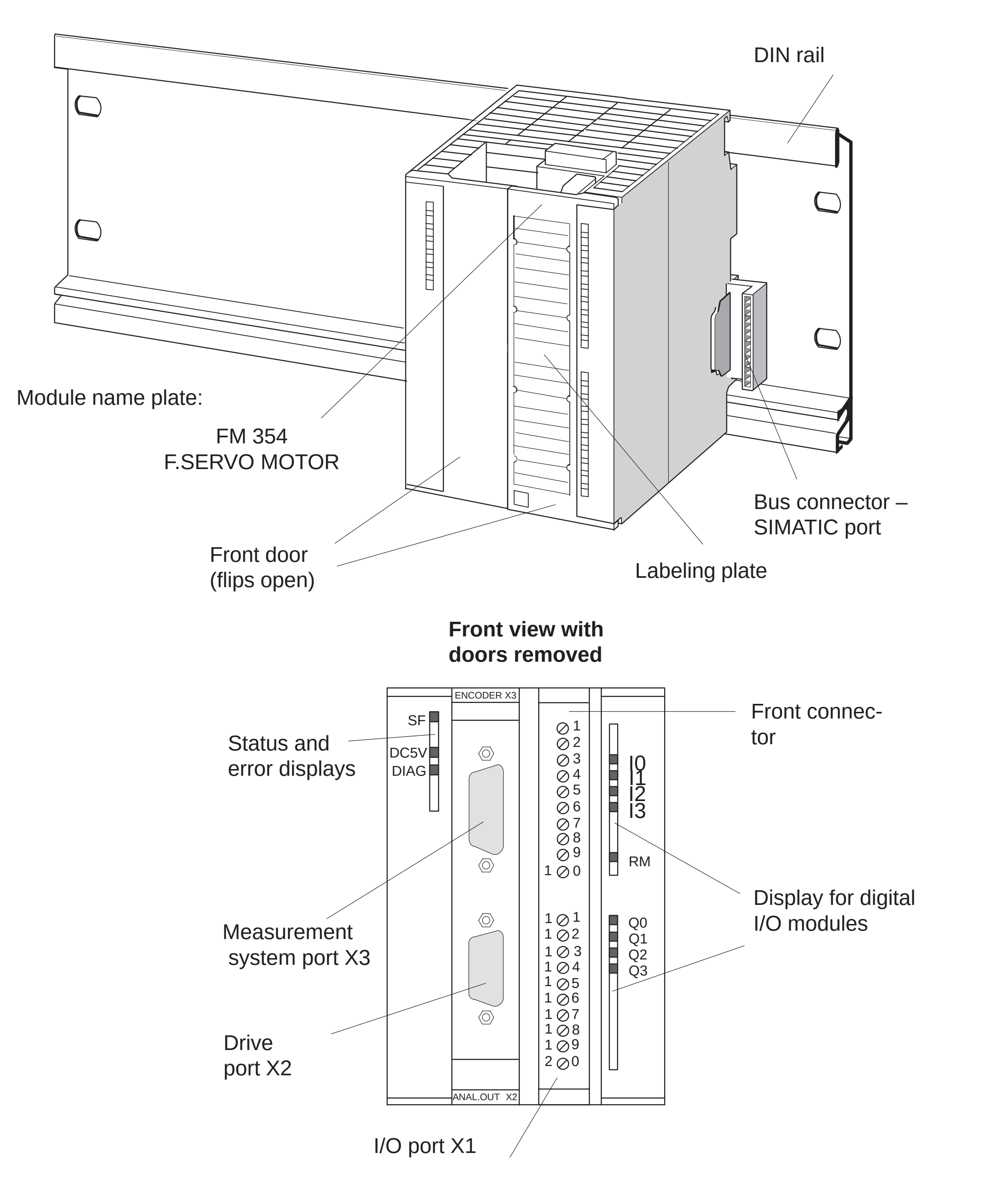


Fig. 1-4 View of the ports and front–panel elements

Ports

A description of the ports is provided in Table 1-3 .

Table 1-3 Ports

Ports	Description
Bus connector – SIMATIC port	Back connector to continue the S7 LAN from module to module
Drive port	9-pin male sub-D connector (X2) to connect the drive unit
Measurement system port	15-pin female sub-D connector (X3) to connect the encoder
I/O port	20-pin male front connector (X1) to connect the load power supply and for digital input and output wiring

LED indicators

Twelve LEDs are arranged on the front panel of the FM 354. Table 1-4 describes these LEDs and what they mean.

Table 1-4 Status and error displays

LED	Significance
SF (red) - Group error	This LED indicates an error condition in the FM 354. (see Troubleshooting, Chapter 11)
5 V DC (green) - Logic power supply is ON	This LED indicates that the hardware is ready for operation.
DIAG (yellow) - Diagnostics	This LED indicates the various diagnostic conditions. (see Troubleshooting, Chapter 11)
I0...I3 (green) - Digital inputs	These LEDs indicate which input is ON.
Q0...Q3 (green) - Digital outputs	These LEDs indicate which output is ON.
RM (green) - Drive unit ready	This LED indicates the drive unit is ready to operate.

Type plate of the FM 354

Figure 1-5 describes all the information contained in the type plate of the FM 354.

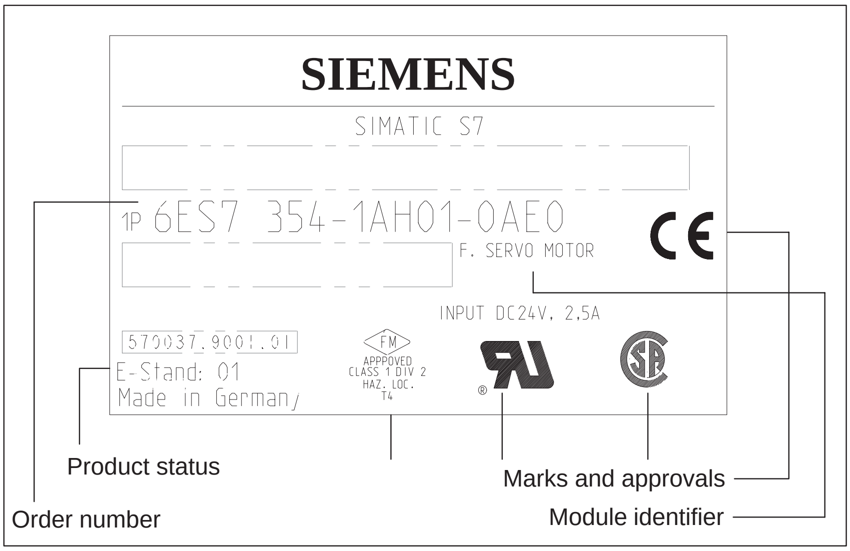


Fig. 1-5 Type plate of the FM 354

1.3 Overview of module functions

Summary

The FM 354 module performs the following functions:

- Mode control
- Actual-value capture
- Servo position control
- Digital inputs and outputs
- Settings and functions that do not depend on operating mode
- Software limit switches
- Process interrupts
- Block sequence control
- Diagnostics and troubleshooting
- Data storage on the FM 354

Operating mode control

The user program passes the operating mode to the FM.

The FM 354 has the following modes available:

- Jogging
- Open–loop control
- Reference point approach
- Incremental mode, relative
- Manual data input (MDI)
- Automatic
- Automatic single block

Encoders

Incremental or absolute encoders (SSI) may be connected to the measuring system port.

Position control

The position controller performs the following tasks:

- Guidance of the drive commensurate with speed during movement sequence (e.g. adjustable acceleration and delay, jerk limitation, following error monitoring), see Section 9.7 “Position control”)
- Precise approach of the axis to the programmed target position (e.g. approach to position; see Section 9.7 “Position controller”)
- Maintenance of the axis in position when outside factors interfere. (e.g. offset compensation; see Section 9.7 “Position control”)

Digital inputs/outputs

Four digital inputs and four outputs can be used as specified by the user.

You might connect:

- Reference-point switches
- Switches for external starting
- Touch probes
- Position reached, Stop ("PEH")
- Forward/backward rotation

The switching function is assigned to a given I/O number by way of the machine data.

Settings and functions not dependent on operating mode

Special functions can be activated by specific settings in the user program, in addition to the mode (e.g. inprocess measurement, retrigger reference point, etc.).

Software limit switches

The operating range (specified by software limit switches) is automatically monitored after synchronization is recorded.

Process interrupts

Process interrupts are triggered by such events as:

- Position reached
- Length measurement completed
- On-the-fly block change
- Inprocess measurement

Process interrupts are selected by way of machine data.

Block sequence control

Automatic processing of a traversing program, including subprograms created during the parameterization process. A number of traversing programs are available for execution on the module.

Diagnostics and troubleshooting

Startup and ongoing operation of the module are monitored by fault and diagnostic interrupts. Faults or errors are reported to the system and displayed by the LEDs on the module.

Data storage on the FM 354

Parameterization data (machine data, tool compensation data, traversing programs and increment sizes) is retained in storage on the FM 354.

■

Basic Principles of Positioning

2

What is positioning?

Positioning means moving a load to a defined position within a defined time, taking all influencing forces and torques into account.

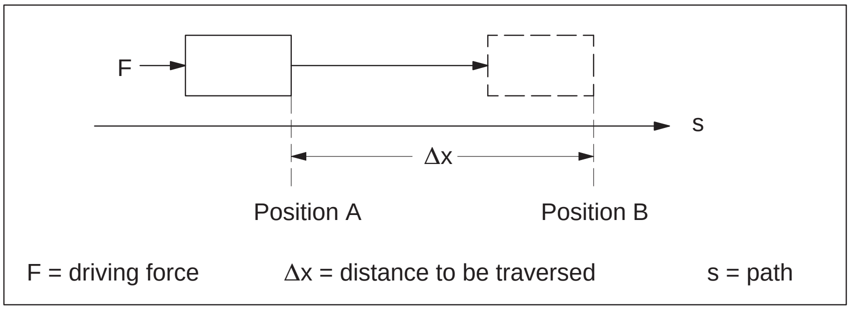


Fig. 2-1 Principle of a positioning action

Servo–controlled positioning

Servo–controlled positioning is:

- guidance of the drive at the proper speed during execution of a movement
- precise approach of the axis to the programmed target position
- maintenance of the axis in position in the face of interfering factors.

Structure of a positioning circuit

Figure 2-2 shows the structure of a position control circuit with the FM 354.

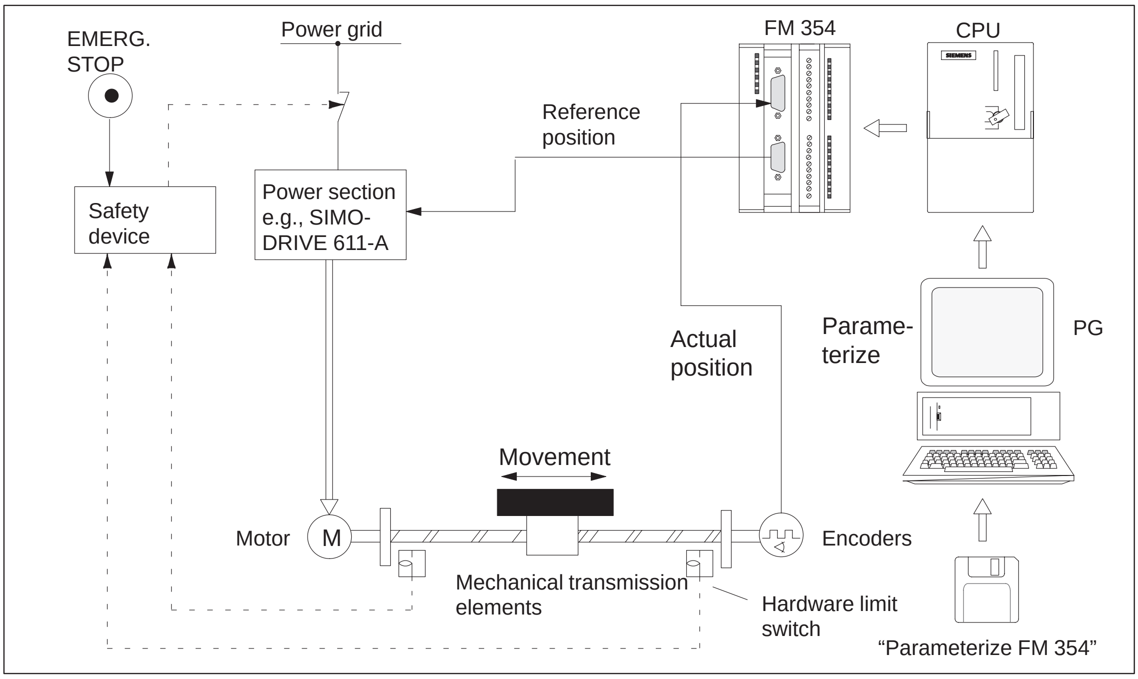


Fig. 2-2 Setup of servo-controlled positioning, example

FM 354

Servo-controlled positioning with output of an analog actuating signal for the drive.

Power section

The power section processes the analog actuating signal and delivers the proper electric power to the motor.

Motor

The motor is actuated by the power section and drives the axis.

Mechanical transmission elements

These include not only the axis, but also gear trains and clutch systems.

Encoder

The encoder detects movement of the axis. It supplies pulses to the FM 354. The number of pulses is proportional to the distance traversed.

Peripherals

All other additional equipment is covered by the term peripherals.

Peripherals mainly include:

- Limit switches to limit the positioning range (safety devices).
- A programming device (PG) and the “Parameterize FM 354” parameterization software.

■

Installing and Removing the FM 354 3

Overview

The FM 354 is intended for installation as an I/O module in the SIMATIC S7-300 programmable logic controller.

Important safety rules

There are important rules which you must follow when integrating an FM 354 in the S7-300 PLC in a plant or system.

These rules and specifications are described in the manual *S7-300 Programmable Controller, Hardware and Installation*.

Mechanical set–up

The options for the mechanical set–up and its configuration are described in the manual *S7-300 Programmable Controller; Hardware and Installation*, Order No.: 6ES7 030-0AA01-8AA0.

Below, we give only a few supplementary pointers.

Installation position

The module should preferably be installed horizontally.

In vertical installations, please observe the ambient temperature restrictions (max. 40 °C).

What you should know about the mechanical layout

The FM 354 can be mounted in any of the eight available slots (slots 4 to 11) for I/O modules on the mounting rail.

In configuring the mechanical layout of your controller, you should note the following rules:

1. No more than eight SMs or FMs per tier (rack).
2. The maximum number of modules is limited by module width and by the length of your DIN rail.

   The FM 354 requires an installation width of 80 mm (3.12 inches).
3. The maximum number of modules is limited by the total power that all modules to the left of the CPU or IM, as the case may be, consume from the 5 V backplane bus.

   The CPU 314, for example, can supply a maximum of 1.2 A.

   The FM 354 requires 100 mA of this amount.

Chapter overview

3.1 Installing the FM 354

Rules

No particular protective measures (EGB Guidelines) are necessary for the installation of the FM 354.

Warning

Install the FM 354 only after all power to the S7-300 has been turned OFF.

Tools required

A 4.5 mm (.18 inch) screwdriver.

Procedure

To install the FM 354:

1. The FM 354 comes with a bus connector. Plug this into the bus plug of the module to the left of the FM 354. (The bus plug is on the back; you may have to loosen the module already in place.)

   If further modules are to be mounted to the right, plug the bus connector of the next module into the right backplane bus connector on the FM 354.

   If the FM 354 is the last module in the rack, do not connect this bus connector.
2. Hook the FM 354 onto the rail and swing it down into position.
3. Screw the FM 354 down (torque approx. 80-110 Nm).
4. After the modules have been mounted, you can also assign each of them a slot number. Slot labels for this purpose are enclosed with the CPU.

   The numbering scheme and how to plug in the slot labels are described in the manual *S7-300 Programmable Controller, Hardware and Installation*, Order No. : 6ES7 030-0AA01-8AA0, for the numbering scheme to follow and how to apply the slot labels.

Note

The slot determines the initial address of each module. To find out how to allocate the module start address, please refer to the manual *S7–300 Programmable Controller, Hardware and Installation*, Order No.: 6ES7 030-0AA01-8AA0.

The FM 354 is addressed in the same way as an analog module.

3.2 Removing the FM 354

Rules

No particular protective measures (EGB Guidelines) are necessary for the removal of the FM 354.

Warning

Remove the FM 354 only after all power to the S7-300 has been turned OFF.

Tools required

A 4.5 mm (.18 inch) screwdriver.

Procedure

To remove the FM 354:

1. Open the front doors. If necessary, remove the labeling strips.
2. Detach the power-supply connections from the terminal block.
3. Detach the sub-D plugs from the encoder and drive unit.
4. Release the protective device on the front connector and unplug it.
5. Loosen the fastening screws and swing the module up and out.

3.3 Module replacement

Overview

If a defective FM 354 has to be replaced, and no programming device/PC is available for parameterization, or the module is to be replaced while the system is switched on, please note the following start–up requirements (CPU, FM):

- An SDB $\geq$ 1 000 should be generated in order to complete the startup (for storing the parameter data); see Section 5.5.
- In the user program:
  - Interrupt communication with the FM 354 before removing the old FM, and resume communication after installing the new FM.
  - If data/parameters are modified during operation and stored modally on the FM, please follow the instructions in Section 9.3.1.

Replacing an FM 354

To replace a parameterized but defective FM 354:

1. Replace the FM 354 when the system is switched off (CPU, FM)

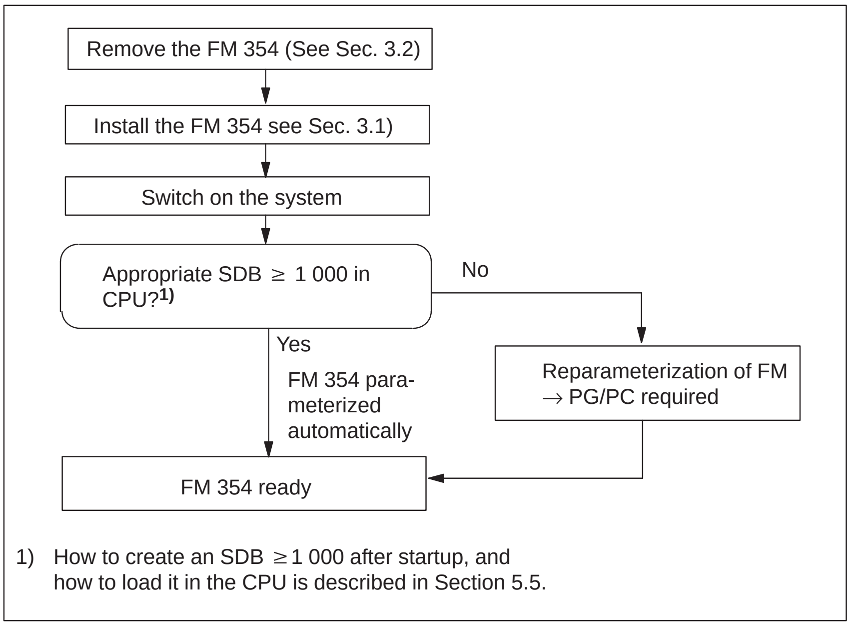


Fig. 3-1 Replacing the FM 354 with the system switched off

■

Wiring the FM 354

4

Safety rules

In order to ensure the safe operation of your plant, you should introduce the following additional measures, and adjust them appropriately to your system's conditions:

- An EMERGENCY STOP concept meeting appropriate safety regulations (e.g. European standards EN 60204, EN 418 and associated standards).
- Additional measures for limiting the end position of axes (e.g. hardware limit switches).
- Equipment and measures for protecting the motors and power electronics in accordance with the installation guidelines for SIMODRIVE.

We also recommend you carry out a risk analysis in accordance with basic safety requirements / Appendix 1 of the EC machine directive, in order to identify sources of danger affecting the complete system.

Further references

Please refer also to the following chapters in the manual *S7-300 Programmable Controller, Hardware and Installation*:

- Guidelines for handling of electrostatic sensitive devices (ESDs): Appendix B.
- Configuring the electrical installation: Chapter 4

For further information about EMC guidelines, we recommend the description in: *Equipment for Machine Tools, EMC guidelines for WS/WF equipment*, Order No.: 6ZB5 440-0QX01-0BA1.

Standards and specifications

When wiring the FM 354 you must observe the relevant VDE guidelines.

Chapter overview

4.1 Wiring an FM 354

Summary

Figure 4-1 shows how the individual components of the positioning controller with FM 354 are linked together.

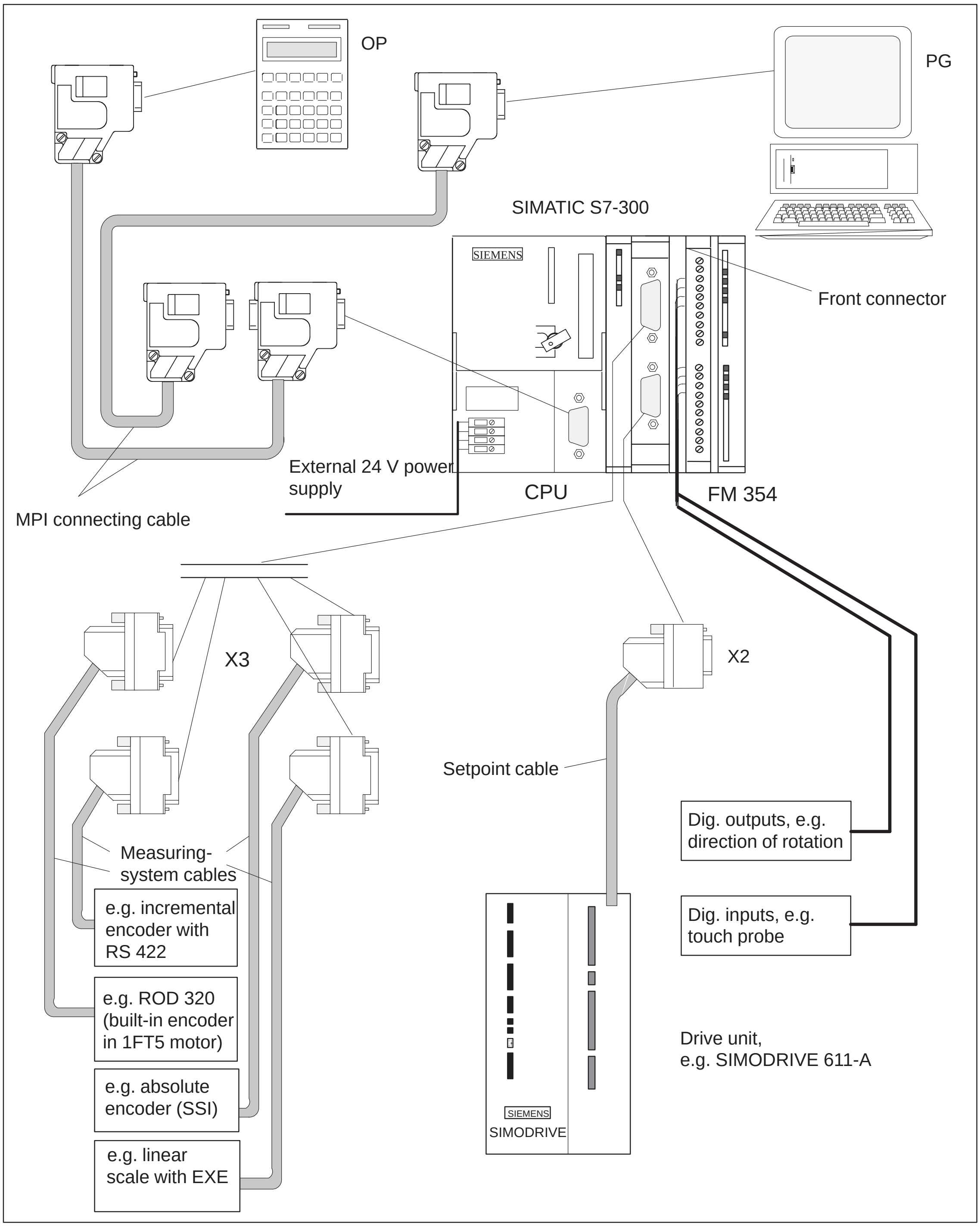


Fig. 4-1 Wiring diagram of an FM 354

Connecting cables

Table 4-1 lists the connecting cables for a positioning controller with the FM 354.

Table 4-1 Connecting cables for a positioning controller with FM 354

Type	Order No.	Description
MPI connecting cable	see *Catalog ST 70*, Order No. E86060-K4670-A101-A2	Connection between OP, PG and S7-300 CPU
Setpoint cable	6FX2 002-3AB01-1□□0 see *Catalog NC Z* Order No.: E86060-K4490-A001-A4	Setpoint cable for FM 354 and SIMODRIVE 611-A, ±10 V; one axis
Measurement system cable	6FX2 002-2CD01-1□□0 see *Catalog NC Z* Order No.: E86060-K4490-A001-A4	Incremental encoder with RS 422 and FM 354 (EXE with linear scale)
Measuring-system cables	6FX2 002-2CE01-1□□0 see *Catalog NC Z* Order No.: E86060-K4490-A001-A4	ROD 320 encoder with 1FT5 motor and FM 354
Measuring-system cables	6FX2 002-2CC01-1□□0 see *Catalog NC Z* Order No.: E86060-K4490-A001-A4s	Connection of absolute encoder (SSI) and FM 354

Front connector

You need a 20–pin screw–type front connector for wiring the I/Os. It must be ordered separately.

Order No.: 6ES7 392-1AJ00-0AA0

see *Catalog ST 70*, Order No. E86060-K4670-A101-A2

see *Catalog NC 60.1*, Order No. E86060-K4460-A101-A3

4.2 Description of the drive interface

Connector for the drive unit

Power sections with an analog interface (± 10 V) can be connected to the 9–pin sub–D X2 connector of the FM 354.

The FM 354 also provides an enable signal.

Connector location

Figure 4-2 shows the installation position and identification of the plug on the module.

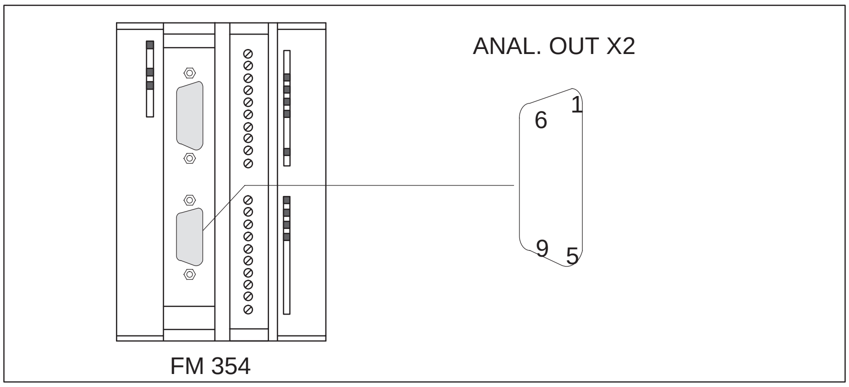


Fig. 4-2 Position of X2 connector

Connector pinout

Connector identifier: **X2 ANAL. OUT X2**
Connector type: 9-pin sub-D plug connector

Table 4-2 Pinout of the X2 connector

Pin	Name	type	Pin	Name	type
1	SW	VO	6	BS	VO
2	open		7	open	
3	open		8	open	
4	open		9	RF.1	K
5	RF.2	K			

Signal names

SW Setpoint ± 10 V
BS Reference potential for setpoint
RF.1...2 Contact for CL controller enable

Signal type

VO Voltage outlet
K Switching contact

Signals

One voltage signal and one enable signal are provided.

- **SETPOINT (SW)**

  An analog voltage signal in the range ± 10 V, for output of an rpm set-point.

- **REFERENCE SIGNAL (BS)**

  A reference potential (analog ground) for the setpoint signal, internally connected with the logic ground.

- **SERVO ENABLE (RF)**

  A relay contact pair used to switch the axis-specific enables of the power section, for example of a SIMODRIVE drive unit. The FM 354 activates this signal when cyclic control operation is begun - in other words, when power-up and parameterization have been completed.

Signal parameters

The setpoint is output as an analog differential signal.

Table 4-3 Electrical parameters of the setpoint signal

Parameters	Min	Max	Unit
Rated voltage range	–10	10	V
Output current	–3	3	mA

Relay contacts

The axis enables are switched via relay outputs (“make” contacts).

Table 4-4 Electrical parameters of the relay contacts

Parameters	Max	Unit
Switching voltage	50	V
Switching current	1	A
Switching capacity	30	VA

Connecting cables to drive unit

Acceptable length: up to 35 m (115 ft)

4.3 Connecting the drive unit

To connect the connecting cables

Please note:

Note

Use only shielded twisted pairs for lines. The shielding must be connected to the metallic or metallized connector jacket on the controller side. To protect the analog setpoint signal against low-frequency interference, we recommend that you not ground the shielding on the drive-unit side.

The cable set supplied as an accessory offers excellent immunity against interference.

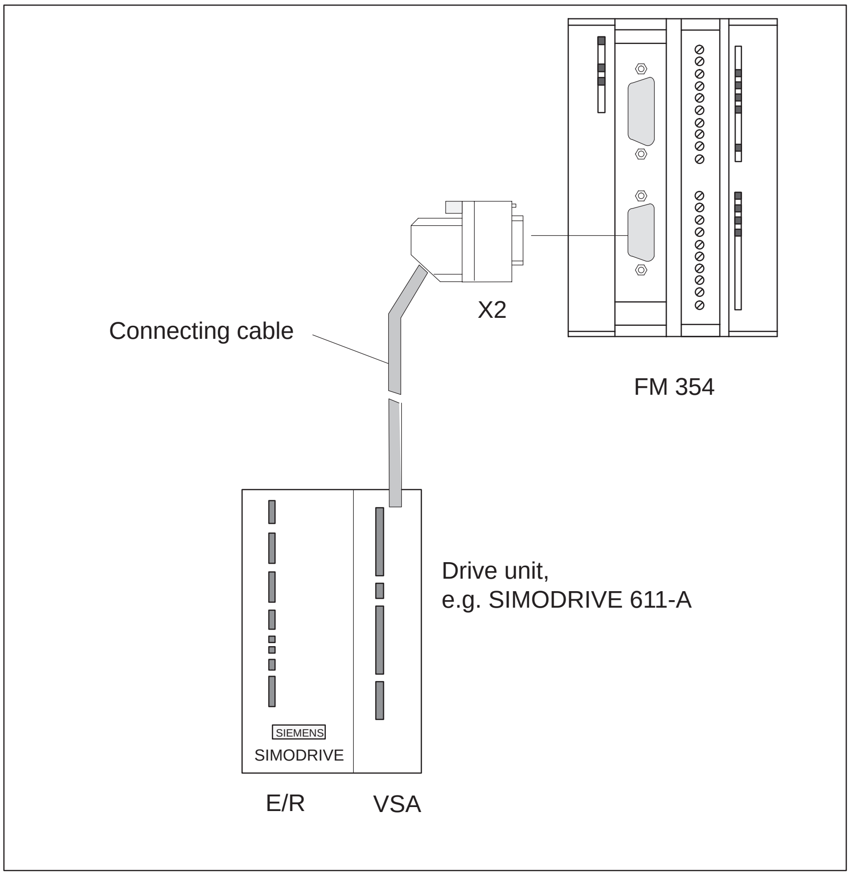


Fig. 4-3 Connecting a SIMODRIVE 611-A drive unit

Procedure to connect the connecting cable

Connect the drive unit as follows:

1. Wire the free cable end of the connecting cable to the terminals of the drive unit. (The terminal identifiers on the cable ends indicate the proper terminals for SIMODRIVE units.)
2. Open the front door and plug the sub-D connector into the module.
3. Lock the connector in place with the knurled screws. Close the front door.

Designation of the connecting cable

The connecting cable is a prefabricated cable for an axis with an analog interface, terminal designation for SIMODRIVE drive units.

Order No.: 6FX2 002-3AB01-0☐☐0

The connecting cable is available in a variety of lengths.

see *Catalog NC Z* , Order No.: E86060-K4490-A001-A4.

4.4 Description of the measurement system interface

Connectors for encoders

A 15–pin female sub D connector is provided for the connection of incremental encoders or absolute encoders (serial port).

Location of connector

Figure 4-4 shows where the connector is installed on the module, and how it is identified.

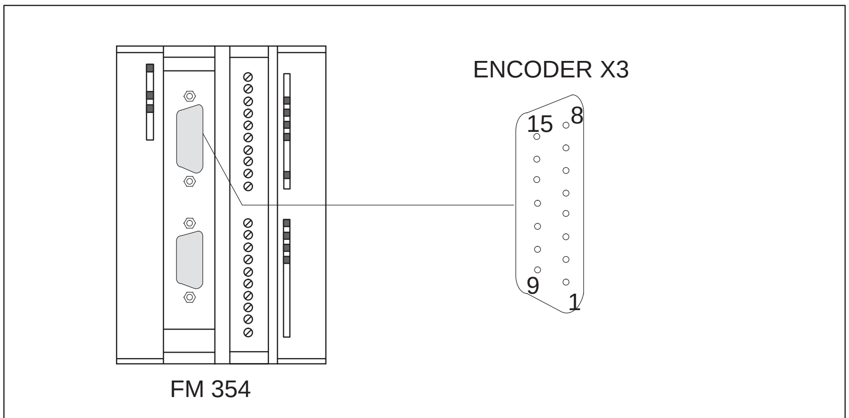


Fig. 4-4 Location of the X3 connector

Connector pinout

Identifier: **X3** **ENCODER X3**
Type: 15-pin female sub-D plug connector

Table 4-5 Pinout of the X3 connector

Pin	Encoders		type	Pin	Encoders		type
	Incremental	Absolute			Incremental	Absolute	
1	W_N		I	9	MEXT		VO
2		CLS	O	10	N		I
3		CLS_N	O	11	N_N		I
4	P5EXT		VO	12	B_N		I
5	P24EXT		VO	13	B		I
6	P5EXT		VO	14	A_N	DATA_N	I
7	MEXT		VO	15	A	DATA	I
8	open						

Signal names

W_N	Maintenance signal negated (incremental encoder)
A, A_N	Track A true / negated (incremental encoder)
B, B_N	Track B true / negated (incremental encoder)
N, N_N	Zero mark true / negated (incremental encoder)
CLS, CLS_N	SSI sliding pulse true / negated (absolute encoder)
DATA, DATA_N	SSI data true / negated (absolute encoder)
P5EXT	+5 V power supply
P24EXT	+24 V power supply
MEXT	ground power supply

Signal type

VO	Voltage outlet (power supply)
O	Output (5 V signal)
I	Input (5 V signal)

Connectable encoder types

Incremental or absolute (SSI) encoders may be connected directly (e.g. digital-rotary encoders); they are then selected via machine data.

Encoders with SINE/COSINE signals (e.g. length scales) may be connected by way of an external electronic pulse shaper (EXE) that converts the signals to 5 V levels.

Encoder characteristics

Both encoders that can be connected directly and EXEs must meet the following requirements:

Incremental Encoders

Transfer procedure:	Differential transfer with 5 V rectangular signals (such as RS422 standard)
Output signals:	Track A as true and negated signal (U_{a1}, $\overline{U_{a1}}$) Track B as true and negated signal (U_{a2}, $\overline{U_{a2}}$) Zero signal N as true and negated signal (U_{a0}, $\overline{U_{a0}}$)
Maximum output frequency:	1 MHz
Phase shift, track A to B:	90° ± 30°
Power consumption:	Not more than 300 mA

Absolute Encoders (SSI)

Transfer procedure:	Synchronous-serial interface (SSI) with 5 V differential-signal transfer signals (such as RS422 standard)
Output signals:	Data as true and negated signal
Input signals:	Sliding pulse as true and negated signal
Resolution:	Not more than 25 bits
Maximum transfer frequency:	1.25 Mbps
Power consumption:	Not more than 300 mA

Encoder power supply

The 5 V or 24 V power supply to the encoders is generated within the module and is available on the female sub-D connector, and so you can power the encoders by way of the connecting cable, without additional wiring. The available voltage is electronically protected against shorting and thermal overload, and is monitored.

Table 4-6 Electrical parameters of encoder power supply

Parameters	Min	Max	Unit
5 V power supply			
Voltage	5,1	5,3	V
Ripple		50	mV_{ss}
Current carrying capacity		0,3	A
24 V power supply			
Voltage	20,4	28,8	V
Ripple		3,6	V_{ss}
Current carrying capacity		0,3	A

Connecting cables to encoder

The maximum cable length depends on the specifications of the encoder power supply, and on the transfer frequency. For trouble-free operation, you should not exceed the following values when using SIEMENS cable sets:

Table 4-7 Maximum cable length as a function of encoder power supply

Supply voltage	Power consumption	Max. cable length
5 V DC	$\leq$ 300 mA	25 m (82 ft)
5 V DC	$\leq$ 220 mA	35 m (115 ft)
24 V DC	$\leq$ 300 mA	100 m (328 ft)

Note

If you want to use incremental encoders with cable lengths longer than 25 or 35 m (82 or 115 ft), select a type that uses a 24 V power supply.

Table 4-8 Maximum cable length as a function of transfer frequency

Encoder type	Frequency	Max. cable length
Incremental encoder	1 MHz	10 m (32.8 ft)
	500 kHz	35 m (115 ft)
Absolute encoder (SSI)	1.25 Mbps	10 m (32.8 ft)
	125 kbps	100 m (328 ft)

4.5 Connecting the encoders

To connect the connecting cables

Please note:

Note

Use only shielded cables. The shielding must be connected to the metallic or metallized connector jacket.

The cable sets supplied as an accessory offer excellent immunity from interference, as well as cross-sections large enough for the power supply to the encoders.

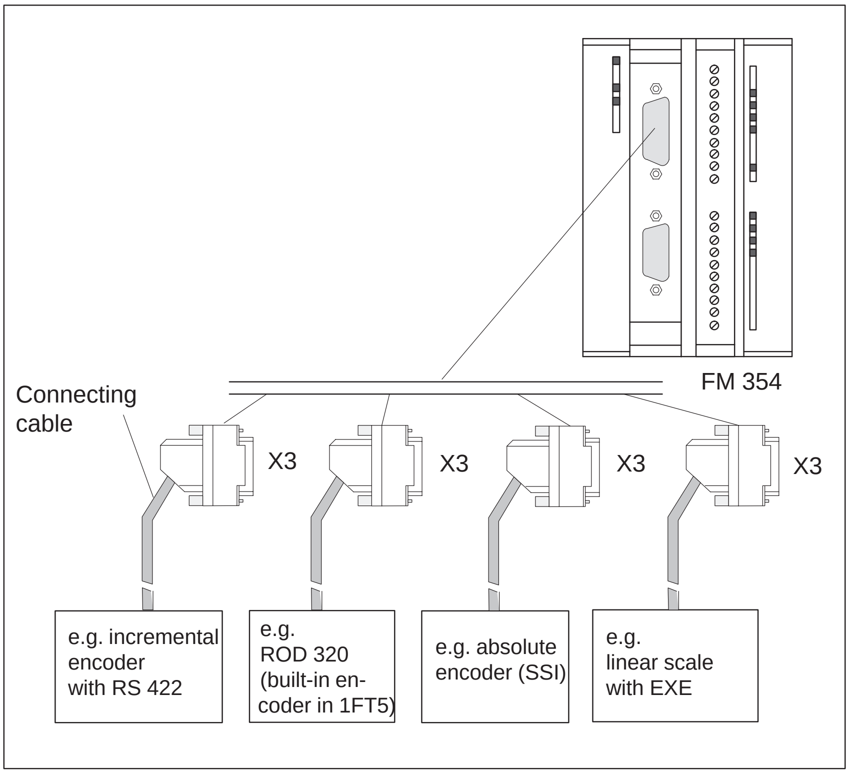


Fig. 4-5 Connecting the encoders

Procedure for connecting encoders

To connect the encoders:

1. Connect the connecting cables to the encoders.

   For absolute encoders (SSI) it may be necessary to cut and add connectors to the cable (end of the cable to the encoder) according to the manufacturer's instructions.

2. Open the front door and plug the sub-D connector into the module.

3. Lock the connector in place with the knurled screws. Close the front door.

Available connecting cables for encoders

Cable set for add-on encoders or EXEs (for connection of linear scales)

Order No.: 6FX2 002-2CD01-1☐☐0

Cable set for built-in encoders with 17-pin round plugs.

Order No.: 6FX2 002-2CE01-1☐☐0

Cable set for absolute encoders (SSI) with a free cable end.

Order No.: 6FX2 002-2CC01-1☐☐0

Connecting cables are available in a variety of lengths.

see *Catalog NC Z* , Order No.: E86060-K4490-A001-A4.

4.6 Description of the I/O interface

Front connector

Four digital input/output modules and the standby signal (controller message) may be connected to the 20-pin front connector X1 with its single-wire terminal.

Location of connector

Figure 4-6 shows the front connector in position to be wired, and the labeling on the inside of the front door.

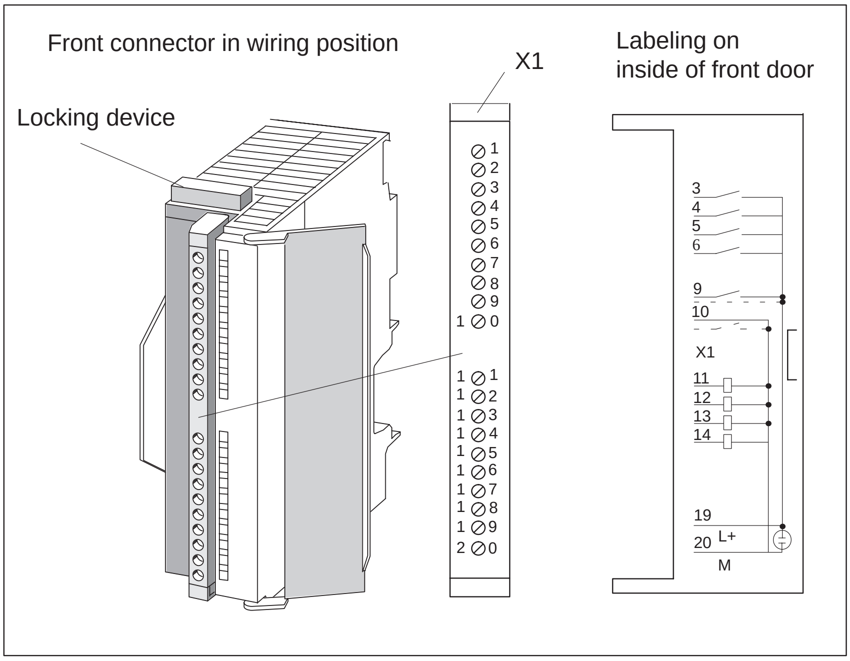


Fig. 4-6 Location of X1 connector

Connector pinout

Connector identifier: **X1**
Connector type: 20-pin S7 front connector for single-wire terminal

Table 4-9 Pinout of the X1 connector

Pin	Name	Type	Pin	Name	Type
1	open		11	DA1	O
2	open		12	DA2	O
3	DI1	I	13	DA3	O
4	DI2	I	14	DA4	O
5	DI3	I	15	open	
6	DI4	I	16	open	
7	open		17	open	
8	open		18	open	
9	RM_P	I	19	L+	VI
10	RM_N	I	20	M	VI

Signal names

DI1...4 Digital input 1...4
DQ1...4 Digital output 1...4
RM_P Positive input for controller message
RM_N Negative input for controller message
L+, M 24 V load power supply / ground

Signal type

O Output
I Input
VI Voltage input

4 digital inputs (DI1...4)

All inputs have equal priority. Switching functions are allocated to an input number by way of machine data; input polarity is selected in the same way (starting and shutdown slopes).

These fast inputs are PLC-compatible (24 V current-sourcing). Switches or contactless sensors (2-wire or 3-wire sensors) can be connected.

Possible uses include:

- As reference-point switches
- As switches for external Start/Stop, external block change
- As touch probes

See Section 5.3.1 for further applications.

Table 4-10 Electrical parameters of digital inputs

Parameters	Value	Unit	Notes
1 signal, voltage range	11...30	V	
1 signal, power consumption	6...15	mA	
0 signal, voltage range	–3...5	V	or input open
Signal delay 0 → 1	15	µs	
Signal delay 1 → 0	150	µs	

“Controller message” (RM) input

The standby signal of the drive power section (controller message) can be connected to a further input.

Note

The “controller message” input is configured as an isolated optical coupler input. This allows both a current-sourcing and a current-sinking output of the power section to be connected. See Section 4.7for details about wiring.

Table 4-11 Electrical parameters of “controller message” input

Parameters	Value	Unit	Notes
1 signal, voltage range	15...30	V	
1 signal, power consumption	2...6	mA	
0 signal, voltage range	–3...5	V	or input open
Signal delay 0 → 1	30	µs	
Signal delay 1 → 0	150	µs	

There are two ways of powering the standby signal:

- from the controller
- from the drive unit

Power from the open-loop control

Figure 4-7 shows examples of how to power the standby signal from the controller (e.g. SIMODRIVE 611 drive unit).

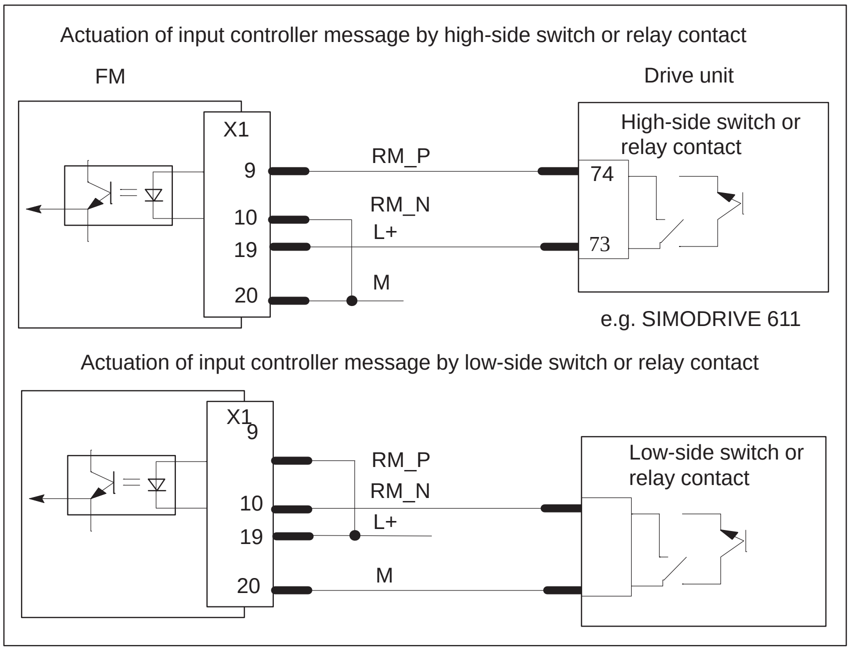


Fig. 4-7 Actuation of the input controller message, power supply from the control

Power from the drive unit

Figure 4-8 shows examples of how to power the standby signal from the drive unit.

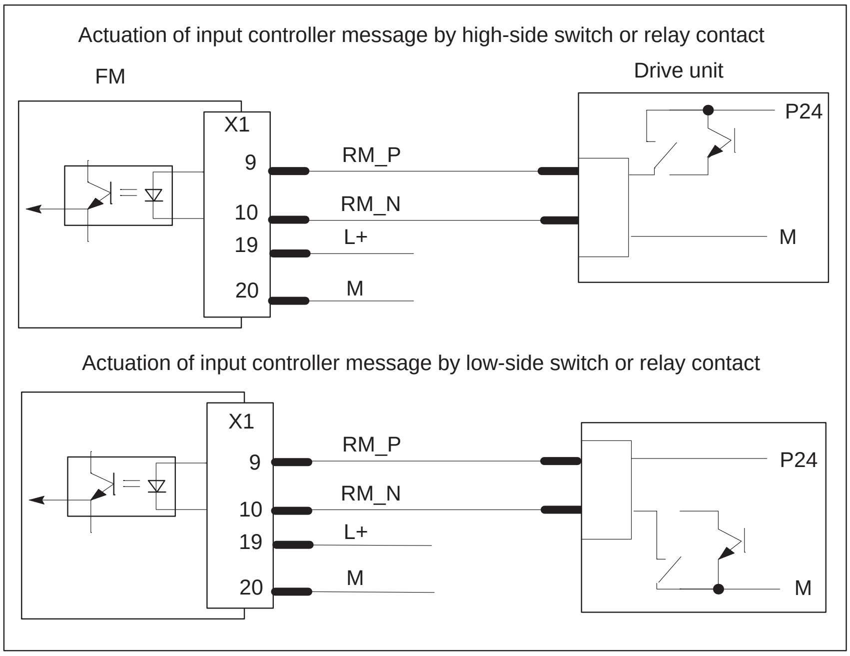


Fig. 4-8 Actuation of the input controller message, power supply from the drive unit

4 digital outputs (DO1...4)

All outputs have equal priority. Switching functions are allocated to an output number by way of machine data.

These four outputs are intended for wiring of application-specific signals.

Possible uses include:

- Position reached and stopped
- Switching function M command
- Forward/backward rotation

See Section 5.3.1 for further applications.

Table 4-12 Electrical parameters of digital outputs

Supply voltage	24 V DC (allowable range: 20.4...28.8 V)
Electrical isolation	No
Output voltage	• 0 Signal: Residual current max. 2 mA • 1 Signal: (Power supply –3 V)
Output current on signal “1” • at ambient temperature of 40°C – Rated value – Permissible value range – Lamp load • at ambient temperature of 60°C – Rated value – Permissible value range	 0.5 A (total current 2 A) 5 mA...0.6 A (over power supply range) max. 5 W 0.1 A (total current 0.4 A) 5 mA...0.12 A (over power supply)
Switching rate	• Resistive load: max. 100 Hz • Inductive load: 0.25 Hz

Load power supply (L+, M)

Here a 24-V load power supply must be connected. The unit will not operate if the poles are reversed.

Note

Note the SIMATIC setup guidelines. In particular, the M terminal (reference potential) must be connected with the chassis ground of the programmable controller (M terminal on the terminal block of the S7-300 CPU).

see manual *S7–300 Programmable Controller, Hardware and Installation*, Order No.: 6ES7 030-0AA01-8AA0.

4.7 Wiring up the front connector

Wiring the front connector

Figure 4-9 shows you how to install the conductors on the front connector and how to relieve the strain with the shield connection element.

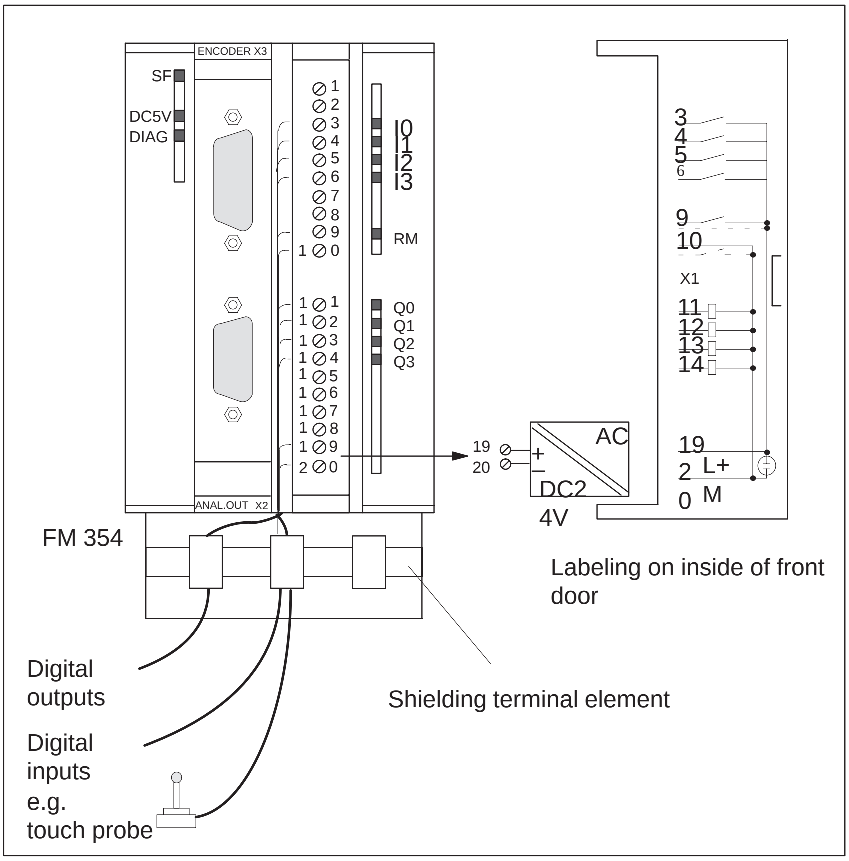


Fig. 4-9 Wiring up the front connector

Connecting cables

Flexible conductor, cross–sectional area 0.25...1.5 mm^2

Ferrules are not necessary.

You can use ferrules without insulating collars per DIN 46228, Shape A, long configuration.

You can connect two lines measuring 0.25...0.75 mm^2 in a single ferrule.

Note

To provide optimum immunity to interference, shielded cables should be used to connect touch probes or sensors.

Tools required

A 3.5 mm (.13 inches) screwdriver or power screwdriver.

Procedure for wiring the front connector

To wire the terminal strip:

1. Strip 6 mm (.23 inches) of insulation from the cable; apply ferrules, if any.
2. Open the front door. Move the front connector into position for wiring (while pressing down on the locking elements as shown in Figure 4-6).

   Lock the connector in place without any electrical contact to the module.
3. Apply the cable grip to the connector.
4. If you are leading the lines out from below, start wiring from the bottom; otherwise start from the top. Screw down unused terminals as well.

   The tightening torque should be 60-80 Nm.
5. Tighten the cable grip on the cable strand.
6. Move the front connector into operating position (while pressing down on the locking elements).
7. You can fill out the labeling strip enclosed with the product and slip it into the front panel.

Shielded cables

When using shielded cables, the following additional steps are necessary:

1. After leading the cable into the cabinet, connect the cable shielding with a grounded shielding bus (strip the insulation from the cable itself).

   For this you can use the shielding terminal element mounted on the DIN rail; it will accept up to eight shielding terminals.

   see manual *S7-300 Programmable Controller, Hardware and Installation*, Order No.: 6ES7 030-0AA01-8AA0.
2. Connect the shielded line to the module, but do not connect the shielding there.

Shielding terminal element

To provide a shielding end lead for shielded cables, this element can be inserted in the DIN rail. It can accept up to eight shielding terminals (KLBÜ line from Weidmüller).

Order No.: Terminal element: 6ES7 390-5AA00-0AA0
Shielding terminal: 6ES7 390-5CA00-7AA0

see *Catalog NC 60.1*, Order No. E86060-K4460-A101-A3

see *Catalog ST 70*, Order No. E86060-K4670-A101-A2

■

Defining Parameters of the FM 354 **5**

Summary

This chapter gives you an overview of how to define the parameters of the FM 354 with the “Parameterize FM 354” tool.

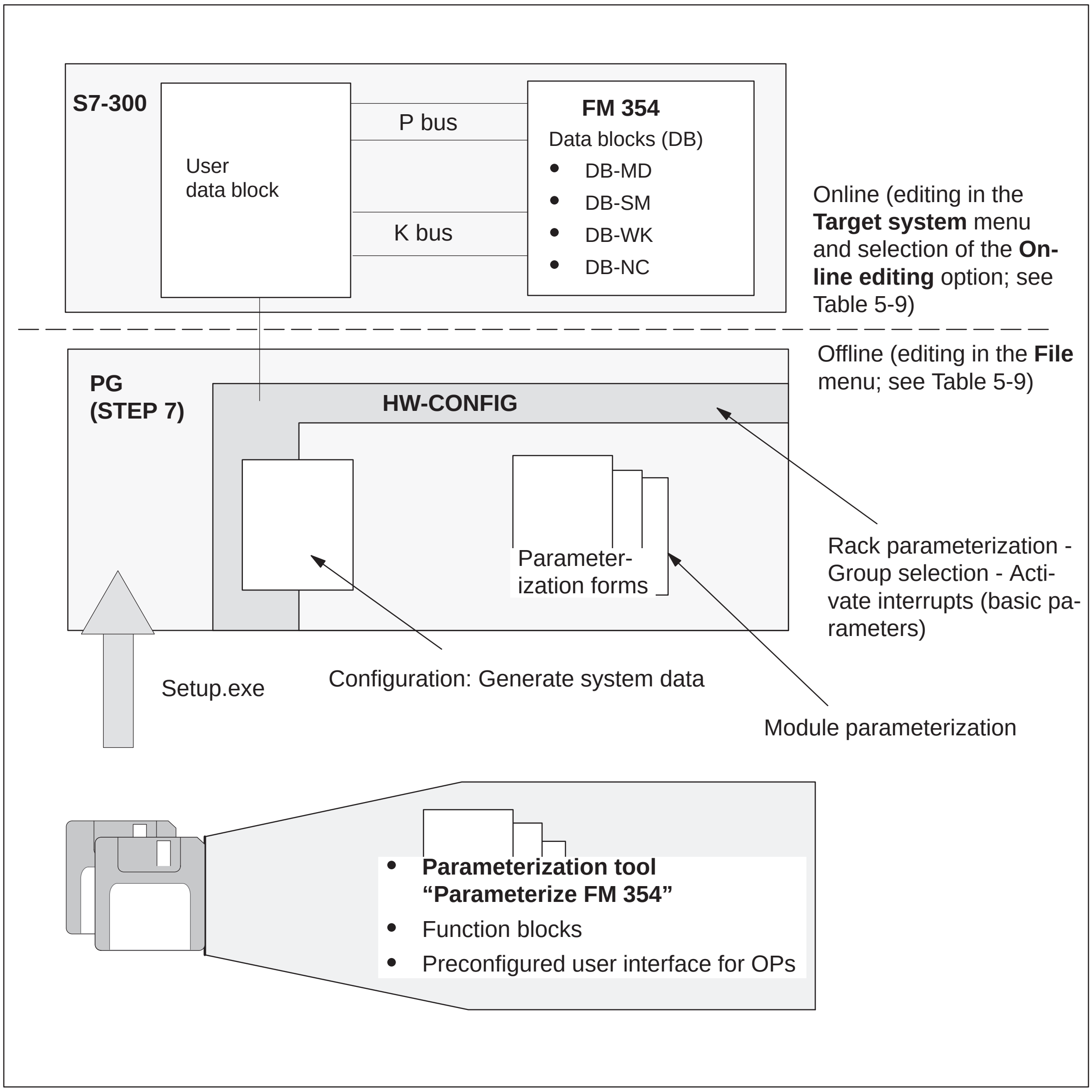


Fig. 5-1 Overview of parameterization

Chapter overview

5.1 Installing “Parameterize FM 354”

Prerequisites

The Windows 95 operating system and appropriate STEP 7 program (V3.1 or higher) must already be installed on the programming device/PC.

For online operation, the link between the PG and the S7-300 CPU must already be set up (see Figure 4-1). For distributed use of the FM (under development), a link from the programming device/PC to the L2-DP network must already be set up.

Installation

The entire software (parameterization tool, function blocks and preconfigured user interface for OPs) is stored on two 3.5–inch diskettes and is installed complete.

Install the software as follows:

1. Insert diskette 1 in the floppy disk drive of your programming device/PC.
2. In Windows 95, start the interactive routine for installing the software by double–clicking the “Software” symbol in the Control Panel.
3. Select the floppy disk drive and the file **Setup.exe** in the dialog box, and start the installation program.
4. Follow the instructions displayed by the installation program step by step.

   Result: The software is installed in the following directories:

   – “Parameterize FM 354” parameterization tool: **SIEMENS\STEP7\S7FLAG**
   – Function blocks: **SIEMENS\STEP7\S7LIBS\FMST_SRV**
   – User interface for OPs: **SIEMENS\STEP7\EXAMPLES\S7OP_BSP**
   – Example applications: **SIEMENS\STEP7\EXAMPLE1\FMSTSVEX**

Note

If you chose a directory other than **SIEMENS/STEP7** when you installed **STEP 7**, this directory is entered instead.

5.2 Getting started with “Parameterize FM 354”

Prerequisites

You have installed the software on your programming device/PC, as described in Section 5.1.

Configuration

Before you can configure your system, you must create a project in which to save the parameters. You will find further information on how to configure modules in your user manual *Standard Software for S7 and M7, STEP 7*. The description below outlines only the most important steps.

1. Start the *SIMATIC Manager* and open your project.
2. Insert a **SIMATIC 300 station** in the menu **Insert** ▸ **Station**.
3. Select the **SIMATIC 300 station**. Call up the S7 hardware configuration from the menu **Edit** ▸ **Open Object**.
4. Select a rack and assign it.
5. Select the FM 354 positioning module with the correct order number from the module catalog, and insert it in the hardware table as appropriate for your configuration.
6. Double–click a module to configure it.

   The **Properties** dialog box appears.

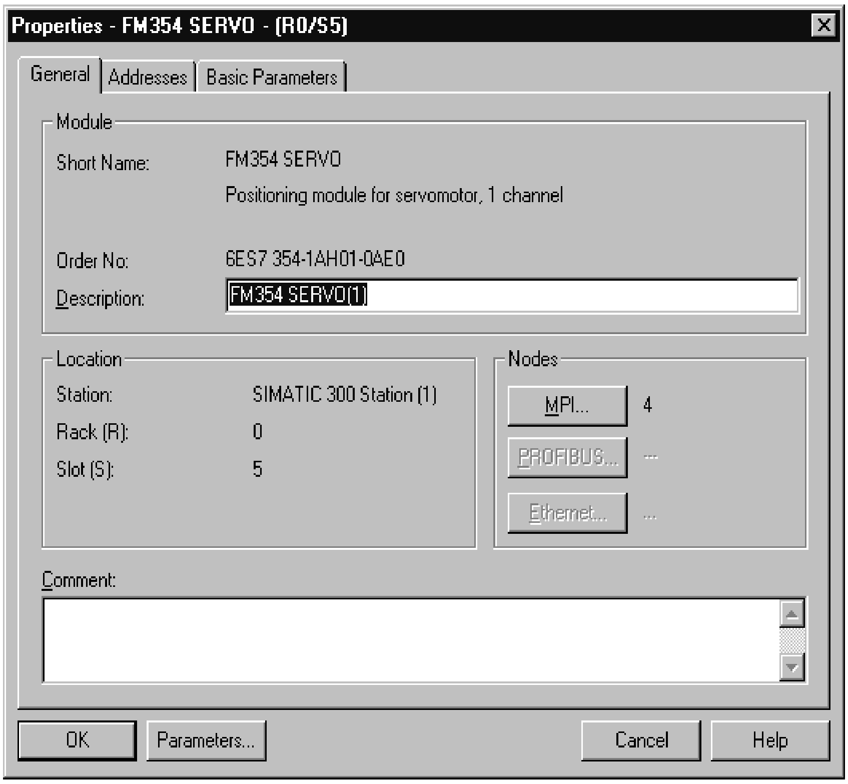


Fig. 5-2 Getting started with “Parameterize FM 354”

7. By clicking the tabs in this window (General, Addresses and Basic Parameters), you can
   – Name the FM 354
   – Change the address of the FM 354
   – Configure the interrupts.

Note:

Further operation of the FM 354 is not possible with the CPU in the STOP state.

Click the **Parameters** button to call up the screen for setting the parameters.

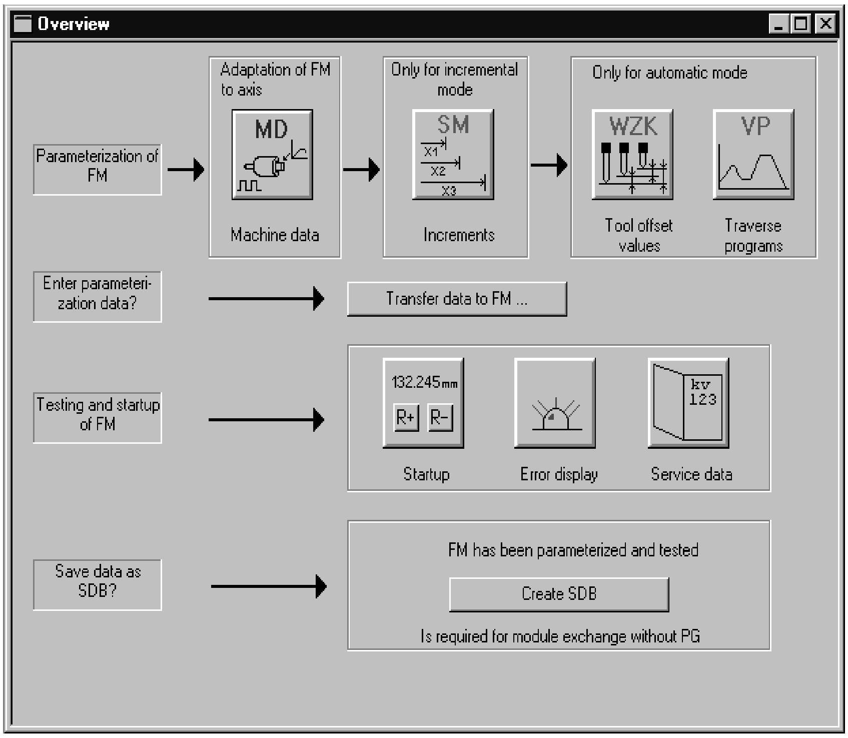


Fig. 5-3 Overview display for parameterization

You can return to this display at any point during parameterization by selecting the menu **View** ▸ **Overview**.

The FM 354 module for servo–controlled positioning is parameterized by way of parameter DBs that reside in memory on the module. Here a key function is performed by the “Machine data” data block (DB-MD), since it is always needed, regardless of what technological function the module performs. All other parameter DBs are only needed as a function of the technology involved.

You can now set the parameters of your module. This chapter gives you an overview of the parameters that can be set.

You can use the mouse to change the size of the window for entering the parameter data and the size of the overview display.

Proceed as follows:

1. Position the mouse pointer on the top border of the window, so that it changes into an arrow.
2. Press the left mouse button, and drag the pointer downwards by moving the mouse.
3. Release the mouse button.
4. Position the mouse pointer on the bar with the name of the window.
5. Press the left mouse button, and drag the pointer upwards by moving the mouse. When you have moved the window to the correct position, release the mouse button.

When you have configured your project, you can call up the **Properties** screen in S7 Configuration by selecting the module and activating the menu command **Edit ▸ Object Properties**.

Note

Once you have imported a project from STEP 7 V2.1, you cannot store any parameter data (data blocks) in this project. You can only read the parameter data in this project by importing the data from a file, or store the data by exporting the data to a file.

Suggestion!

1. Create a new project in STEP 7 V3.1.
2. Copy all programs (*.awl) into the new project.

Integrated help

The parameterization user interface has an integrated help system to support you when you set the parameters of the positioning module. To call up the integrated help:

- Select the menu command **Help ▸ Help Topics ...** or
- press the **F1** key or
- select the symbol  and then move to the element or window you want information about and press the left mouse button.

5.3 Parameter data

What can I parameterize?

You can parameterize the following data storage areas:

- Machine data (MD)
- Increment sizes (SM)
- Tool offset data (TO)
- Traversing programs (NC)
- User data (user data blocks)

This data is stored in data blocks (DBs) within the numerical range 1001 to 1239 (not including user data).

The MD, SM, TO and NC data blocks are transferred to the FM 354 and reside in memory there.

Parameterization of SM, TO and NC may be omitted if the associated functions are not used.

The user data block must be stored in the CPU. Only then can it be filled with data online (see Chapter 6).

Parameterization data (except for user data) can also be created, edited and saved offline on the PU.

Data blocks (DB) of the FM 354

Table 5-1 gives you an overview of the data blocks in the FM 354 and their meaning.

Table 5-1 Data blocks

Data block	Significance
DB-MD	**Machine data (DB No. = 1200)** **Block size (rounded in bytes) = 250** Machine data serves to adapt the FM 354 to the user's own specific application. Parameterization with machine data is essential in order for the FM's functions to be activated. The parameterized DB-MD should be loaded to the FM. As it is written to the FM 354, the DB-MD is checked for the input limits of the individual values and their interdependencies. It is then stored only if all values are allowed. Otherwise data error messages are displayed by way of the MPI. A defective DB will not be retained when the power is turned off. The machine data can then be activated by way of “Activate machine data” or by switching the equipment on and off.

Table 5-1 Data blocks, continued

Data block	Significance
DB-SM	**Increments (DB No. = 1230)** **Block size (rounded in bytes) = 460** Increments serve in the “Relative incremental” operating mode as user-definable relative path distances for individual positioning. You can define from 1 to 100 increment sizes (see Section 5.3.2). Modifications can be made in all operating modes (even in “Incremental relative” mode) during movement. The modifications of the increments must always be complete before a new movement is started in “Incremental relative” mode. If this is not the case, the error message “incremental dimensions do not exist” is output Cl. 2/No. 13.
DB-WK	**Tool offset data (DB No. = 1220)** **Block size (rounded in bytes) = 310** The use of tool length compensation and wear values is described in Section 10.1. Up to 20 compensation or wear values are available. Tool offset data are required for the “Automatic and Automatic single block” modes. Modifications can be made in all operating modes and during movement. If modifications are made during starting or at block transitions when the tool compensation is active (internal access to offset values), the error message “tool offset value does not exist” is output Cl.3/No.35.
DB-NC	**Traversing programs (Program No. + 1000 = DB No. = 1001...1199)** **Block size (rounded in bytes) = 110 + (20 x no. of traversing blocks)** Traversing programs are required for the “Automatic and Automatic single block” modes. • Programs which are not selected can always be modified. • If modifications are made to a preselected program, including the subprogram, preselection of the program is canceled. You must then select the program again. A modification can be made to a program when BL = 0 (start of program/end of program) and on Stop.
System data block SDB $\geq$ 1 000	**For module replacement without PG** All the parameter data of the FM 354 (DB-MD, DB-SM, DB-WK, DB-NC) are stored in SDB $\geq$ 1 000. This SDB is loaded into the CPU and is used as an additional means of data storage.
DB-SS	**Data block for status messages (DB no. 1000)** The DB-SS is an internal DB on the FM for testing, start–up and operator control and monitoring.
DB 1249	Internal DB on the FM, not relevant for user.

User data block

Chapter 6 describes how to generate a user data block.

You can use “Parameterize FM 354” to fill the user DB with the data described in Table 5-2.

The menu **Target system** ▶ **Online editing** ▶ **User data** allows you to select and edit your user DB.

Table 5-2 User DB

Data block	Significance
User DB	For the structure and data formats see Chapter 6 You can preload the following data to the DB provided the DB itself has been loaded to the CPU: • Module address[1)] • Channel address[1)] • Channel offset Reserved • Zero offset • Set actual value • Set actual value on-the-fly • Set reference point • Setpoint for increment • Speed Level 1 • Speed Level 2 • Voltage Level 1 • Voltage Level 2 • MDI block • MDI block on the fly • Program selection, program number • Program selection, block number • Program selection, working direction • Code application data 1 • Code application data 2 • Code application data 3 • Code application data 4

1) You can only view these data. The data are edited by the FC INIT_DB (see Chapter 6).

Data block structure

Table 5-3 gives a rough picture of data block structure.

Table 5-3 Data block structure

Addresses/Offset	Contents	Comment
	DB header	System information, not relevant for user
0 and above	User data area / structure header	Information for labeling of data block within the system
24 and above for MD, otherwise 32	User data	Parameterization data

Detailed data block structures and parameterization data for the individual types of data blocks can be found in the following sections.

5.3.1 Machine data

DB structure

Table 5-4 gives you an overview of the structure of the “machine data” data block (DB-MD).

Table 5-4 DB structure – Machine data

Byte	Variable type	Value	Significance of the variables	Comment
			DB header	
0	WORD		Rack slot	Module address
2	WORD		DB No. (≥ 1000)	As in DB header
4	DWORD		reserviert	
8	WORD		Error No. (from FM)	With HMI services
10	WORD	1	Channel number	
12	2 STRING	MD	DB identifier/type	2 ASCII characters
16	DWORD	354	Module identifier	FM 354
20	4 CHAR	0	Version number/block number	(DB structure)
24 and above...			See machine data list MD5...MD45	

Entering values

In “Parameterize FM 354” select the menu **File ▸ New ▸ Machine Data** to call up the following display.

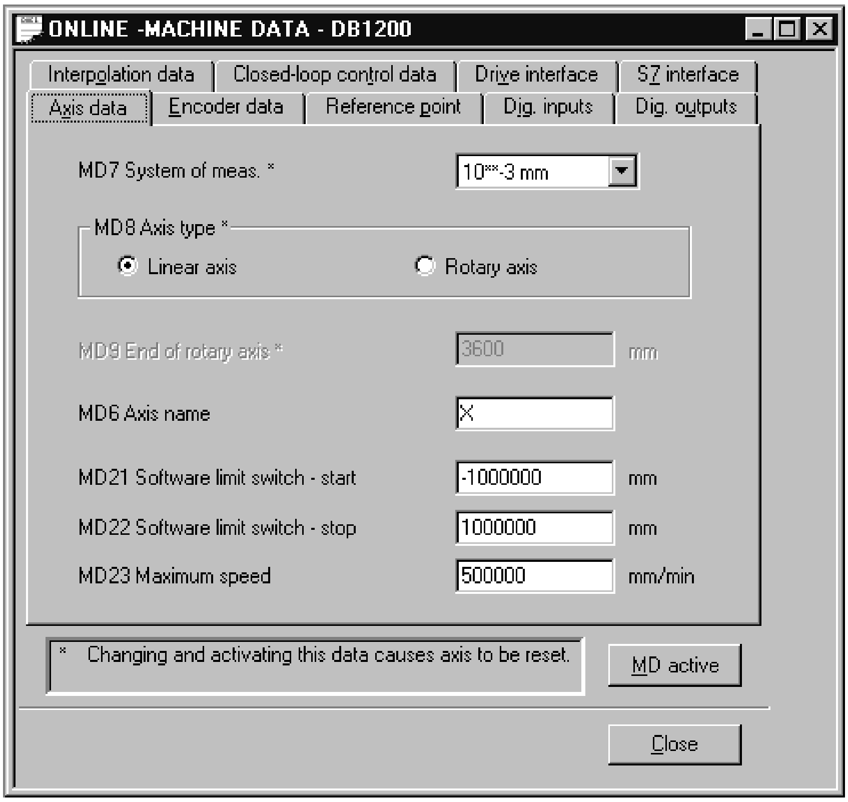


Fig. 5-4 Entering values for machine data

Enter the machine data in the tab windows.

You can also enter your values in a table by selecting **View▸ Table form**.

When creating the MD DBs you must follow the instructions in Section 7 “Starting up the FM 354”.

Note

The measurement system (MD7) must match the measurement system specified in the other DBs.

The measurement system raster (MSR) is the smallest distance unit in the active system of measurement.

If at some point you have failed to take this precaution:

1. Delete all data blocks (which do not match the measurement system) or clear the memory of the FM 354 completely.
2. Modify the other data blocks on the PG.
3. Reload the data blocks to the FM 354.

Machine data list

All machine data of the FM 354 are listed in Table 5-5.

Notes to the machine data list:

K stands for configuration data: see Section9.3.3

E stands for user-definable machine data settings for readjustment (startup optimization) and technology; see Section 9.3.3

The units of measurement refer to the value representation in the machine data DB.

Table 5-5 Machine data list

No.	Designation	Default values	Value/Meaning	Data type/ Unit/Comments	See Section
1...4				open	
5 E	Process interrupt generation	0	0 = Position reached 1 = Length measurement completed 3 = Change block on-the-fly 4 = Inprocess Measurement	BITFIELD32	9.10
6	Axis name	X	max. 2 ASCII characters1)	4 bytes	
7 K	System of measurement	1	1 = 10^{-3} mm 2 = 10^{-4} inch 3 = 10^{-4} degrees 4 = 10^{-2} degrees	DWORD (MSR)	9.4
8 K	Axis type	0	0 = linear axis 1 = rotary axis	DWORD	9.5
9 K	Rotary axis end2)	$36 \cdot 10^5$	0...1 000 000 000	DWORD (MSR)	
10 K	Encoder type	1	0 = not present 1 = incremental encoder 3 = absolute encoder (SSI, 13-bit) 4 = absolute encoder (SSI, 25-bit) 13 = absolute encoder (SSI, 13-bit) 14 = absolute encoder (SSI, 25-bit)	DWORD GRAY Code GRAY Code Binary Code Binary Code	9.6.1 9.6.2
11 K	Travel per motor revolution (division period)2	10 000	1...1,000,000 ,000	DWORD (MSR) (integer component)	
12 K	Residual distance per encoder revolution (division period)2	0	0...2^{32}–1	DWORD (2^{-32}MSR) (fractional component)	
13 K	Increments per encoder revolution (division period)2	2 500	2^1...2^{25}	DWORD With incremental encoders, evaluation takes place at $4 \cdot$ MD.	
14 K	Number of rotations – absolute encoder	0	0/1 = single-turn encoders 2^1...2^{12} for multi-turn encoders	DWORD Only powers of two are allowed.	

MSR = measurement-system grid RPS = reference-point switch

1) The variable axis name is implemented as an axis letter (X, Y, Z, ...) with an address extension (1...9).
Permissible characters: X, Y, Z, A, B, C, U, V, W, Q, E, 1...9 e.g.: “X”, “X1”

2) see Dependencies

Table 5-5 Machine data list, continued

No.	Designation	Default values	Value/Meaning	Data type/ Unit/Comments	See Section
15 K	Baud rate – absolute encoder	2	1 = 78,000 2 = 156,000 3 = 312,000 4 = 625,000 5 = 1,250,000	DWORD	9.6.1 9.6.2
16 K	Reference-point coordinate	0	–1,000,000,000...+1,000,000,000	DINT (MSR)	9.2.3
17 K	Absolute-encoder readjustment	0	0...$2^{25}-1$	DWORD (Encoder grid) absolute encoder	9.6.3
18 K	Type of reference-point approach (reference-point approach direction)	0	0 = direction +, zero pulse right 1 = direction +, zero pulse left 2 = direction –, zero pulse right 3 = direction –, zero pulse left 4 = direction +, RPS center 5 = direction –, RPS center 8 = direction +, RPS edge 9 = direction –, RPS edge	DWORD Code identifies position for synchronization point with ref. to RPS for incremental encoders only	9.2.3
19 K	Direction adjustment	0	0 = invert direction of measurand 1 = invert analog value	BITFIELD32	9.7
20 K	Hardware monitoring	0	0 = cable break (incremental encoder) 1 = error, absolute encoder 2 = pulse monitoring (incremental encoder) 3 = Voltage monitoring - encoder	BITFIELD32	9.6.1 9.6.2
21 E	Software limit switches, beginning[2)]	-10^9	–1 000 000 000...1 000 000 000	DINT (MSR)	9.7 9.9
22 E	Software limit switch – end[2)]	10^9	–1 000 000 000...1 000 000 000		
23 E	Maximum speed	$30 \cdot 10^6$	10...500,000,000	DWORD (MSR/min)	9.7
24 E	Target range (position reached, stop)	1 000	0...1 000 000	DWORD (MSR)	
25 E	Monitoring time	0	0 = no monitoring 1...100,000	DWORD (ms) rounded to 2-ms steps	
26 E	Stationary range	10^4	1...1 000 000	DWORD (MSR)	
27 E	Reference-point shift	0	–1,000,000,000...+1,000,000,000	DINT (MSR)	9.2.3
28 E	Referencing velocity[2)]	$6 \cdot 10^6$	10...500,000,000	DWORD (MSR/min)	9.2.3
29 E	Reducing velocity[2)]	$3 \cdot 10^6$	10...500,000,000		

MSR = measurement-system grid   RPS = reference-point switch

1) The variable axis name is implemented as an axis letter (X, Y, Z, ...) with an address extension (1...9).
Permissible characters: X, Y, Z, A, B, C, U, V, W, Q, E, 1...9   e.g.: "X", "X1"

2) see Dependencies

Table 5-5 Machine data list, continued

No.	Designation	Default values	Value/Meaning	Data type/ Unit/Comments	See Section
30 E	Backlash compensa-tion	0	–1 000 000...+1 000 000	DINT (MSR)	9.7
31 E	Directional reference of backlash	0	0 = as in search for reference (not for absolute encoders) 1 = positive 2 = negative	DWORD	9.7
32 K	M-function output type	1	**during positioning:** 1 = time-controlled 2 = acknowledgment-controlled **before positioning:** 3 = time-controlled 4 = acknowledgment-controlled **after positioning:** 5 = time-controlled 6 = acknowledgment-controlled	DWORD serial output of up to 3 M functions in NC block	10.3 9.1
33 K	M-function output time	10	1...100,000	DWORD (ms) rounded to 2-ms steps	
34 K	Digital inputs2)	0	0 = external start 1 = input for enable 2 = external block change 3 = set actual value on-the-fly 4 = measure 5 = RPS for search for reference 6 = reversing switch for search for reference	BITFIELD32 bit-coded function al-location: Bit No. I/O 0 Bit No. + 8 I/O 1 Bit No. + 16 I/O 2 Bit No. + 24 I/O 3 Front edge always activates the function	9.2.3 9.8
35 K	Digital outputs2)	0	0 = Position reached, stop 1 = Axis movement forward 2 = Axis movement reverse 3 = Change M97 4 = Change M98 5 = Enable Start 7 = Direct output		9.8

MSR = measurement-system grid RPS = reference-point switch

1) The variable axis name is implemented as an axis letter (X, Y, Z, ...) with an address extension (1...9).
Permissible characters: X, Y, Z, A, B, C, U, V, W, Q, E, 1...9 **e.g.: “X”, “X1”**

2) see Dependencies

Table 5-5 Machine data list, continued

No.	Designation	Default values	Value/Meaning	Data type/ Unit/Comments	See Section
36 K	Input adjustment (signal processing inverted)	0	8 = I0 inverted 9 = I1 inverted 10 = I2 inverted 11 = I3 inverted	BITFIELD32	9.8
37 K	Servo control signals	1	0 = controller enable active 2 = controller ready active 3 = controller ready inverted 7 = time override active 15 = continue operation after emergency stop (drive enable [AF]) 16 = automatic drift compensation active		9.7 9.1.1
38 E	Positioning loop amplification	1 000	1...10,000	DWORD ((MSR/min)/MSR)	
39 E	Minimum following error, dynamic	0	0 = no monitoring 1...1 000 000	DWORD (MSR)	
40 E	Acceleration	1 000	0 = without ramp 1...100,000	DWORD (10^3MSR/ s^2)	9.7
41 E	Deceleration	1 000			
42 E	Jolt time	0	0...10,000	DWORD (ms)	
43 E	Set voltage, max.	8 000	1,000...10,000	DWORD (mV)	
44 E	Offset compensation	0	–1,000...+1,000	DINT (mV)	
45 E	Voltage ramp	0	0...10,000,000	DWORD (mV/s)	

MSR = measurement-system grid RPS = reference-point switch

1) The variable axis name is implemented as an axis letter (X, Y, Z, ...) with an address extension (1...9).
Permissible characters: X, Y, Z, A, B, C, U, V, W, Q, E, 1...9 **e.g.: “X”, “X1”**

2) see Dependencies

Dependencies

With certain combinations of machine data, restrictions in the value range arise for non–processing of the machine data.

These dependencies are verified on acceptance of the MD DB or individual machine data, and an error message is output in the event of a violation. Some checks are performed on the basis of internally calculated reference variables.

These reference variables and the dependency checks are described in the tables below.

Reference variables generated internally from MD:

Generation of travel per encoder revolution **UMWEG**

UMWEG = MD11 + MD12 · 2^{-32}

Generation of internal measured value factor **MWFAKTOR**

MD10	Measured value factor
0	MWFAKTOR = 1
1	MWFAKTOR = UMWEG / (4 · MD13)
3, 4, 13, 14	MWFAKTOR = UMWEG / MD13

Activation of software limit switches **SEAKT**

<table>
<tr><th>MD21</th><th>MD22</th><th>SEAKT</th></tr>
<tr><td>$= -10^9$</td><td>$= +10^9$</td><td>0 (inactive)</td></tr>
<tr><td>$\neq -10^9$</td><td>$= +10^9$</td><td rowspan="3">1 (active)</td></tr>
<tr><td>$= -10^9$</td><td>$\neq +10^9$</td></tr>
<tr><td>$\neq -10^9$</td><td>$\neq +10^9$</td></tr>
</table>

Internal generation of absolute traversing range limits **VFBABS**

MWFAKTOR	VFBABS
< 1	10^9
≥ 1	10^9 / MWFAKTOR

Verification:

MD9 check

<table>
<tr><th>MD8</th><th>MD10</th><th>MD18</th><th colspan="2">Permissible rotary axis end</th></tr>
<tr><td>0</td><td>–</td><td>–</td><td colspan="2">–</td></tr>
<tr><td rowspan="5">1</td><td>0</td><td></td><td>–</td><td rowspan="5">(MD23/30 000) · $\leq$ MD9 $\leq$ VFBABS</td></tr>
<tr><td rowspan="2">1</td><td>≥ 4</td><td>–</td></tr>
<tr><td>< 4</td><td>MD9 mod UMWEG == 0</td></tr>
<tr><td>3, 13</td><td>–</td><td>UMWEG mod MD9 == 0</td></tr>
<tr><td>4, 14</td><td>–</td><td>(MD14 · UMWEG) mod MD9 == 0</td></tr>
</table>

MD11, MD12, MD13 check → results in MWFAKTOR (see above)

Permissible measured value factor range: 2^{-14} < MWFAKTOR < 2^{14}

MD13 check

MD10	Increments per encoder revolution
0, 1	–
3, 4, 13, 14	2^x x = 1, 2, 3, ...

MD14 check

MD10	No. of revolutions
0, 1, 3, 13	–
4, 14	2^x x = 1, 2, 3, ...

MD21, MD22 check

<table>
<tr><th>SEAKT</th><th>MD8</th><th colspan="3">Permissible software limit switches</th></tr>
<tr><td>0</td><td>–</td><td colspan="3">MD21 = -10^9, MD22 = $+10^9$</td></tr>
<tr><td rowspan="4">1</td><td rowspan="3">0</td><td>MD21 ≥ –VFBABS</td><td>MD10</td><td></td></tr>
<tr><td>MD22 ≤ VFBABS</td><td>0, 1</td><td>–</td></tr>
<tr><td>MD21 < MD22</td><td>3, 13</td><td>MD22–MD21 ≤ UMWEG</td></tr>
<tr><td>1</td><td>0 ≤ MD21 < MD9
0 ≤ MD22 < MD9
MD21 ≠ MD22</td><td>4, 14</td><td>MD22–MD21 ≤ MD14 · UMWEG</td></tr>
</table>

MD28 check

Permissible velocity:	10 ≤ MD28 ≤ MD23

MD29 check

MD10	Permissible velocity:
3, 4, 13, 14	any, not used
0, 1	10 ≤ MD29 ≤ MD23

MD31 check

<table>
<tr><th>MD30</th><th>MD10</th><th>Permissible directional reference of backlash</th></tr>
<tr><td>0</td><td></td><td rowspan="2">–</td></tr>
<tr><td rowspan="2">≠ 0</td><td>0, 1</td></tr>
<tr><td>3, 4, 13, 14</td><td>1, 2</td></tr>
</table>

MD34 check

Permissible:
BYTE0(MD34) ≠ BYTE1(MD34) ≠ BYTE2(MD34) ≠ BYTE3(MD34)

MD35 check

Permissible:
BYTE0(MD35)&0x7F ≠ BYTE1(MD35)&0x7F ≠ BYTE2(MD35)&0x7F ≠ BYTE3(MD35)&0x7F

5.3.2 Increments

DB structure

Table 5-6 gives you a general view of the structure of the “Increments” data block (DB-SM).

Table 5-6 DB structure – increments

Byte	Variable type	Value	Significance of the variables	Comment
			DB header	
0	WORD		Rack slot	Module address
2	WORD		DB No. ($\geq$ 1000)	As in DB header
4	DWORD		Reserved	
8	WORD		Error No. (from FM)	With MMI services
10	WORD	1	Channel number	
12	2 STRING	SM	DB identifier/type	2 ASCII characters
16	DWORD	354	Module identifier	FM 354
20	4 CHAR	0	Version number/block number	(DB structure)
24	DWORD	1...3	Measurement-system grid per MD7	Unit of measurement
28	WORD	0/1	Parameter (DB) backup	Job via MMI
30	WORD		Reserved	
32	DWORD	0...10^9	Increment 1	
36	DWORD	0...10^9	Increment 2 to increment 100	see Section 9.2.4

Input of values

Values are input in the increments menu of the “Parameterize FM 354” parameterization tool.

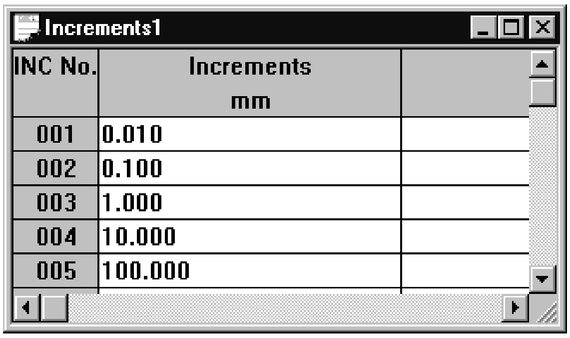


Fig. 5-5 Entering values for incremental dimensions

5.3.3 Tool offset data

DB structure

Table 5-7 gives you a general view of the structure of the “tool offset data” data block (DB-WK).

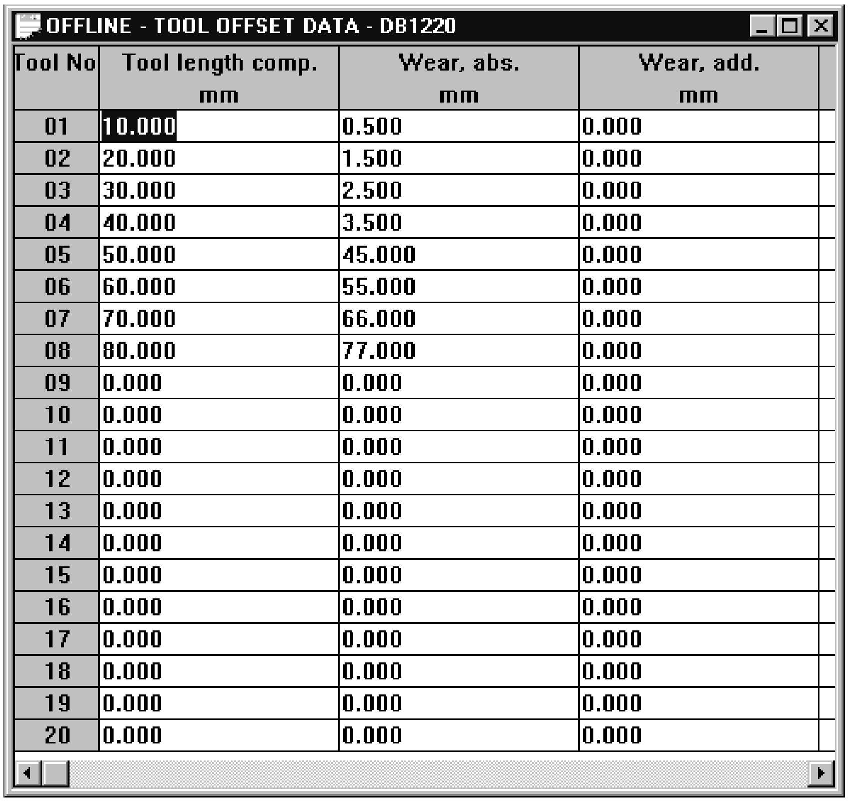
Table 5-7 DB structure – tool offset data

Byte	Variable type	Value	Significance of the variables	Comment
			DB header	
0	WORD		Rack slot	Module address
2	WORD		DB No. (≥ 1000)	As in DB header
4	DWORD		Reserved	
8	WORD		Error No. (from FM)	With MMI services
10	WORD	1	Channel number	
12	2 STRING	TO	DB identifier/type	2 ASCII characters
16	DWORD	354	Module identifier	FM 354
20	4 CHAR	0	Version number/block number	(DB structure)
24	DWORD	1...3	Measurement-system grid per MD7	Unit of measurement
28	WORD	0/1	Parameter (DB) backup	Job via MMI
30	WORD		Reserved	
32	DINT DINT DWORD	$-10^9...10^9$ $-10^9...10^9$ $0...10^9$	Tool length offset 1 Wear value 1 absolute Wear value 1 additive	Tool 1
44	DINT DINT DINT	$-10^9...10^9$ $-10^9...10^9$ $-10^9...10^9$	Tool length offset 2 Wear value 2 absolute Wear value 2 additive to Tool length offset 20 Wear value 20 absolute Wear value 20 additive	Tool 2 to Tool 20 see Section 10.1

Input of values

Values are input in the tool offset data menu of the “Parameterize FM 354” parameterization tool.

If the additive wear value is changed online, the FM calculates the new wear parameter as an absolute value and the additive tool wear is reset to 0.



Tool No	Tool length comp. mm	Wear, abs. mm	Wear, add. mm
01	10.000	0.500	0.000
02	20.000	1.500	0.000
03	30.000	2.500	0.000
04	40.000	3.500	0.000
05	50.000	45.000	0.000
06	60.000	55.000	0.000
07	70.000	66.000	0.000
08	80.000	77.000	0.000
09	0.000	0.000	0.000
10	0.000	0.000	0.000
11	0.000	0.000	0.000
12	0.000	0.000	0.000
13	0.000	0.000	0.000
14	0.000	0.000	0.000
15	0.000	0.000	0.000
16	0.000	0.000	0.000
17	0.000	0.000	0.000
18	0.000	0.000	0.000
19	0.000	0.000	0.000
20	0.000	0.000	0.000

Fig. 5-6 Entering values for tool offset data

5.3.4 Traversing programs

DB structure

Table 5-8 gives you a general view of the structure of the “traversing programs” data block (DB-NC).

Table 5-8 DB structure – traversing programs

Byte	Variable type	Value	Significance of the variables	Comment
			DB header	
0	WORD		Rack slot	Module address
2	WORD		DB No. (≥ 1000)	As in DB header
4	DWORD		Reserved	
8	WORD		Error No. (from FM)	With MMI services
10	WORD	1	Channel number	
12	2 STRING	NC	DB identifier/type	2 ASCII characters
16	DWORD	354	Module identifier	FM 354
20	4 CHAR	0	Version number/block number	(DB structure)
24	DWORD	1...3	Measurement-system grid per MD7	Unit of measurement
28	WORD		Reserved	
30	WORD		Reserved	
32	18 STRING	ASCII characters	NC program name	max. 18 characters
52	STRUCT	NC block	NC block new (modification range)	
72	STRUCT	NC block	1st traversing block	
92	STRUCT	NC block	2nd to 100th traversing block	see Section 9.3.11, 10.1

Input of traversing programs

An empty window is provided for the input of NC traversing programs. Here you can input your traversing program as follows:

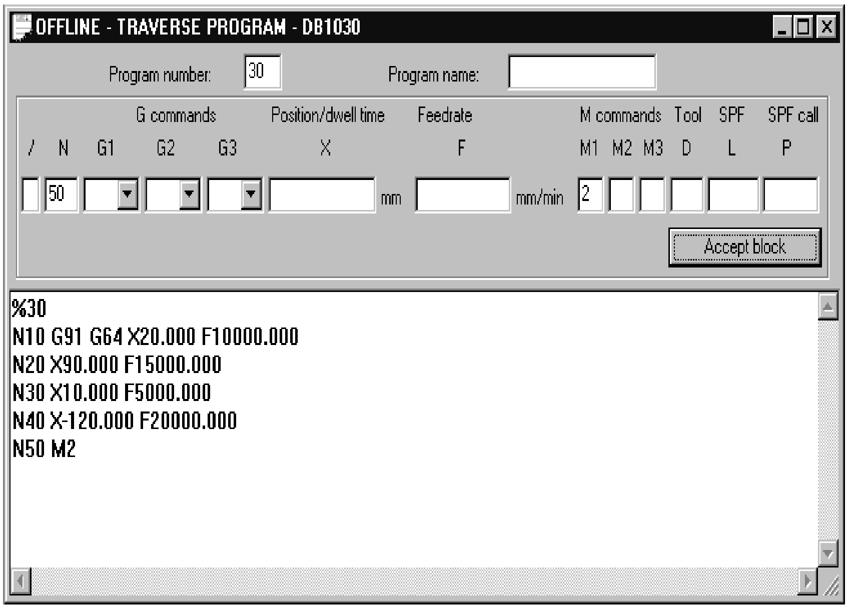


Fig. 5-7 Entry for traversing programs

1. % Program number Program name

   The “%” can be input only in the first line. This input is mandatory. The DB number is formed from the program number.

   The program name is optional and may have up to 18 characters.

2. N<block number> – G<command> (G1, G2, G3) – X<value> – F<value> – M<command> (M1, M2, M3) – D<No.> (tool offset number) – L<No.> – P<No.> ± (for NC programming see Chapter 10).

   – You must enter the block number (N) **first and in ascending order**. The rest of the inputs may be in any desired sequence.

   – Input separators as a blank.

You must enter characters in upper case letters.

You can also use the input area at the top of the screen. The program number and the program name are saved when you exit the input box. You can save the traversing blocks with the “Save Block” button.

5.4 Parameterization with "Parameterize FM 354"

Entering the values

You have a variety of options for entering your parameterization data.

1. User data

   You can input values or select texts in a table. Select input fields with the cursor and enter the values. You can select the associated texts for the values with the space key.

2. Machine data

   The values are entered in dialog boxes and windows selected by option tabs.

   To display the machine data in a table, select the menu **View ▸ Table form**. Here you can enter the values as described in the user data section.

3. Tool compensation data and increment sizes

   You can input the values in a table. Select input fields with the cursor and enter the values.

4. Traversing programs

   Traversing programs are input in text format.

A comment column is included in the tables for MD, SM, and TO values. This comment is not stored in the data block. It can be printed out or stored with the data in the file on export.

Menus of "Parameterize FM 354"

The following table shows you an overview of the menus of "Parameterize FM 354".

Table 5-9 Menus of "Parameterize FM 354"

Menu title or entry (with single command)	Shortcut	Significance
File	–	Create, open, save, print and generate data blocks
New >	–	Creates a new data block
Machine data	–	Creates a new DB-MD
Increment	–	Creates a new DB-SM
Tool offset data	–	Creates a new DB-TO
Traversing program	–	Creates a new DB-NC
Open >	Ctrl + O	Opens the data block stored on the programming device/PC
Machine data	–	Opens the DB–MD stored on the programming device/PC
Increment	–	Opens the DB-SM stored on the programming device/PC
Tool offset data	–	Opens the DB-WK stored on the programming device/PC
Traversing program	–	Opens the DB-NC stored on the programming device/PC
Import...	Ctrl + O	Opens a data block which has been saved as a file

Table 5-9 Menus of “Parameterize FM 354”, continued

Menu title or entry (with single command)	Shortcut	Significance
Close	Ctrl + F4	Closes the window of the current DB
Save	Ctrl + S	Saves the current data block on the programming device/PC
Export...	–	Saves the current data block in a file
Check consistency	–	Checks the data in the current window for errors
Create SDB	–	Reads the FM data blocks which have been generated from this SDB (system data block), and stores them on the programming device/PC.
Display SDB...	–	Displays the SDBs for the FM 354 which exist on the programming device/PC; they can then be deleted.
Print...	Ctrl + P	Prints all or part of the current data block
Print preview	–	Displays the document in the print preview - no editing possible
Set page...	–	Specifies page layout for printing
Set printer...	–	Sets up the printer and sets print options
1 <Name of DB last opened>	–	Opens the DB which was last opened
2 <Name of penultimate DB opened>	–	Opens the DB which was open before the last one
3 <Name of third–last DB>	–	Opens the third–last DB
4 <Name of fourth–last DB>	–	Opens the fourth–last DB
Close	Alt + F4	Closes all parameterization windows and ends parameterization
Edit	–	Undo the last action, cut, copy, paste and delete selected objects, search and default value
Undo	Ctrl + Z	Undoes the last action
Cut	Ctrl + X	Deletes the selected data and saves it in a buffer (clipboard)
Copy	Ctrl + C	Copies the selected data to a buffer (clipboard)
Paste	Ctrl + V	Inserts the clipboard contents at the cursor position
Replace cells	–	Overwrites the field in a table with the clipboard contents
Find	Ctrl + F	Searches for text; the text may also be a number (e.g., MD No.)
Default values	–	Fills the current data block with default values

Table 5-9 Menus of “Parameterize FM 354”, continued

Menu title or entry (with single command)	Shortcut	Significance
Destination system	–	Transfers data and data blocks
√ Communications	–	Establishes or disconnects online connection with destination system
Load >	–	Loads data blocks or user data
in FM	–	Loads the current data block on the FM 354
in PG or FM...	–	Opens a transfer dialog
Online editing >	–	Edits the data blocks on the FM 354
Machine data	–	Edits the machine data on the FM 354
Increment	–	Edits the increments on the FM 354
Tool offset data	–	Edits the tool offset data on the FM 354
Traversing program	–	Edits the traversing programs on the FM 354
User data	–	Edits the user data on the FM 354
Compress FM-RAM	–	Compresses the FM 354 RAM. This is possible only if the CPU is in STOP mode.
Clear flash memory	–	Clears the FLASH memory on the FM 354
Test	–	Startup and troubleshooting
√Startup	–	Opens the startup window. Module control and observation
√Troubleshooting	–	Opens the troubleshooting window. Displays faults in the module
√Service data	–	Opens the window to look at servicing data
View	–	Select different views and presentations
Table form	–	Switches between dialog and table format (only with MD)
Contents of column 5	–	Defines what appears in the last column (MD only)
Default value	–	Displays default values (recommendations)
Limits	–	Displays upper and lower limits
√ Function bar	–	Displays the function bar (on/off)
√ Status line	–	Displays the status line (on/off)
Overview	–	The overview display for parameterization appears

Table 5-9 Menus of “Parameterize FM 354”, continued

Menu title or entry (with single command)	Shortcut	Significance
Extras	–	Settings in the data blocks
Set system of measurement	–	Change the system of measurement in the current window
√ 10^{-3} mm	–	Input in mm
10^{-4} inch	–	Input in inches
10^{-4} grd	–	Input in degrees
10^{-2} degrees	–	Input in degrees
Window	–	Arranges all parameterization windows. Changes to a specified window.
Arrange	–	Arranges all windows
Overlapping	Shift + F5	Stacks all windows, one behind the other
Horizontal	–	Spaces all windows uniformly, top to bottom
Vertical	–	Spaces all windows uniformly, left to right
Arrange icons	–	Arranges parameterization window icons
Close all	–	Closes all open windows
√1 <opened window 1>	–	Changes to window <window name>
n <currently open window n>	–	Changes to window <window name>
Help	–	Search and display help functions
Help topics...	F1	Offers a variety of ways to access help information
Using help	–	Displays information on how to use help
Info...	–	Displays information about the current version of the parameterization tool

5.5 Storing the parameter data in SDB $\geq$ 1 000

Overview

The FM 354 stores its parameter data internally.

In order to ensure that the parameter data are available if a fault develops on the FM 354 and no programming device/PC is at hand, the data can be stored in a system data block in the CPU (SDB $\geq$ 1 000). The CPU transfers the data stored in SDB $\geq$ 1 000 to the FM 354 on each new start. If the FM 354 has no machine data or the internal time stamp (time of creation) is invalid, the data are transferred from SDB $\geq$ 1 000 to the FM 354 and saved there.

You must ensure that the parameter data in SDB $\geq$ 1 000 always match the parameter data on the FM 354 when start–up is complete.

Note

SDB $\geq$ 1 000 should not be created until start–up is finished.

If you need to modify the data subsequently, you should generate SDB $\geq$ 1 000 again and load it into the CPU. You can delete the previous SDB before you load the new one, however the new SDB automatically overwrites the old one when it is generated. The old SDB and the new SDB do not have to be allocated the same number.

Creating the SDB

Prerequisite: Online connection with the FM 354

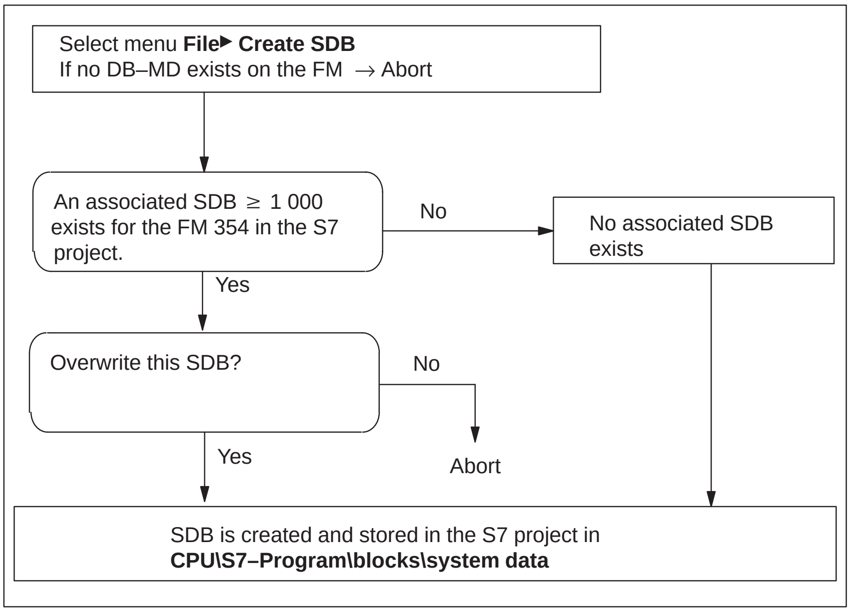


Fig. 5-8 Creating SDB $\geq$ 1 000

Display/delete SDB in the S7 project

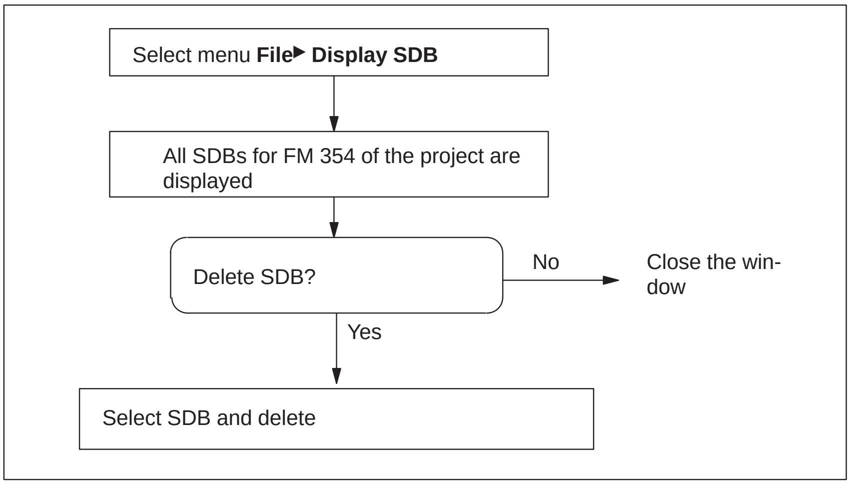


Fig. 5-9 Displaying/deleting SDB $\geq$ 1 000

Loading the SDB in the CPU

When you have created the SDB, you must load the “system data” of the project into the CPU.

There are two ways of proceeding:

1. Method

   Select the online window in the *SIMATIC Manager* (the online and offline windows must be open)

   Copy the system data from the offline project in **CPU\S7-Program\blocks\system data** into the online project (drag with the mouse or select Copy/Paste).

2. method

   Select the system data in the *SIMATIC Manager* in **CPU\S7-Program\blocks\system data**.

   Activate the menu **Target system ▸ Load** (or the right mouse button) to load the system data into the CPU

   or

   Use the menu **Target system ▸ Load in EPROM memory card on CPU**

You can also program the memory card for the CPU on a programming device/PC.

If the configuration is loaded from HW-CONFIG, this SDB is **not** loaded into the CPU.

Deleting SDBs in the CPU

To delete the SDBs in the CPU:

1. Select “Parameterize FM 354”.
2. Select menu **File▸ Display SDB**. Delete the SDB(s).
3. Close “Parameterize FM 354” and in the *SIMATIC Manager* in Online Project **select CPU\S7-Program\blocks\system data**. Delete the system data.
4. Transfer the system data to the CPU again (see above)

■

Programming the FM 354

6

Summary

The present programming instructions describe the functions (FCs) that allow you to establish communications between the CPU and the FM 354 function module in the SIMATIC S7-300.

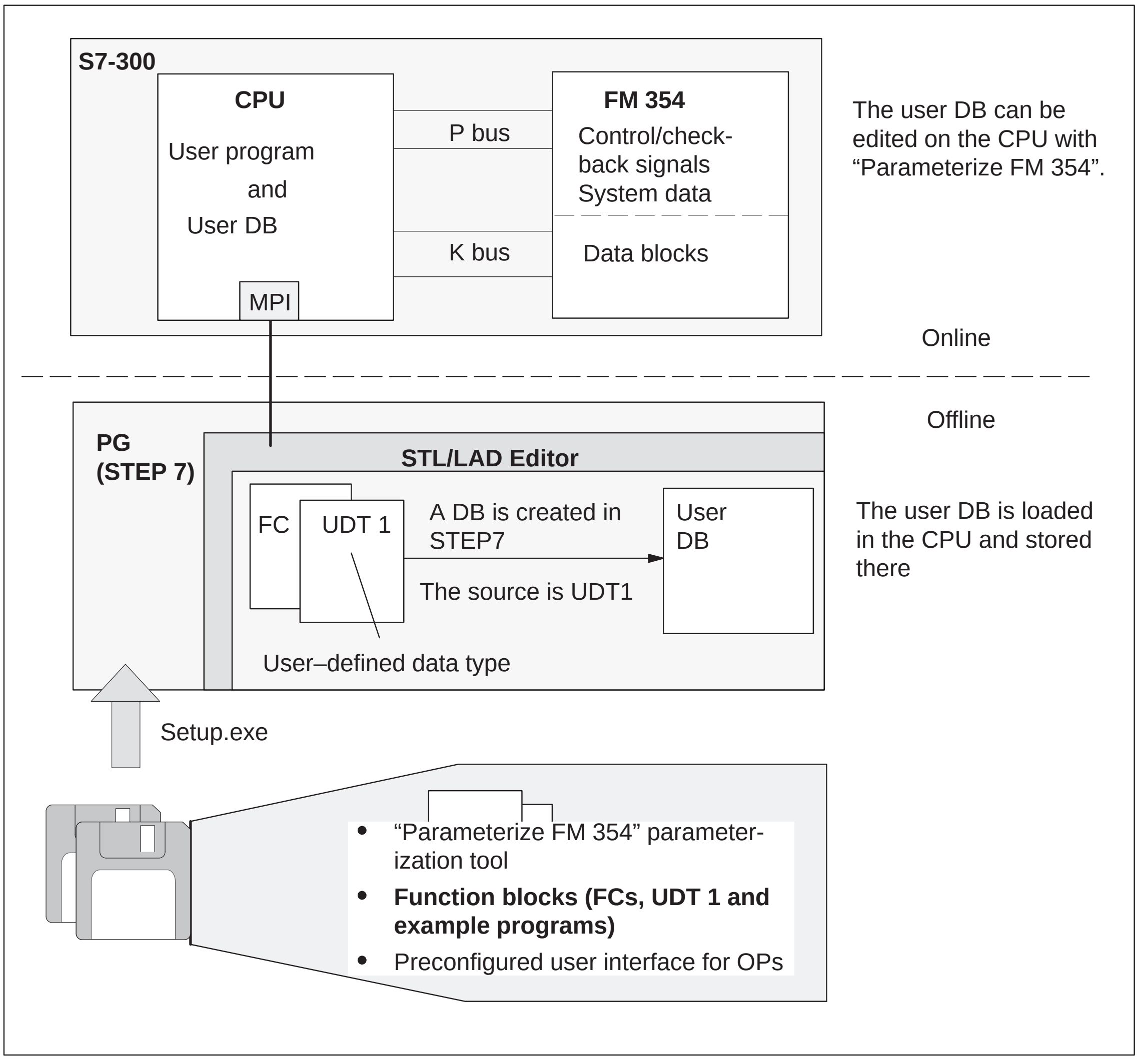


Fig. 6-1 Overview of programming

Prerequisites

The following prerequisites must be fulfilled in order to control the FM 354 from your user program:

- You have installed the software on your programming device/PC, as described in Section 5.1.
- The link between the programming device/PC and the S7-300 CPU must already be set up (see Figure 4-1).

Creating the user DB

Proceed as follows:

1. Generate a data block in STEP 7 (DB 1).
2. Open DB 1 and select the property “with assigned user-specific data type”.

   Result: UDT 1 (user–defined data type) is offered
3. Click UDT 1

   Result: You have created the user DB (DB 1).
4. Load this user DB and save it in the CPU.
5. You can use “Parameterize FM 354” to fill the user DB in the CPU with data.

You must create a user DB for each channel.

Function blocks

The following table gives you a general view of the function-block package (FCs) for the FM 354.

Table 6-1 Technology functions for the FM 354

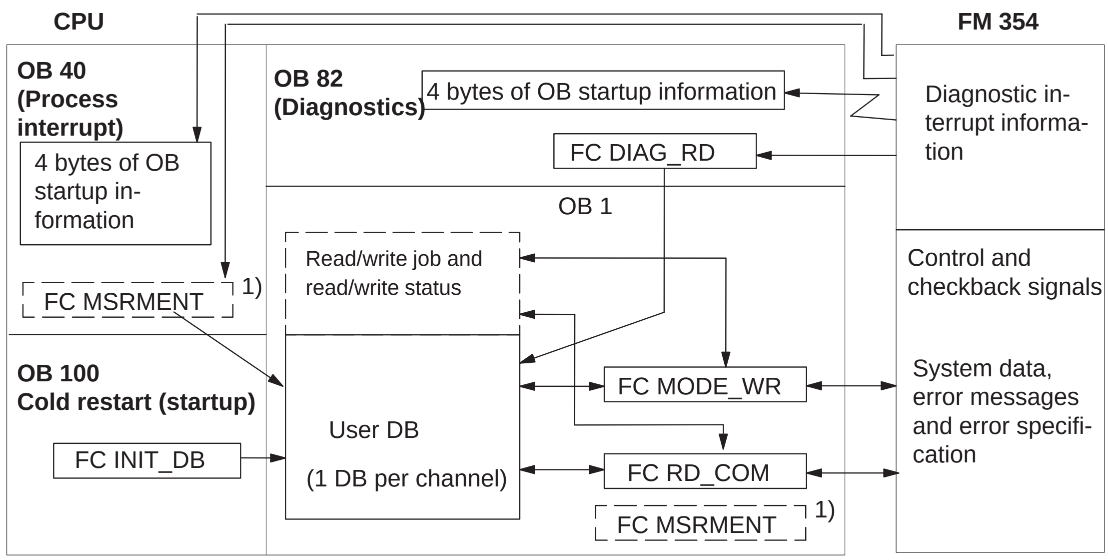
Function Block No.	Function Block Name	Significance
FC 1	INIT_DB	Initialize user DB
FC 2	MODE_WR	Control operating modes and process write jobs
FC 3	RD_COM	Process read jobs cyclically
FC 4	DIAG_RD	Read diagnostic interrupt data in OB 82
FC 5	MSRMENT	Read measured values
FC 6	DIAG_INF	Read diagnostic interrupt data in OB 1

Note

You can change the FC number for your project. The number is changed in the SIMATIC Manager by renaming the FC in your project with a free number. These changes should be entered in the symbol table at the same time.

Linking the FM 354 into the user program

The following figure shows you how the FM 354, the user data block (user DB) and technology functions communicate.



1) This FC can only be called in OB 40 **or** in OB 1, but not in both simultaneously.

Fig. 6-2 Overview of linking the FM 354 into the user program

Tips for the user

The user requires at least FC INIT_DB in order to initialize the user DB and FC MODE_WR for mode and write job processing.

FC RD_COM for reading data is only required if the FM data are to be processed in the user program (e.g. for display purposes).

Regardless of which or how many technology functions you use, you need a data block with a predefined structure (UDT 1) to contain all the necessary data or data storage areas. This data block is implemented as a user data block, and one way to set its default values is with the “Parameterize FM 354” parameterization tool.

Chapter overview

6.1 FC INIT_DB (FC 1) – Initialize user DB

Task

You can use FC INIT_DB to initialize specific areas of your user DB. To do this, call FC INIT_DB once in the start–up organization block OB 100.

The FC performs the following actions:

1. Enters addressing values in the user DB
   - FM address
   - Offset address
2. Deletes the following structures in the user DB
   - CONTROL_SIGNALS
   - CHECKBACK_SIGNALS
   - JOB_WR (write job)
   - JOB_RD (read job)

Call options

Call in LAD notation (ladder diagram)	Call in STL notation (statement list)

FC INIT_DB
EN ENO
DB_NO
CH_NO
LADDR
	`CALL INIT_DB(`
	`DB_NO := ,`
	`CH_NO := ,`
	`LADDR := );`

Description of parameters

The following table describes the parameters of this FC.

Name	Data type	P type	Meaning
DB_NO	WORD	I	Data block number
CH_NO	BYTE	I	Number of axis: 0 or 1 permitted, because single–channel module 4...255 – invalid BIE = 0
LADDR	INT	I	Logical base address of module, transfer entry from HW-CONFIG

Parameter types: I = input parameter

Principle of operation

This function works together with a user DB. The structure of the user DB can be found in the library FMSTSVLI in data type UDT 1. You need a user DB which contains entries for addressing the FM 354 and the data for the individual functions of the FM 354. The DB number is passed when you call the FC with the DB_NO parameter.

Error evaluation

Errors which occur are indicated in the binary result (BIE = 0).

Possible errors are:

Unknown channel number CH_NO; the user DB is not initialized.

Example call

An example call is shown below for FC INIT_DB.

```
STL                                                Explanation
VAR_TEMP
      MODUL_ADR         : INT;                     // Module address
END_VAR

...

      L    512;                                    // Enter module address
      T    MODUL_ADR;
                                                   // Module address
      CALL INIT_DB(                                // DB number
           DB_NO        := W#16#1,                 // Only one channel on the module
           CH_NO        := B#16#1,                 // Module address
           LADDR        := MODUL_ADR);
                                                   // Binary result
      UN   BIE;                                    // Error on initialization
      S    FEHLER_INITIALISIERUNG;
...
```

6.2 FC MODE_WR (FC 2) – Control operating modes and process write jobs

Task

You can use FC MODE_WR to:

- Control modes
- Process write jobs

To do this, you must call FC MODE_WR once in the OB 1 cycle.

The FC performs the following actions:

1. Reads the checkback signals. The values read by the FC are stored in the user DB in the structure CHECKBACK_SIGNALS.
2. Transfers the control signals from the user DB (structure CONTROL_SIGNALS). Depending on the mode selected (CHECKBACK_SIGNALS.MODE) control signals CONTROL_SIGNALS.START, CONTROL_SIGNALS.DIR_P and CONTROL_SIGNALS.DIR_M are cleared when a start is detected (edge generation of signals for FM).
3. Executes the write job from the user DB (JOB_WR), transfers the associated data from the user DB, and displays the write job status.
4. Generates the status bits JOB_WR.MODE_BUSY (operating mode busy, i.e. started) and JOB_WR.POS_REACHED (position reached).

Call options

Call in LAD notation (ladder diagram)	Call in STL notation (statement list)

FC MODE_WR
EN — ENO
DB_NO — RET_VAL
	`CALL MODE_WR(`
	`DB_NO := ,`
	`RET_VAL := );`

Description of parameters

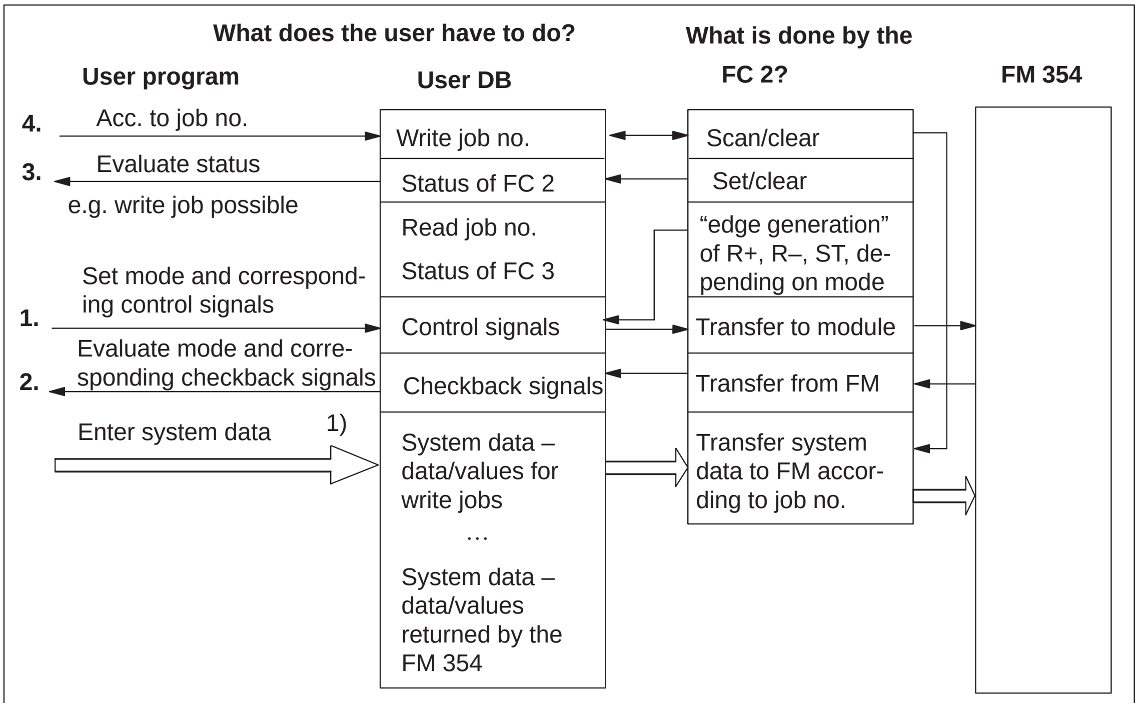
The following table describes the parameters of this FC.

Name	Data type	P type	Meaning
DB_NO	WORD	Q	Data block number
RET_VAL	INT	A	Return code of SFC 58 “WR_REC”

Parameter types: I = input parameter, Q = output parameter

Principle of operation

This function works together with a user DB. The structure of the user DB can be found in the library FMSTSVLI in data type UDT 1. You need a user DB which contains entries for addressing the FM 354 and the data for the individual functions of the FM 354. The DB number is passed when you call the FC with the DB_NO parameter.



1) Before the appropriate write job no. is set in the user DB, the system data to be written must be entered.

Error evaluation

Errors which occur are indicated in the binary result (BIE = 0).

Possible errors are:

- Unknown write job (see JOB_WR.UNKNOWN)
- Data transfer error during communication with SFC 58 “WR_REC”. The error is returned in the output parameter RET_VAL (see reference manual *System Software for S7-300/400; System and Standard Functions*).
- The transferred data are verified and interpreted by the module. If a data error occurs, CHECKBACK_SIGNALS.DATA_ERR in the user DB structure is set to “1”. Further information on data errors can be found in the parameterization tool in the menu **Test** ▸ **Alarms** and in Section 11.

Example call

An example call is shown below for FC MODE_WR.

```
STL                                                  Explanation

...

      O    DB_FM.JOB_WR.BUSY;                        // Write job busy
      O    DB_FM.JOB_WR.IMPOSS;                      // Write job processing impossible
      SPB  DAWR;                                     // Jump to call

AT02: U    G_STUFE_SETZEN;
      SPEN STRS;
      L    B#16#1;                                   // Write job no. 1 for velocity level
      SPA  EINT;

STRS: L    B#16#0;                                   // Only transfer control signals

EINT: T    DB_FM.JOB_WR.NO;                          // Write job no. in user DB

DAWR: CALL MODE_WR(                                  // FC Write data
           DB_NO     := W#16#1,
           RET_VAL   := FEHLERCODE_SCHREIBEN)

      UN   BIE;                                      // Binary result
      S    FEHLER_SCHREIBFKT;                        // Error on initialization
...
```

6.2.1 Process write jobs

Overview

Before a write job is processed, the data area belonging to the write job must be filled with the appropriate values. The last write job must have finished processing, i.e. JOB_WR.NO in the user DB (data byte DBB0) must have been cleared and status bit JOB_WR.DONE enabled.

You initiate a write job by entering the write job no. in JOB_WR.NO.

The following write jobs (JOB_WR.NO) are known:

Legend for the table below:

Operating mode:

T – Jogging
STE – Open–loop control
REF – Reference point approach
SM – Incremental relative
MDI – MDI (Manual Data Input)
A/AE – Automatic/Automatic single block

Operating modes / System data	Job no.	Addr. in user DB	T	STE	REF	SM	MDI	A/AE	see Section
Reference data is data/parameters for the corresponding mode.									
VLEVEL_1_2 – Velocity levels 1, 2	1	90.0	□	○	○	□	○	○	9.2.1
CLEVEL_1_2 – Voltage levels 1, 2	2	98.0	○	□	○	○	○	○	9.2.2
TARGET_254 – Setpoint for increment	3	86.0	○	○	○	□	○	○	9.2.4
MDI_BLOCK	6	106.0	○	○	○	○	□	○	9.2.5
Reference data with execution activates settings/functions that apply in multiple modes.									
PAR_CHAN – change parameter/data	8	126.0	x	x	x	x	x	x	9.3.1
SINGLE_FUNCTIONS	10	40.0	□	x	□	□	□	□	9.3.2
SINGLE_COMMANDS	11	42.0	x	x	x	x	x	x	9.3.3
ZERO_OFFSET	12	44.0	x	x	–	x	x	x	9.3.4
SETTING_ACT_VALUE	13	48.0	x	x	–	x	x	x	9.3.5
FLYING_SETTING_ACT_VALUE	14	52.0	x	x	–	x	x	–	9.3.6
DIG_IO – digital outputs	15	150.0	x	x	x	x	x	x	9.8.2
MDI_FLY	16	152.0	–	–	–	–	x	–	9.2.5
PROG_SEL – program selection	17	172.0	–	–	–	–	–	□	9.2.6
REQ_APP – request application data	18	176.0	x	x	x	x	x	x	9.3.7
TEACH_IN	19	180.0	x	–	–	x	x	–	9.3.8
SETTING_REFERNCE_POINT	21	56.0	x	x	x	x	x	–	9.3.9
SRV_IN – reserved	22	186.0							

○ Data is accepted and only processed in the corresponding mode.

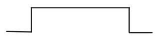
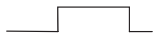
x Data is accepted or processed, as applicable.

– Data are rejected with error message (see Troubleshooting, Table 11-5 Class 4 No. 1).

□ Data required for movement of the axis; the servo enable is required for single settings. Data/single settings are transferred at least once to the FM 354.

Write job status The status of a write job is indicated in the user DB (in data byte DBB1).

Table 6-2 Write job status

Bit in JOB_WR (DBX1.)	Significance
.BUSY, 0	= 1, write job busy This bit is set by FC MODE_WR as soon as it starts processing a write job (JOB_WR.NO > 0 and bit 2 in JOB_WR.IMPOSS = 0). This bit is cleared by FC MODE_WR as soon as the write job has finished running (JOB_WR.NO = 0).
.DONE, 1	= 1, write job finished This bit is set by FC MODE_WR as soon as it has finished a write job (also with error and unknown job). This bit is cleared by FC MODE_WR when a new write job begins. You can clear this bit manually.
.IMPOSS, 2	= 1, write job processing is not possible in this cycle: – because the axis is not parameterized – because test mode is active – because no mode is active – because the selected mode is not yet active In this case, you can leave the write job (JOB_WR) or delete it. FC MODE_WR clears the bit when all the above conditions have been met.
.UNKNOWN, 3	= 1, write job unknown The write job (JOB_WR) which you specified is not within the known range (see error evaluation). FC MODE_WR clears this bit as soon as JOB_WR contains a valid number. The unknown number is retained until then.
.MODE_BUSY, 6	= 1, when a mode/movement is started with the appropriate control signals or on a BL = 1 checkback (busy). MODE_BUSY when FC is called/started  WORKING [BL] when movement is started by FM 
.POS_REACHED, 7	0 = on the checkback POS_ROD = 0 (position reached, stop) or when a mode is started with the appropriate control signals. POS_REACHED  when FC is called/started POS_POD [PEH]  when movement is started by FM

6.2.2 Controlling operating

Overview

Control/checkback signals are required in order to control the axis in the individual operating modes.

The operating modes are described in Section 9.2. The control/checkback signals and their handling are described in Section 9.1.

The user must enter the control signals in the user DB. FC MODE_WR transfers the control signals from the user DB to the FM 354 and transfers the checkback signals from the FM 354 to the user DB.

Byte \ Bit	7	6	5	4	3	2	1	0
Control signals:								
20					BFQ/FSQ		TFB	
21	AF	SA	EFG	QMF	R+	R–	STP	ST
22	operating mode							
23	BP							
24	OVERR							
25								
Checkback signals:								
28	PARA			DF	BF/FS		TFGS	
29		PBR	T-L			WFG	BL	SFG
30	BAR							
31	PEH		FIWS		FR+	FR–	ME	SYN
32	MNR							
33				AMF				

Single settings (in the user DB starting at address 40) and single commands (in user DB starting at address 42) are also required in order to control the FM 354. These are transferred by means of write jobs (system data).

Individual settings	Individual commands
Length measurement Inprocess measurement Retrigger reference point Deactivate enable input Deactivate software end position monitoring Follow–up mode Software end position monitoring Automatic drift compensation **Servo enable** Parking axis Simulation	Activate machine data Delete distance to go Automatic block search backward Automatic block search forward Restart Undo set actual value

Troubleshooting

Checkback signals [BF/FS] and [DF] (group error messages)
Error specification in user program (if necessary) Read out DS 162 (on BF/FS) or read out DS 163 (on DF) See example application 2

Error acknowledgment

Set/clear control signal [BFQ/FSQ]
or
on message [DF] → write a new write job

In the following table, the control and checkback signals are explained in German and English.

Table 6-3 Control/checkback signals

German	English	Significance
Control signals		
BP	MODE PARAMETER	Operating mode parameters Velocity levels 1, 2 Voltage levels 1, 2 Increment selection 1...100, 254
BA	MODE	Operating mode Jogging 01 Open–loop control 02 Reference point approach 03 Incremental relative 04 MDI 06 Automatic 08 Automatic single block 09
R+	DIR_P	Direction plus
R–	DIR_M	Direction minus
STP	STOP	Stop
ST	START	Start
OVERR	OVERRIDE	Override
AF	DRV_EN	Drive enable
SA	SKIP_BLK	Enable bit for block skip
EFG	READ_EN	Read enable
QMF	ACK_MF	Acknowledgment M function
BFQ/FSQ	OT_ERR_A	Acknowledgment operator/travel error
TFB	TEST_EN	Switch P BUS interface to “start–up”
Checkback signals		
MNR	NUM_MF	M function number
BL	WORKING	Program running
SFG	START_EN	Start enable
BF/FS	OT_ERR	Operator/travel error

Table 6-3 Control/checkback signals, continued

German	English	Significance
BAR	MODE	Active operating mode
AMF	STR_MF	Modify M function
PBR	PR_BACK	Program scanning backward
T-L	DT_RUN	Dwell time running
PEH	POS_ROD	Position reached and stopped
FR+	GO_P	GO_plus
FR–	GO_M	GO_minus
ME	MSR_DONE	Measurement done
SYN	SYNC	Channel synchronized
DF	DATA_ERR	Data error
FIWS	FAVEL	Flying actual value done
TFGS	TST_STAT	Switchover, P-BUS port done
WFG	WAIT_EN	Wait for external enable
PARA	PARA	Channel parameterized

6.3 FC RD_COM (FC 3) – Process read jobs cyclically

Task

You can use FC RD_COM to execute read jobs. To do this, call FC RD_COM once in the OB1 cycle.

The last read job must have finished processing, i.e. JOB_RD.NO in the user DB (data byte DBB2) must have been cleared and status bit JOB_RD.DONE enabled.

You initiate a new read job by entering the read job no. in JOB_RD.NO.

Do not include FC RD_COM in your user program if you do not process any read jobs.

The FC performs the following action:

Executes the read job (JOB_RD) from the user DB, transfers the associated data into the user DB, and displays the read job status.

Call options

Call in LAD notation (ladder diagram)	Call in STL notation (statement list)

FC RD_COM
EN ENO
DB_NO RET_VAL
	`CALL RD_COM(`
	`DB_NO := ,`
	`RET_VAL := );`

Description of parameters

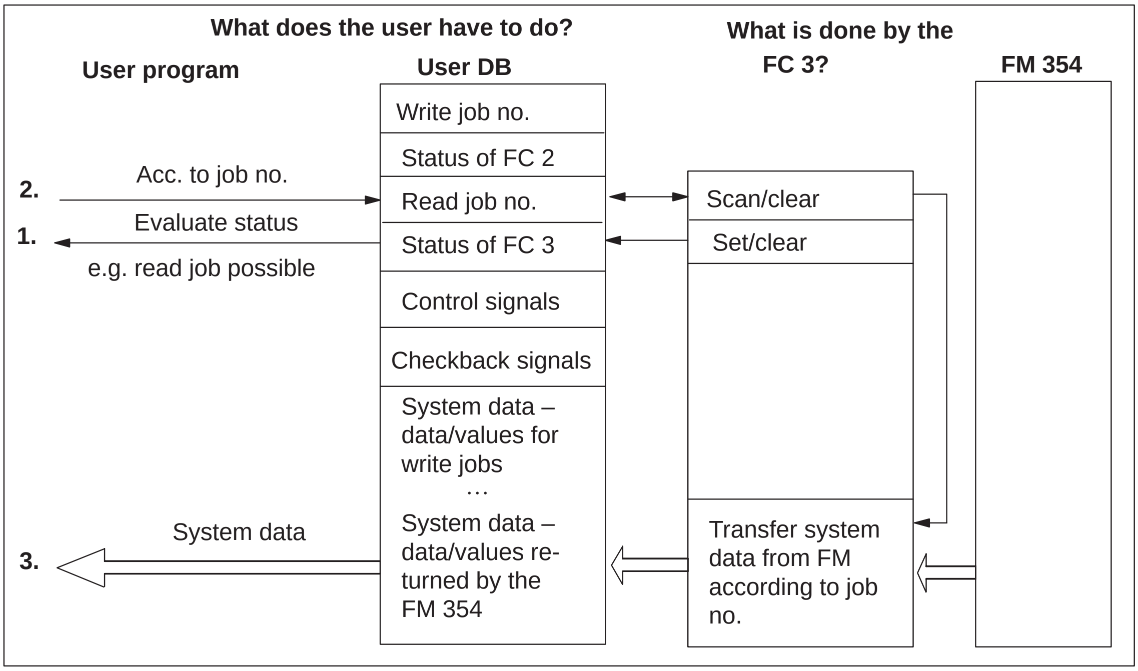
The following table describes the parameters of this FC.

Name	Data tpe	P type	Meaning
DB_NO	WORD	I	Data block number
RET_VAL	INT	Q	Return code of SFC 59 “RD_REC”

Parameter types: I = input parameter, Q = output parameter

Principle of operation

This function works together with a user DB. The structure of the user DB can be found in the library FMSTSVLI in data type UDT 1. You need a user DB which contains entries for addressing the FM 354 and the data for the individual functions of the FM 354. The DB number is passed when you call the FC with the DB_NO parameter.



The following read jobs (JOB_RD.NO) are known:

Legend for the table below:

Operating mode:

T	–	Jogging
STE	–	Open–loop control
REF	–	Reference point approach
SM	–	Incremental relative
MDI	–	MDI (Manual Data Input)
A/AE	–	Automatic/Automatic single block

Operating modes / System data	Job no.	Addr. in user DB	T	STE	REF	SM	MDI	A/AE	see Section
Display data is data/parameters returned by the FM.									
DIG_IO – dig. inputs/outputs	101	150.0	x	x	x	x	x	x	9.8
OP_DAT – basic operating data	102	198.0	x	x	x	x	x	x	9.3.11
ACT_BLCK – active NC block	103	230.0						x	9.3.12
NXT_BLCK – next NC block	104	250.0						x	
APP_DAT – application data	105	270.0	x	x	x	x	x	x	9.3.13
BLCK_EXT – actual value block change	107	286.0						x	9.3.14
SERV_DAT – service data	108	290.0	x	x	x	x	x	x	9.3.15
SRV_OUT – reserved	109	322.0							
OP_DAT1 – additional operating data	110	354.0	x	x	x	x	x	x	9.3.16
PAR_READ – parameters/data	114	366.0	x	x	x	x	x	x	9.3.17

○ Data is accepted and only processed in the corresponding mode.
x Data is accepted or processed, as applicable.
– Data are rejected with error message (see Troubleshooting, Table 11-5 Class 4 No. 1).

Read job status

The status of a read job is indicated in the user DB (in data byte DBB3).

Bit in JOB_RD (DBX3.)	Significance
.BUSY, 0	= 1, read job busy This bit is set by FC RD_COM as soon as it starts processing a read job (JOB_RD.NO > 0 and STATUS_RD.IMBOSS = 0) This bit is cleared by FC RD_COM as soon as the read job has finished running (JOB_RD.NO = 0).
.DONE, 1	= 1, read job finished This bit is set by FC RD_COM as soon as it has finished a read job (also with error and unknown job) This bit is cleared by FC RD_COM when a new read job begins. You can clear this bit manually.
.IMPOSS, 2	= 1, Read job not possible at the present time Read job processing is not possible: – because the axis is not parameterized – because no mode is preselected – because test mode is active In this case, you can leave the read job (JOB_RD.NO) or delete it. FC RD_COM clears the bit when all the above conditions have been met.
.UNKNOWN, 3	= 1, read job unknown The read job (JOB_RD.NO) which you specified is not within the known range (see error evaluation). FC RD_COM clears this bit as soon as JOB_RD.NO contains a valid number. The unknown number is retained until then.

Error evaluation

Errors which occur are indicated in the binary result (BIE = 0).

Possible errors are:

- Unknown read job (see JOB_RD.UNKNOWN)
- Data transfer error during communication with SFC 59 “RD_REC”. The error is returned in the output parameter RET_VAL (see reference manual *System Software for S7-300/400; System and Standard Functions*).

Example call An example call is shown below for FC RD_COM.

```
STL                                                 Explanation

...

      O     DB_FM.JOB_RD.BUSY;                      // Read job busy
      O     DB_FM.JOB_RD.IMPOSS;                    // Read job processing impossible
      SPB   DARD;                                   // Jump to call

      L     B#16#66;                                // Read job 102 for basic operating data
      T     DB_FM.JOB_RD.NO;                        // Store in job box

DARD: CALL  RD_COM(                                 // Call read data FC
            DB_NO     := W#16#1,                    // DB number
            RET_VAL   := FEHLERCODE_LESEN)          // Return value

      UN    BIE;                                    // Binary result
      S     FEHLER_LESEFKT;                         // Error on initialization
...
```

6.4 Reading diagnostic information

Overview FC DIAG_RD (FC 4) and FC DIAG_INF (FC 6) are used to read the diagnostic interrupt information in the user DB.

FC 4 and FC 6 are intended as alternatives which you can use according to your needs.

6.4.1 FC DIAG_RD (FC 4) – Read diagnostic interrupt data in OB 82

Task You are only allowed to call FC DIAG_RD in interrupt OB 82.

Can only be used with appropriate S7-300 CPU versions; see compatibility list in Table 1-1.

Call options

Call in LAD notation (ladder diagram)	Call in STL notation (statement list)

FC DIAG_RD
EN ENO
DB_NO RET_VAL
IN_DIAG
	`CALL DIAG_RD(`
	`DB_NO := ,`
	`RET_VAL := ,`
	`IN_DIAG := );`

Description of parameters

The following table describes the parameters of the FC DIAG_RD.

Name	Data tpe	P type	Meaning
DB_NO	WORD	I	Data block number
RET_VAL	INT	Q	Return code of SFC 59 RD_REC
IN_DIAG	BOOL	I/Q	Initiate reading of diagnostic data; is cleared after execution of FC 4.

Parameter types: I = input parameter, Q = output parameter,
I/Q= in/out parameter (initiation parameter)

Principle of operation

This function works together with a user DB. The DB number is passed when you call the FC with the DB_NO parameter.

Reading of the diagnostic interrupt data DIAGNOSTIC_INT_INFO (in user DB starting at address 72) is started when you set the in/out parameter IN_DIAG to one. The parameter is reset by the FC after the job is executed.

The in/out parameter remains set while the job is running. Data transfer is complete when the in/out parameter is reset (IN_DIAG = FALSE).

Error evaluation

Errors which occur are indicated in the binary result (BIE = 0).

Possible errors are:

Data transfer error during communication with SFC 59 “RD_REC”. The error is returned in the output parameter RET_VAL (see reference manual *System Software for S7-300/400; System and Standard Functions*).

<table>
<tr><th colspan="4">Diagnostic interrupt
Message to the CPU (precondition: interrupt message activated (see Section 5.2)</th></tr>
<tr><td rowspan="3">No OB 82 exists → CPU switches to STOP</td><td colspan="2">OB 82</td><td>OB 1</td></tr>
<tr><td>Enters the diagnostic information in the diagnostic buffer of the CPU (4 bytes) and calls SFC 52</td><td>Enters the diagnostic information in the user DB starting at address 72 and calls FC 4</td><td>Calls FC 6</td></tr>
<tr><td colspan="3">On operating error:
(addr. in user DB 80.7)
For further error specification by reading out DS 164 in OB 1 see example application 2</td></tr>
</table>

Diagnostic data

The following table contains the diagnostic information DIAGNOSTIC_INT_INFO in the user DB starting at address 72.

Table 6-4 Diagnostic information

Data format	Byte.Bit No.	Significance
4-byte	0.0	Module/group disturbances (incoming and outgoing)
	0.1	Internal error/HW error (group error bytes 2, 3)
	0.2	External error
	0.3	External channel error (group error byte 8)
	0.6	Module not parameterized
	1.0...3	Type class of module; for FM 354 = 08H
	1.4	Channel information present
	2.1	Communication disturbance (K bus)
	2.3	Time monitoring actuated/Watchdog
	2.4	Module internal power supply failed (NMI)
	3.2	FEPROM error
	3.3	RAM error
	3.6	Process interrupt lost
10-byte	4	FM Pos identifier (74H)
	5	Length of diagnostic information (16)
	6	No. of channels (1)
	7	Channel error vector (1)
	8.0	Cable break (incremental encoder)
	8.1	Error in absolute encoder
	8.2	Error pulse incr. or zero mark missing
	8.3	Voltage monitoring - encoder
	8.7	Operating error (see Chapter 11, Troubleshooting)
	9...13	Free

Hints to the user

In a diagnostic event, bytes 0 to 3 are automatically transmitted to the CPU, and the diagnostic organization block (OB82) is called up. The diagnostic OB should be included in the user program; otherwise the CPU will go to the Stop state. Byte 0 contains group error messages that are set simultaneously with the corresponding messages in bytes 2, 3 and 8.

The operating error (byte 8.7) is specified again. The error numbers are available for display purposes in the diagnostic buffer of the FM 354 and in the data block for status messages (DB-SS) (see Section 8.1). For special error evaluations in the user program, these error numbers are available in DS164.

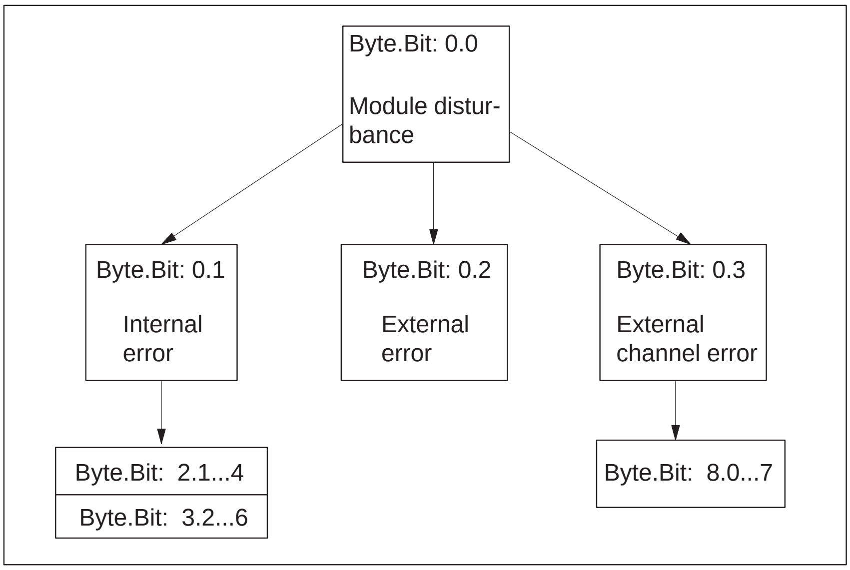


Fig. 6-3 Evaluation of diagnostic information

Example call in OB 82

An example call is shown below for FC DIAG_RD.

```
STL                                              Explanation

...

     S    DIAG_READ;                             // Initiate read function

     CALL DIAG_INF(                              // Call diagnostic information FC
          DB_NO     := W#16#1,                   // DB number
          RET_VAL   := FEHLERCODE_LESEN,         // Return value
          IN_DIAG   := DIAG_READ);               // Initiate reading

     UN   BIE;                                   // Binary result
     S    FEHLER_LESEFKT;                        // Error on initialization
...
```

6.4.2 FC DIAG_INF (FC 6) – Read diagnostic interrupt data in OB 1

Task

You can call FC DIAG_INF in OB 1 (or at another cyclical program level).

For call options, parameters and evaluation, see Section 6.4.1.

Principle of operation

This function works together with a user DB. The DB number is passed when you call the FC with the DB_NO parameter.

Reading of the diagnostic interrupt data DIAGNOSTIC_INT_INFO (in user DB starting at address 72) is started when you set the in/out parameter IN_DIAG to one. The parameter is reset by the FC after the job is executed. The FC must be called up until it has reset the in/out parameter. When the FM 354 is used centrally, the Read job is processed within a single function-block callup. When the FM 354 is used in a distributed configuration, it may take several function-block callups to process the Read job.

The in/out parameter remains set while the job is running. Data transfer is complete when the in/out parameter is reset (IN_DIAG = FALSE).

Example call in OB 1

An example call is shown below for FC DIAG_INF.

```
STL                                               Explanation
...

      U    DIAG_READ;                             // Call FC if initiation flag set
      SPB  DIRD;
      S    DIAG_READ;                             // Initiate read function

DIRD: CALL DIAG_INF(                              // Call diagnostic information FC
           DB_NO     := W#16#1,                   // DB number
           RET_VAL   := FEHLERCODE_LESEN,         // Return value
           IN_DIAG   := DIAG_READ);               // Initiate reading

      U    DIAG_READ;                             // Jump to end if read job not // yet fin-
      SPB  END;                                   ished

      UN   BIE;                                   // Binary result
      S    FEHLER_LESEFKT;                        // Error on read function

END:  NOP  0;

...
```

6.5 FC MSRMENT (FC 5) – Read measured values

Task

You use FC MSRMENT to read the measured values into the user DB (starting at address 60). You can call FC MSRMENT in OB 40, if the process interrupt was activated (see Section 5.2), or in OB 1. You are not allowed to call FC 5 simultaneously in both OBs.

FC 5 can only be used in OB 40 with the appropriate S7-300 CPU versions; see compatibility list in Table 1-1.

Centralized use	Distributed use (under development)
Called either in interrupt OB 40 (under development) or in the OB 1 context	Call in interrupt OB 40 not possible, since data transfer from the DP (distributed I/O) is not synchronous; hence call in the OB 1 context

Call options

Call in LAD notation (ladder diagram)	Call in STL notation (statement list)

FC MSRMENT
EN ENO
DB_NO RET_VAL
IN_MSR
	`CALL MSRMENT(`
	`DB_NO := ,`
	`RET_VAL := ,`
	`IN_MSR := );`

Description of parameters

The following table describes the parameters of FC MSRMENT.

Name	Data type	P type	Meaning
DB_NO	WORD	I	Data block number
RET_VAL	INT	Q	Return code of SFC 59 RD_REC
IN_MSR	BOOL	I/Q	Start Read process

Parameter types: I = input parameter, Q = output parameter,
I/Q = in/out parameter (initiation parameter)

Principle of operation

This function works together with a user DB. The DB number is passed when you call the FC with the DB_NO parameter.

Reading of the measured values MEASUREMENT_VALUES (in user DB starting at address 60) is started when you set the in/out parameter IN_MSR to one. The parameter is reset by the FC after the job is executed. The FC must be called up until it has reset the in/out parameter. When the FM 354 is used centrally, the Read job is processed within a single function-block callup. When the FM 354 is used in a distributed configuration, it may take several function block calls to process the read job (only applies when called in OB 1).

The in/out parameter remains set while the job is running. Data transfer is complete when the in/out parameter is reset (IN_MSR = FALSE).

Error evaluation

Errors which occur are indicated in the binary result (BIE = 0).

Possible errors are:

Data transfer error during communication with SFC 59 “RD_REC”. The error is returned in the output parameter RET_VAL (see reference manual *System Software for S7-300/400; System and Standard Functions*).

Example call in OB 1

An example call is shown below in OB 1.

```
STL                                                   Explanation
      U    DB_FM.CHECKBACK_SIGNALS.MSR_DONE;          // “Measurement completed” checkback
      FP   FLANKENMERKER_MESSUNG_BEENDET;             signal
      S    ”ANSTOSS_LESEFKT”;                         // Edge flag for “Measurement com-
                                                      pleted”
      CALL MSRMENT(                                   // Set initiation parameters
           DB_NO     := W#16#1,
           RET_VAL   := FEHLERCODE_LESEN,             // CALLUP OF FC MSRMENT
           IN_MSR    := ”ANSTOSS_LESEFKT”);

      U    ”ANSTOSS_LESEFKT”;
      SPB  NWE;
                                                      // Initiation bit is still set

      UN   BIE;
      S    FEHLER_LESEFKT;
                                                      // Communication error
NWE:  NOP 0;                                          // Indicate error in Read function
```

Example call in OB 40

An example call is shown below in OB 40.

```
STL                                                 Explanation
...
      S    MW_LESEN;                                // Set job

      CALL MSRMENT(                                 // Call FC for reading measured values
           DB_NO        := W#16#1,                  // DB number
           RET_VAL      := FEHLERCODE_LESEN,        // Return value
           IN_MSR       := MW_LESEN);               // Initiation parameter

      UN   BIE;                                     // Binary result
      S    FEHLER_LESEFKT;                          // Display error on read function
...
```

Note

The activation of “measurement” and the generation of the measured values is described in Section 9.3.10.

6.6 User data block

Overview

The following table provides you with a description of the user data block structure.

Table 6-5 User DB for the FM 354

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
Job box for FC MODE_WR						
	0.0		JOB_WR	STRUCT		Write jobs
0.0	+0.0		NO	BYTE	B#16#0	Write job number
1.0	+1.0		BUSY	BOOL	FALSE	Write job busy
1.1	+1.1		DONE	BOOL	FALSE	Write job finished
1.2	+1.2		IMPOSS	BOOL	FALSE	Write job impossible
1.3	+1.3		UNKNOWN	BOOL	FALSE	Write job unknown
1.4	+1.4		BIT1_4	BOOL	FALSE	Reserved
1.5	+1.5		BIT1_5	BOOL	FALSE	Reserved
1.6	+1.6		MODE_BUSY	BOOL	FALSE	Start an operating mode
1.1	+1.7		POS_REACHED	BOOL	FALSE	Position reached
	=2.0			END_STRUCT		
Job box for FC RD_COM						
	2.0		JOB_RD	STRUCT		Read jobs
2.0	+2.0		NO	BYTE	B#16#0	Read job number
3.0	+3.0		BUSY	BOOL	FALSE	Read job busy
3.1	+3.1		DONE	BOOL	FALSE	Read job finished
3.2	+3.2		IMPOSS	BOOL	FALSE	Read job impossible
3.3	+3.3		UNKNOWN	BOOL	FALSE	Read job unknown
	=2.0			END_STRUCT		
is entered by FC INIT_DB						
4.0	+4.0		WORD4	WORD	W#16#0	Reserved
6.0	+6.0		WORD6	WORD	W#16#0	Reserved
8.0	+8.0		WORD8	WORD	W#16#0	Reserved
10.0	+10.0		WORD10	WORD	W#16#0	Reserved
12.0	+12.0	stat	MOD_ADR	WORD	W#16#0	Module address
14.0	+14.0	stat	CH_ADR	DWORD	DW#16#0	Channel address
18.0	+18.0	stat	DS_OFFS	BYTE	B#16#0	Offset for channel–specific data set number
19.0	+19.0	stat	RESERV_2	BYTE	B#16#0	Reserved

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
Control signals:			**FC MODE_WR**			
	20.0	stat	CONTROL_ SIGNALS	STRUCT		Control signals
20.0	+0.0		BIT0_0	BOOL	FALSE	Reserved
20.1	+0.1		TEST_EN	BOOL	FALSE	Switchover, P-bus interface
20.2	+0.2		BIT0_2	BOOL	FALSE	Reserved
20.3	+0.3		OT_ERR_A	BOOL	FALSE	Acknowledge operator-control/ traversing error
20.4	+0.4		BIT0_4	BOOL	FALSE	Reserved
20.5	+0.5		BIT0_5	BOOL	FALSE	Reserved
20.6	+0.6		BIT0_6	BOOL	FALSE	Reserved
20.7	+0.7		BIT0_7	BOOL	FALSE	Reserved
21.0	+1.0		START	BOOL	FALSE	Start
21.1	+1.1		STOP	BOOL	FALSE	Stop
21.2	+1.2		DIR_M	BOOL	FALSE	Direction minus
21.3	+1.3		DIR_P	BOOL	FALSE	Direction plus
21.4	+1.4		ACK_MF	BOOL	FALSE	Acknowledge M function
21.5	+1.5		READ_EN	BOOL	FALSE	Enable read-in
21.6	+1.6		SKIP_BLK	BOOL	FALSE	Skip block
21.7	+1.7		DRV_EN	BOOL	FALSE	Drive enable
22.0	+2.0		MODE	BYTE	B#16#0	Mode
23.0	+3.0		MODE_ PARAMETER	BYTE	B#16#0	Mode parameter
24.0	+4.0		OVERRIDE	BYTE	B#16#0	Override
25.0	+5.0		BYTE5	BYTE	B#16#0	Reserved
26.0	+6.0		BYTE6	BYTE	B#16#0	Reserved
27.0	+7.0		BYTE7	BYTE	B#16#0	Reserved
	=8.0			END_STRUCT		
Checkback signals:			**FC MODE_WR**			
	28.0	stat	CHECKBACK_ SIGNALS	STRUCT		Checkback signals
28.0	+0.0		DAIN	BOOL	FALSE	Reserved
28.1	+0.1		TST_STAT	BOOL	FALSE	P bus interface switchover complete
28.2	+0.2		BIT0_2	BOOL	FALSE	Reserved
28.3	+0.3		OT_ERR	BOOL	FALSE	Operating/travel error

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
28.4	+0.4		DATA_ERR	BOOL	FALSE	Data error
28.5	+0.5		FM_NSTQ	BOOL	FALSE	Reserved
28.6	+0.6		FM_NST	BOOL	FALSE	Reserved
28.7	+0.7		PARA	BOOL	FALSE	parameterized
29.0	+1.0		START_EN	BOOL	FALSE	Start enable
29.1	+1.1		WORKING	BOOL	FALSE	Processing in progress
29.2	+1.2		WAIT_EN	BOOL	FALSE	Waiting for external enable
29.3	+1.3		BIT1_3	BOOL	FALSE	Reserved
29.4	+1.4		BIT1_4	BOOL	FALSE	Reserved
29.5	+1.5		DT_RUN	BOOL	FALSE	Dwell running
29.6	+1.6		PR_BACK	BOOL	FALSE	M21.Program scanning backward
29.7	+1.7		BIT1_7	BOOL	FALSE	Reserved
30.0	+2.0		MODE	BYTE	B#16#0	Active operating mode
31.0	+3.0		SYNC	BOOL	FALSE	synchronized
31.1	+3.1		MSR_DONE	BOOL	FALSE	End of measurement
31.2	+3.2		GO_M	BOOL	FALSE	GO_minus
31.3	+3.3		GO_P	BOOL	FALSE	GO_plus
31.4	+3.4		BIT3_4	BOOL	FALSE	Reserved
31.5	+3.5		FAVEL	BOOL	FALSE	Flying actual value done
31.6	+3.6		BIT3_6	BOOL	FALSE	Reserved
31.7	+3.7		POS_ROD	BOOL	FALSE	Position reached, Stop (“PEH”)
32.0	+4.0		NUM_MF	BYTE	B#16#0	M function No.
33.0	+5.0		BIT5_0	BOOL	FALSE	Reserved
33.1	+5.1		BIT5_1	BOOL	FALSE	Reserved
33.2	+5.2		BIT5_2	BOOL	FALSE	Reserved
33.3	+5.3		BIT5_3	BOOL	FALSE	Reserved
33.4	+5.4		STR_MF	BOOL	FALSE	Change of M function
33.5	+5.5		BIT5_5	BOOL	FALSE	Reserved
33.6	+5.6		BIT5_6	BOOL	FALSE	Reserved
33.7	+5.7		BIT5_7	BOOL	FALSE	Reserved
34.0	+6.0		WORD6	WORD	W#16#0	Reserved
36.0	+8.0		DWORD8	DWORD	DW#16#0	Reserved
	=12.0			END_STRUCT		

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
Single functions:			**FC MODE_WR, job no. 10**			
	40.0	stat	SINGLE_FUNCTIONS	STRUCT		Single functions
40.0	+0.0		SERVO_EN	BOOL	FALSE	Servo enable
40.1	+0.1		GAUG_FLY	BOOL	FALSE	Inprocess measurement
40.2	+0.2		BIT0_2	BOOL	FALSE	Reserved
40.3	+0.3		BIT0_3	BOOL	FALSE	Reserved
40.4	+0.4		BIT0_4	BOOL	FALSE	Reserved
40.5	+0.5		TRAV_MON	BOOL	FALSE	Reserved
40.6	+0.6		PARK_AX	BOOL	FALSE	Parking axis
40.7	+0.7		SIM_ON	BOOL	FALSE	Simulation on
41.0	+1.0		BIT1_0	BOOL	FALSE	Reserved
41.1	+1.1		BIT1_1	BOOL	FALSE	Reserved
41.2	+1.2		MSR_EN	BOOL	FALSE	Linear measurement
41.3	+1.3		REFTRIG	BOOL	FALSE	Retrigger reference point
41.4	+1.4		DI_EN	BOOL	FALSE	Switch off enable input
41.5	+1.5		FOLLOWUP	BOOL	FALSE	Follow–up mode
41.6	+1.6		SSW_DIS	BOOL	FALSE	SW- Switch off SW end position monitoring
41.7	+1.7		DRIFTOFF	BOOL	FALSE	Switch off automatic drift compensation
	=2.0			END_STRUCT		
Single commands:			**FC MODE_WR, job no. 11**			
	42.0	stat	SINGLE_COMMANDS	STRUCT		Single commands
42.0	+0.0		BIT0_0	BOOL	FALSE	Reserved
42.1	+0.1		BIT0_1	BOOL	FALSE	Reserved
42.2	+0.2		BIT0_2	BOOL	FALSE	Reserved
42.3	+0.3		BIT0_3	BOOL	FALSE	Reserved
42.4	+0.4		BIT0_4	BOOL	FALSE	Reserved
42.5	+0.5		BIT0_5	BOOL	FALSE	Reserved
42.6	+0.6		BIT0_6	BOOL	FALSE	Reserved
42.7	+0.7		BIT0_7	BOOL	FALSE	Reserved
43.0	+1.0		MDATA_EN	BOOL	FALSE	Activate MD
43.1	+1.1		DEL_DIST	BOOL	FALSE	Delete distance–to–go

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
43.2	+1.2		SEARCH_F	BOOL	FALSE	Automatic block search forward
43.3	+1.3		SEARCH_B	BOOL	FALSE	Automatic block search in reverse
43.4	+1.4		BIT1_4	BOOL	FALSE	Reserved
43.5	+1.5		RESET_AX	BOOL	FALSE	Restart
43.6	+1.6		AVAL_REM	BOOL	FALSE	Set actual value reset
43.7	+1.7		BIT1_7	BOOL	FALSE	Reserved
	=2.0			END_STRUCT		
Zero offset:			**FC MODE_WR, job no. 12**			
44.0	44.0	stat	ZERO_OFFSET	DINT	L#0	Zero offset
Set Actual Value:			**FC MODE_WR, job no. 13**			
48.0	48.0	stat	SETTING_ ACT_VALUE	DINT	L#0	Istwert setzen
Set actual value on the fly:			**FC MODE_WR, job no. 14**			
52.0	52.0	stat	FLYING_ SETTING_ ACT_VALUE	DINT	L#0	Set actual value on the fly
Set Reference Point:			**FC MODE_WR, job no. 21**			
56.0	56.0	stat	SETTING_ REFERENCE_ PIONT	DINT	L#0	Set reference point
Measured values:			**FC MSRMENT**			
	60.0	stat	MEASUREMENT_VALUES	STRUCT		Measured values
60.0	+0.0		BEGIN_VALUE	DINT	L#0	Begin value or measured value on the fly
64.0	+4.0		END_VALUE	DINT	L#0	End value
68.0	+8.0		LENGTH_ VALUE	DWORD	DW#16#0	Measured length value
	=12.0			END_STRUCT		
Diagnostic interrupt data:			**FC DIAG_RD/FC DIAG_INF**			
	72.0	stat	DIAGNOSTIC_ INT_INFO	STRUCT		Diagnostic interrupt data
72.0	+0.0		BYTE0	BYTE	B#16#0	For system–specific diagnostic data, see Section 6.4
73.0	+1.0		BYTE1	BYTE	B#16#0	
74.0	+2.0		BYTE2	BYTE	B#16#0	
75.0	+3.0		BYTE3	BYTE	B#16#0	
76.0	+4.0		BYTE4	BYTE	B#16#0	Channel type

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
77.0	+5.0		BYTE5	BYTE	B#16#0	Info length per channel
78.0	+6.0		BYTE6	BYTE	B#16#0	No. of channels
79.0	+7.0		BYTE7	BYTE	B#16#0	Channel error vector
80.0	+8.0		BYTE8	BYTE	B#16#0	For individual errors, see Section 6.4
81.0	+9.0		BYTE9	BYTE	B#16#0	
82.0	+10.0		BYTE10	BYTE	B#16#0	Reserved
83.0	+11.0		BYTE11	BYTE	B#16#0	Reserved
84.0	+12.0		BYTE12	BYTE	B#16#0	Reserved
85.0	+13.0		BYTE13	BYTE	B#16#0	Reserved
	=14.0			END_STRUCT		
Setpoint for increment:			**FC MODE_WR, job no. 3**			
86.0	86.0	stat	TARGET_254	DWORD	DW#16#0	Setpoint for increment
Velocity levels 1 and 2:			**FC MODE_WR, job no. 1**			
	90.0	stat	VLEVEL_1_2	STRUCT		Velocity levels 1 and 2
90.0	+0.0		VLEVEL_1	DWORD	DW#16#0	Velocity level 1
94.0	+4.0		VLEVEL_2	DWORD	DW#16#0	Velocity level 2
	=8.0			END_STRUCT		
Voltage levels 1 and 2			**FC MODE_WR, job no. 2**			
	98.0	stat	CLEVEL_1_2	STRUCT		Voltage levels 1 and 2
98.0	+0.0		CLEVEL_1	DWORD	DW#16#0	Voltage level 1
102.0	+4.0		CLEVEL_2	DWORD	DW#16#0	Voltage level 2
	=8.0			END_STRUCT		
MDI block:			**FC MODE_WR, job no. 6**			
	106.0	stat	MDI_BLOCK	STRUCT		MDI block
106.0	+0.0		BYTE0	BYTE	B#16#0	Reserved
107.0	+1.0		BYTE1	BYTE	B#16#0	
108.0	+2.0		G_1_EN	BOOL	FALSE	G function group 1
108.1	+2.1		G_2_EN	BOOL	FALSE	G function group 2
108.2	+2.2		BIT2_2	BOOL	FALSE	Reserved
108.3	+2.3		BIT2_3	BOOL	FALSE	Reserved
108.4	+2.4		X_T_EN	BOOL	FALSE	Position/dwell
108.5	+2.5		BIT2_5	BOOL	FALSE	Reserved
108.6	+2.6		BIT2_6	BOOL	FALSE	Reserved
108.7	+2.7		BIT2_7	BOOL	FALSE	Reserved
109.0	+3.0		V_EN	BOOL	FALSE	Velocity

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
109.1	+3.1		M_1_EN	BOOL	FALSE	M function group 1
109.2	+3.2		M_2_EN	BOOL	FALSE	M function group 2
109.3	+3.3		M_3_EN	BOOL	FALSE	M function group 3
109.4	+3.4		BIT3_4	BOOL	FALSE	Reserved
109.5	+3.5		BIT3_5	BOOL	FALSE	Reserved
109.6	+3.6		BIT3_6	BOOL	FALSE	Reserved
109.7	+3.7		BIT3_7	BOOL	FALSE	Reserved
110.0	+4.0		G_1_VAL	BYTE	B#16#0	G function no. of group 1
111.0	+5.0		G_2_VAL	BYTE	B#16#0	G function no. of group 2
112.0	+6.0		BYTE6	BYTE	B#16#0	Reserved
113.0	+7.0		BYTE7	BYTE	B#16#0	Reserved
114.0	+8.0		X_T_VAL	DINT	L#0	Value of position/dwell
118.0	+12.0		V_VAL	DINT	L#0	Velocity value
122.0	+16.0		M_1_VAL	BYTE	B#16#0	M function no. of group 1
123.0	+17.0		M_2_VAL	BYTE	B#16#0	M function no. of group 2
124.0	+18.0		M_3_VAL	BYTE	B#16#0	M function no. of group 3
125.0	+19.0		BYTE19	BYTE	B#16#0	Reserved
	=20.0			END_STRUCT		
Change parameters/data			**FC MODE_WR, job no. 8**			
	126.0	stat	PAR_CHAN	STRUCT		Change parameters/data
126.0	+0.0		PAR_TYP	BYTE	B#16#0	DB type
127.0	+1.0		PAR_NUMB	BYTE	B#16#0	Number
128.0	+2.0		PAR_COUN	BYTE	B#16#0	Quantity
129.0	+3.0		PAR_JOB	BYTE	B#16#0	Job
130.0	+4.0		PAR_DATA	BYTE	B#16#0	Data field
131.0	+5.0		BYTE5		B#16#0	
132.0	+6.0		BYTE6		B#16#0	
133.0	+7.0		BYTE7		B#16#0	
134.0	+8.0		BYTE8		B#16#0	
135.0	+9.0		BYTE9		B#16#0	
136.0	+10.0		BYTE10		B#16#0	
137.0	+11.0		BYTE11		B#16#0	
138.0	+12.0		BYTE12		B#16#0	

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
139.0	+13.0		BYTE13		B#16#0	
140.0	+14.0		BYTE14		B#16#0	
141.0	+15.0		BYTE15		B#16#0	
142.0	+16.0		BYTE16		B#16#0	
143.0	+17.0		BYTE17		B#16#0	
144.0	+18.0		BYTE18		B#16#0	
145.0	+19.0		BYTE19		B#16#0	
146.0	+20.0		BYTE20		B#16#0	
147.0	+21.0		BYTE21		B#16#0	
148.0	+22.0		BYTE22		B#16#0	
149.0	+23.0		BYTE23		B#16#0	
	=24.0			END_STRUCT		
Digital I/Os			**FC MODE_WR, job no. 15/FC RD_COM, job no. 101**			
	150.0	stat	DIG_IO	STRUCT		Digital inputs and outputs
150.0	+0.0		D_IN0	BOOL	FALSE	Digital input 0
150.1	+0.1		D_IN1	BOOL	FALSE	Digital input 1
150.2	+0.2		D_IN2	BOOL	FALSE	Digital input 2
150.3	+0.3		D_IN3	BOOL	FALSE	Digital input 3
150.4	+0.4		BIT0_4	BOOL	FALSE	Reserved
150.5	+0.5		BIT0_5	BOOL	FALSE	Reserved
150.6	+0.6		BIT0_6	BOOL	FALSE	Reserved
150.7	+0.7		BIT0_7	BOOL	FALSE	Reserved
151.0	+1.0		D_OUT0	BOOL	FALSE	Digital output 0
151.1	+1.1		D_OUT1	BOOL	FALSE	Digital output 1
151.2	+1.2		D_OUT2	BOOL	FALSE	Digital output 2
151.3	+1.3		D_OUT3	BOOL	FALSE	Digital output 3
151.4	+1.4		BIT1_4	BOOL	FALSE	Reserved
151.5	+1.5		BIT1_5	BOOL	FALSE	Reserved
151.6	+1.6		BIT1_6	BOOL	FALSE	Reserved
151.7	+1.7		BIT1_7	BOOL	FALSE	Reserved
	=2.0			END_STRUCT		
MDI block on the fly:			**FC MODE_WR, job no. 16**			
	152.0	stat	MDI_FLY	STRUCT		MDI block on the fly
152.0	+0.0		BYTE0	BYTE	B#16#0	Reserved
153.0	+1.0		BYTE1	BYTE	B#16#0	Reserved

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
154.0	+2.0		G_1_EN	BOOL	FALSE	G function group 1
154.1	+2.1		G_2_EN	BOOL	FALSE	G function group 2
154.2	+2.2		BIT2_2	BOOL	FALSE	Reserved
154.3	+2.3		BIT2_3	BOOL	FALSE	Reserved
154.4	+2.4		X_T_EN	BOOL	FALSE	Position/dwell
154.5	+2.5		BIT2_5	BOOL	FALSE	Reserved
154.6	+2.6		BIT2_6	BOOL	FALSE	Reserved
154.7	+2.7		BIT2_7	BOOL	FALSE	Reserved
155.0	+3.0		V_EN	BOOL	FALSE	Velocity
155.1	+3.1		M_1_EN	BOOL	FALSE	M function group 1
155.2	+3.2		M_2_EN	BOOL	FALSE	M function group 2
155.3	+3.3		M_3_EN	BOOL	FALSE	M function group 3
155.4	+3.4		BIT3_4	BOOL	FALSE	Reserved
155.5	+3.5		BIT3_5	BOOL	FALSE	Reserved
155.6	+3.6		BIT3_6	BOOL	FALSE	Reserved
155.7	+3.7		BIT3_7	BOOL	FALSE	Reserved
156.0	+4.0		G_1_VAL	BYTE	B#16#0	G function no. 1
157.0	+5.0		G_2_VAL	BYTE	B#16#0	G function no. 2
158.0	+6.0		BYTE6	BYTE	B#16#0	Reserved
159.0	+7.0		BYTE7	BYTE	B#16#0	Reserved
160.0	+8.0		X_T_VAL	DINT	L#0	Value of position/dwell
164.0	+12.0		V_VAL	DINT	L#0	Velocity value
168.0	+16.0		M_1_VAL	BYTE	B#16#0	M function no. of group 1
169.0	+17.0		M_2_VAL	BYTE	B#16#0	M function no. of group 2
170.0	+18.0		M_3_VAL	BYTE	B#16#0	M function no. of group 3
171.0	+19.0		BYTE19	BYTE	B#16#0	Reserved
	=20.0			END_STRUCT		
Select program			**FC MODE_WR, job no. 17**			
	172.0	stat	PROG_SEL	STRUCT		Select program
172.0	+0.0		PROG_NO	BYTE	B#16#0	Program number
173.0	+1.0		BLCK_NO	BYTE	B#16#0	Block number
174.0	+2.0		PROG_DIR	BYTE	B#16#0	Direction of machining
175.0	+3.0		BYTE3	BYTE	B#16#0	Reserved

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
	=4.0			END_STRUCT		
Request application data:			**FC MODE_WR, job no. 18**			
	176.0	stat	REQ_APP	STRUCT		Request application data
176.0	+0.0		CODE_AP1	BYTE	B#16#0	Application data 1
177.0	+1.0		CODE_AP2	BYTE	B#16#0	Application data 2
178.0	+2.0		CODE_AP3	BYTE	B#16#0	Application data 3
179.0	+3.0		CODE_AP4	BYTE	B#16#0	Application data 4
	=4.0			END_STRUCT		
Teach In:			**FC MODE_WR, job no. 19**			
	180.0	stat	TEACH_IN	STRUCT		Teach In
180.0	+0.0		PROG_NO	BYTE	B#16#0	Program number
181.0	+1.0		BLCK_NO	BYTE	B#16#0	Block number
	=2.0			END_STRUCT		
FC MODE_WR, job no. 22						
	182.0	stat	SRV_IN	STRUCT		Reserved
182.0	+0.0		SRV_IN1	DINT	L#0	
186.0	+4.0		SRV_IN2	DINT	L#0	
190.0	+8.0		SRV_IN3	DINT	L#0	
194.0	+12.0		SRV_IN4	DINT	L#0	
	=16.0			END_STRUCT		
basic operating data			**FC RD_COM, job no. 102**			
	198.0	stat	OP_DAT	STRUCT		Basic operating data
198.0	+0.0		ACT_VAL	DINT	L#0	Actual position
202.0	+4.0		SPEED	DWORD	DW#16#0	Actual speed
206.0	+8.0		REM_DIST	DINT	L#0	Distance to go
210.0	+12.0		SET_POS	DINT	L#0	Setpoint
214.0	+16.0		SUM_OFST	DINT	L#0	Total of active coordinate shifts for tool offset, zero offset
218.0	+20.0		TRAV_SPE	DWORD	DW#16#0	Speed
222.0	+24.0		DWORD24	DINT	L#0	Reserved
226.0	+28.0		DWORD28	DINT	L#0	Reserved
	=32.0			END_STRUCT		
Active NC block:			**FC RD_COM, job no. 103**			
	230.0	stat	ACT_BLCK	STRUCT		Active NC block
230.0	+0.0		PROG_NO	BYTE	B#16#0	Program number

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
231.0	+1.0		BLCK_NO	BYTE	B#16#0	Block number
232.0	+2.0		G_1_EN	BOOL	FALSE	G function group 1
232.1	+2.1		G_2_EN	BOOL	FALSE	G function group 2
232.2	+2.2		G_3_EN	BOOL	FALSE	G function group 3
232.3	+2.3		BIT2_3	BOOL	FALSE	Reserved
232.4	+2.4		X_T_EN	BOOL	FALSE	Position/dwell
232.5	+2.5		SR_L_EN	BOOL	FALSE	No. of SR calls
232.6	+2.6		SR_N_EN	BOOL	FALSE	SR call
232.7	+2.7		SKIP_EN	BOOL	FALSE	Skip block
233.0	+3.0		V_EN	BOOL	FALSE	Velocity
233.1	+3.1		M_1_EN	BOOL	FALSE	M function group 1
233.2	+3.2		M_2_EN	BOOL	FALSE	M function group 2
233.3	+3.3		M_3_EN	BOOL	FALSE	M function group 3
233.4	+3.4		TO_EN	BOOL	FALSE	Tool offset
233.5	+3.5		BIT3_5	BOOL	FALSE	Reserved
233.6	+3.6		BIT3_6	BOOL	FALSE	Reserved
233.7	+3.7		BIT3_7	BOOL	FALSE	Reserved
234.0	+4.0		G_1_VAL	BYTE	B#16#0	G function no. of group 1
235.0	+5.0		G_2_VAL	BYTE	B#16#0	G function no. of group 2
236.0	+6.0		G_3_VAL	BYTE	B#16#0	G function no. of group 3
237.0	+7.0		BYTE7	BYTE	B#16#0	Reserved
238.0	+8.0		X_T_VAL	DINT	L#0	Value
242.0	+12.0		V_VAL	DINT	L#0	Value
246.0	+16.0		M_1_VAL	BYTE	B#16#0	M function no. of group 1
247.0	+17.0		M_2_VAL	BYTE	B#16#0	M function no. of group 2
248.0	+18.0		M_3_VAL	BYTE	B#16#0	M function no. of group 3
249.0	+19.0		TO_VAL	BYTE	B#16#0	Tool offset no.
	=20.0			END_STRUCT		
Next NC block:			**FC RD_COM, job no. 104**			
	250.0	stat	NXT_BLCK	STRUCT		Next NC block
250.0	+0.0		PROG_NO	BYTE	B#16#0	Program number
251.0	+1.0		BLCK_NO	BYTE	B#16#0	Block number
252.0	+2.0		G_1_EN	BOOL	FALSE	G function group 1

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
252.1	+2.1		G_2_EN	BOOL	FALSE	G function group 2
252.2	+2.2		G_3_EN	BOOL	FALSE	G function group 3
252.3	+2.3		BIT2_3	BOOL	FALSE	Reserved
252.4	+2.4		X_T_EN	BOOL	FALSE	Position/dwell
252.5	+2.5		SR_L_EN	BOOL	FALSE	No. of SR calls
252.6	+2.6		SR_N_EN	BOOL	FALSE	SR call
252.7	+2.7		SKIP_EN	BOOL	FALSE	Skip block
253.0	+3.0		V_EN	BOOL	FALSE	Velocity
253.1	+3.1		M_1_EN	BOOL	FALSE	M function group 1
253.2	+3.2		M_2_EN	BOOL	FALSE	M function group 2
253.3	+3.3		M_3_EN	BOOL	FALSE	M function group 3
253.4	+3.4		TO_EN	BOOL	FALSE	Tool offset
253.5	+3.5		BIT3_5	BOOL	FALSE	Reserved
253.6	+3.6		BIT3_6	BOOL	FALSE	Reserved
253.7	+3.7		BIT3_7	BOOL	FALSE	Reserved
254.0	+4.0		G_1_VAL	BYTE	B#16#0	G function no. of group 1
255.0	+5.0		G_2_VAL	BYTE	B#16#0	G function no. of group 2
256.0	+6.0		G_3_VAL	BYTE	B#16#0	G function no. of group 3
257.0	+7.0		BYTE7	BYTE	B#16#0	Reserved
258.0	+8.0		X_T_VAL	DINT	L#0	Value
262.0	+12.0		V_VAL	DINT	L#0	Value
266.0	+16.0		M_1_VAL	BYTE	B#16#0	M function no. of group 1
267.0	+17.0		M_2_VAL	BYTE	B#16#0	M function no. of group 2
268.0	+18.0		M_3_VAL	BYTE	B#16#0	M function no. of group 3
269.0	+19.0		TO_VAL	BYTE	B#16#0	Tool offset no.
	=20.0			END_STRUCT		
Application data:			**FC RD_COM, job no. 105**			
	270.0	stat	APP_DAT	STRUCT		Application data
270.0	+0.0		APP1	DINT	L#0	Application data 1
274.0	+4.0		APP2	DINT	L#0	Application data 2
278.0	+8.0		APP3	DINT	L#0	Application data 3
282.0	+12.0		APP4	DINT	L#0	Application data 4
	=16.0			END_STRUCT		

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
Actual value block change			**FC RD_COM, job no. 107**			
286.0	286.0	stat	BLCK_EXT	DWORD	DW#16#0	Actual value block change
Servicing data			**FC RD_COM, job no. 108**			
	290.0	stat	SERV_DAT	STRUCT		Servicing data
290.0	+0.0		OUT_VAL	DINT	L#0	DAC output value
294.0	+4.0		ENC_VAL	DINT	L#0	Actual encoder value
298.0	+8.0		PULS_ERR	DINT	L#0	Pulse errors
302.0	+12.0		KV_FA	DINT	L#0	K_v factor
306.0	+16.0		FOLL_ERR	DINT	L#0	Following error
310.0	+20.0		FERR_LIM	DINT	L#0	Following error limit
314.0	+24.0		OSC_ERR	DINT	L#0	s overshoot value/ switch adjustment
318.0	+28.0		DR_TIME	DINT	L#0	Positioning time/ response time constant
	=32.0			END_STRUCT		
FC RD_COM, job no. 109						
	322.0	stat	SRV_OUT	STRUCT		Reserved
322.0	+0.0		SRV_OUT1	DINT	L#0	
326.0	+4.0		SRV_OUT2	DINT	L#0	
330.0	+8.0		SRV_OUT3	DINT	L#0	
334.0	+12.0		SRV_OUT4	DINT	L#0	
338.0	+16.0		SRV_OUT5	DINT	L#0	
342.0	+20.0		SRV_OUT6	DINT	L#0	
346.0	+24.0		SRV_OUT7	DINT	L#0	
350.0	+28.0		SRV_OUT8	DINT	L#0	
	=32.0			END_STRUCT		
Additional operating data:			**FC RD_COM, job no. 110**			
	354.0	stat	OP_DAT1	STRUCT		Additional operating data
354.0	+0.0		OVERRIDE	BYTE	B#16#0	Override
355.0	+1.0		PROG_NO	BYTE	B#16#0	NC traverse program no.
356.0	+2.0		BLCK_NO	BYTE	B#16#0	NC block no.
357.0	+3.0		LOOP_NO	BYTE	B#16#0	Callup SR loop no. counter
358.0	+4.0		G90_91	BYTE	B#16#0	Active G90/91
359.0	+5.0		G60_64	BYTE	B#16#0	Active G60/64
360.0	+6.0		G43_44	BYTE	B#16#0	Active G43/44

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
361.0	+7.0		TO_NO	BYTE	B#16#0	Active D no. see
362.0	+8.0		BIT8_0	BOOL	FALSE	Reserved
362.1	+8.1		LIM_SP	BOOL	FALSE	Velocity limitation
362.2	+8.2		LIM_10	BOOL	FALSE	Limitation to ± 10 V
362.3	+8.3		LIM_SU	BOOL	FALSE	Limitation of the minimum acceleration or deceleration
362.4	+8.4		BIT8_4	BOOL	FALSE	Reserved
362.5	+8.5		BIT8_5	BOOL	FALSE	Reserved
362.6	+8.6		BIT8_6	BOOL	FALSE	Reserved
362.7	+8.7		BIT8_7	BOOL	FALSE	Reserved
363.0	+9.0		LIM_FR	BOOL	FALSE	Reserved
363.1	+9.1		LIM_FV	BOOL	FALSE	Reserved
363.2	+9.2		BIT9_2	BOOL	FALSE	Reserved
363.3	+9.3		LIM_FS	BOOL	FALSE	Reserved
363.4	+9.4		BIT9_4	BOOL	FALSE	Reserved
363.5	+9.5		BIT9_5	BOOL	FALSE	Reserved
363.6	+9.6		BIT9_6	BOOL	FALSE	Reserved
363.7	+9.7		BIT9_7	BOOL	FALSE	Reserved
364.0	+10.0		BYTE10	BYTE	B#16#0	Reserved
365.0	+11.0		BYTE11	BYTE	B#16#0	Reserved
	=12.0			END_STRUCT		
Parameters/data:			**FC RD_COM, job no. 114**			
	366.0	stat	PAR_READ	STRUCT		Parameters/data
366.0	+0.0		PAR_TYP	BYTE	B#16#0	DB-Typ
367.0	+1.0		PAR_NO	BYTE	B#16#0	Number
368.0	+2.0		PAR_COUN	BYTE	B#16#0	Quantity
369.0	+3.0		BYTE3	BYTE	B#16#0	Reserved
370.0	+4.0		PAR_DATA	BYTE	B#16#0	Data field
371.0	+5.0		BYTE5	BYTE	B#16#0	
372.0	+6.0		BYTE6	BYTE	B#16#0	
373.0	+7.0		BYTE7	BYTE	B#16#0	
374.0	+8.0		BYTE8	BYTE	B#16#0	
375.0	+9.0		BYTE9	BYTE	B#16#0	
376.0	+10.0		BYTE10	BYTE	B#16#0	

Table 6-5 User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
377.0	+11.0		BYTE11	BYTE	B#16#0	
378.0	+12.0		BYTE12	BYTE	B#16#0	
379.0	+13.0		BYTE13	BYTE	B#16#0	
380.0	+14.0		BYTE14	BYTE	B#16#0	
381.0	+15.0		BYTE15	BYTE	B#16#0	
382.0	+16.0		BYTE16	BYTE	B#16#0	
383.0	+17.0		BYTE17	BYTE	B#16#0	
384.0	+18.0		BYTE18	BYTE	B#16#0	
385.0	+19.0		BYTE19	BYTE	B#16#0	
386.0	+20.0		BYTE22	BYTE	B#16#0	
387.0	+21.0		BYTE23	BYTE	B#16#0	
388.0	+22.0		BYTE24	BYTE	B#16#0	
389.0	+23.0		BYTE23	BYTE	B#16#0	
	=24.0			END_STRUCT		
Human-machine interface						
	390.0	stat	USR_CON	STRUCT		Man-machine interface
390.0	+0.0		BITC_0	BOOL	FALSE	Write MD
390.1	+0.1		BITC_1	BOOL	FALSE	Read MD
390.2	+0.2		BITC_2	BOOL	FALSE	Transfer MDI block
390.3	+0.3		BITC_3	BOOL	FALSE	Transfer program selection
390.4	+0.4		BITC_4	BOOL	FALSE	Transfer Teach In
390.5	+0.5		BITC_5	BOOL	FALSE	Transfer increment
390.6	+0.6		BITC_6	BOOL	FALSE	Transfer velocity levels
390.7	+0.7		BITC_7	BOOL	FALSE	Transfer voltage levels
391.0	+1.0		BITC_8	BOOL	FALSE	Transfer MDI block on the fly
391.1	+1.1		BITC_9	BOOL	FALSE	Transfer actual value setting
391.2	+1.2		BITC_10	BOOL	FALSE	Transfer zero offset
391.3	+1.3		BITC_11	BOOL	FALSE	Reserved
391.4	+1.4		BITC_12	BOOL	FALSE	Reserved
391.5	+1.5		BITC_13	BOOL	FALSE	Diagnostic interrupt
391.6	+1.6		BITC_14	BOOL	FALSE	Data error
391.7	+1.7		BITC_15	BOOL	FALSE	Operator/travel error
	=2.0			END_STRUCT		
392.0	392.0	stat	MD_NO	WORD	W#16#0	MD-Nr.
394.0	394.0	stat	MD_VALUE	DINT	L#0	MD value

Table 6-5   User DB for the FM 354, continued

Absolute address	Relative address	Declaration	Variable	Data type	Initial value	Comments
398.0	398.0	stat	INC_NO	BYTE	B#16#0	INC No.
399.0	399.0	stat	RESERV_3	BYTE	B#16#0	Reserved
400.0	400.0	stat	PICT_NO	WORD	W#16#0	Picture number
402.0	402.0	stat	KEY_CODE	WORD	W#16#0	Keyboard code
404.0	404.0	stat	RESERV_4	WORD	W#16#0	Reserved
	406.0	stat	OP_MODE	STRUCT		Selection of operating mode
406.0	+0.0		BITA_0	BOOL	FALSE	Mode = Control
406.1	+0.1		BITA_1	BOOL	FALSE	Reference-point approach
406.2	+0.2		BITA_2	BOOL	FALSE	Rel. increment travel
406.3	+0.3		BITA_3	BOOL	FALSE	MDI
406.4	+0.4		BITA_4	BOOL	FALSE	Automatic/single block
406.5	+0.5		BITA_5	BOOL	FALSE	Automatic
406.6	+0.6		BITA_6	BOOL	FALSE	Tipping
406.7	+0.7		BITA_7	BOOL	FALSE	Reserved
407.0	+1.0		BITA_8	BOOL	FALSE	Reserved
407.1	+1.1		BITA_9	BOOL	FALSE	Reserved
407.2	+1.2		BITA_10	BOOL	FALSE	Reserved
407.3	+1.3		BITA_11	BOOL	FALSE	Reserved
407.4	+1.4		BITA_12	BOOL	FALSE	Reserved
407.5	+1.5		BITA_13	BOOL	FALSE	Reserved
407.6	+1.6		BITA_14	BOOL	FALSE	Acknowledge error
407.7	+1.7		BITA_15	BOOL	FALSE	Acknowledge diagnostic interrupt
	=2.0			END_STRUCT		

6.7 Example applications

Example 1

See STEP 7 example application FMSTSVEX\EXAMPLE1

The following blocks are required, in addition to the technology functions, in order to run this example application:

- DB 1 (user DB), FC 100 (example call)
- (cycle) and OB 100 (cold restart)

The following operating modes are supported in example 1:

- Jogging
- Reference-point approach
- MDI block

The associated data (velocity levels, MDI block, single functions) are transferred automatically to the FM after Power **On** or when the CPU switches from STOP to RUN. These values can be transferred again by setting the appropriate write memory (M17.4 to M17.6).

OB 100 contains certain default settings for velocity levels, MDI block, single functions (servo enable, simulation), operating mode (Jogging mode is active on start), mode parameters and override; however these can be changed according to the application.

Table 6-6 Memories: example application 1

Input memories used
M16.0 Start
M16.1 Stop
M16.2 Direction minus
M16.3 Direction plus
M16.4 Not used
M 16.5 Not used
M 16.6 Not used
M 16.7 Drive enable
M 17.0 Not used
M 17.1 Acknowledge operator/travel error
M 17.2 Mode selection
M 17.3 Not used
M17.4 Transfer velocity levels
M17.5 Transfer MDI block
M17.6 Transfer single functions
M 17.7 Not used
MB 18 Operating mode (encoded)
MB19 Override

Example 2

See STEP 7 example application FMSTSVEX\EXAMPLE2

The following blocks are required, in addition to the technology functions, in order to run this example application:

- DB 1 (user DB), FC 100 (example call),
- OB 1 (cycle), OB 40 (process interrupt), OB 82 (diagnostic interrupt) and OB 100 (cold restart).

The following operating modes are supported in the example:

- Jogging
- Reference-point approach
- MDI block
- Automatic

The associated data (velocity levels, MDI block, single functions, single commands and program selection) are transferred automatically to the FM when the appropriate write memories are set. If no action is performed on the write job memories (M17.4 to M17.7), only control/checkback signals are transferred.

If the memory “READ DATA” (M17.3) is set, data are read (basic operating data).

You can acknowledge a diagnostic interrupt by setting the memory “RE-START” (M17.0).

An example for calling data set DS 162 (evaluation of operator/travel errors) is provided for special error evaluation at the end of FC 100. Data set 163 (evaluation of data errors) and data set 164 (evaluation of operating errors) are called similarly.

OB 100 contains certain default settings for velocity levels, MDI block, servo enable, simulation), operating mode (Jogging mode is active on start), mode parameters and override (100 %); however these can be changed according to the application.

Table 6-7 Memories: example application 2

Input memories used	Output memories used
M16.0 Start	M20.0 Free
M16.1 Stop	M20.1 Operator/travel errors
M16.2 Direction minus	M20.2 Data error
M16.3 Direction plus	M20.3 Channel parameterized
M16.4 Not used	M20.4 Start enable
M16.5 Read–in enable	M20.5 Processing in progress
M16.6 Skip block	M20.6 Not used
M16.7 Drive enable	M20.7 Dwell in progress
M17.0 Restart	M21.0 Program scanning backward
M17.1 Acknowledge operation/error	M21.1 Synchronized
M17.2 Operating mode selection active	M21.2 Free
M17.3 Read data	M21.3 Travel minus

Table 6-7 Memories: example application 2, continued

Input memories used	Output memories used
M17.4 Transfer velocity levels	M21.4 Travel plus
M17.5 Transfer MDI block	M21.5 Not used
M17.6 Transfer single functions	M21.6 Position reached, stop
M17.7 Transfer program selection	M21.7 Free
MB18 Operating mode (encoded)	MB22 Active mode
MB19 Override	MB23 Not used

Example call for DS 162

The call for data sets 163 and 164 is programmed in the same way as the DS 162 call.

```
STL                                                 Explanation
VAR_TEMP
       R_DS162        : BOOL;                       // Auxiliary bit for data set
       REQ            : BOOL;                       // Parameter for SFC 59
       IOID           : BYTE;                       //
       LADDR          : WORD;                       //
       RECNUM         : BYTE;                       //
       DSNR           : BYTE;                       //
       BUSY           : BOOL;                       //
END_VAR

BEGIN

...

NETWORK                                             // Network
TITLE = DS162 LESEN

       UN    DB_FM.CHECKBACK_SIGNALS.OT_ERR;        // Only read DS162 if operating error
       SPB   NW5E;                                  // otherwise jump to end of network
       U     R_DS162;                               // If read job already active,
       SPB   D162;                                  // jump to call
INI1:  L     162;                                   // Absolute data set number
       T     DSNR;
       S     R_DS162;                               // Set read job

D162:  CALL  SFC 59 (                               // Read operator control/guidance error
             REQ      := TRUE,                      no (DS162)
             IOID     := B#16#54,                   // Request
             LADDR    := DB1.DBW12,                 // IOID
             RECNUM   := DSNR,                      // Module address
             RECORD   := P#M30.0 BYTE 4,            // Data set number
             BUSY     := BUSY,                      // Pointer (error no. in memory word
             RET_VAL  := FEHLERCODE_LESEN);         30)
       UN    BUSY;                                  // Busy
       R     R_DS162;                               // Return value
       UN    BIE;                                   // If read job finished,
       S     FEHLER_LESEFKT;                        // reset read job DS162
NW5E:  NOP   0;                                     // Binary result
                                                    // Display read function error
...
```

Example 3

See STEP 7 example application FMSTSVEX\EXAMPLE3

The following blocks are required, in addition to the technology functions, in order to run this example application:

- DB 1 (user DB), FC 100 (example call),
- OB 1 (cycle), OB 100 (cold restart).

When you set memory M16.0 (P bus interface switchover), the job is transferred to the FM by means of control signals. If memory M20.0 is enabled, the job was executed successfully on the FM. You can now start up, test and optimize the FM with the “Parameterize FM 354” tool.

Table 6-8 Memory: example application 3

Input memories used	Output memories used
M16.0 Switch P bus interface to “start–up”	M20.0 Switchover to P bus interface done
M16.1 Not used	M20.1 Not used
M16.2 Not used	M20.2 Not used
M16.3 Not used	M20.3 Not used
M16.4 Not used	M20.4 Not used
M 16.5 Not used	M20.5 Not used
M 16.6 Not used	M20.6 Not used
M16.7 Not used	M20.7 Not used
MB17 Not used	MB21 Not used

6.8 Technical specifications

Memory allocation

The following table gives you an overview of the memory allocated to FCs.

Table 6-9 Memory allocated to FCs

No.	FC	Block in bytes	MC7 code in bytes	Local data in bytes
1	INIT_DB	224	120	4
2	MODE_WR	1226	970	26
3	RD_COM	774	584	24
4	DIAG_RD	302	180	34
5	MSRMENT	288	172	26
6	DIAG_INF	282	166	26

Processing times

The following average processing times for FCs were measured with a CPU 314. The specified times are rounded:

Table 6-10 Processing times of FCs

FC	Transfer	Cycle 1	Cycle 2	Cycle 3
INIT_DB	–	–	–	–
MODE_WR	Write control/checkback signals without data (job = 0)	1.3 ms	–	–
	Write control/checkback signals with data (job > 1)	1.3 ms	3.0 ms	1.3 ms
RD_COM	Read data	3.0 ms	–	–
DIAG_RD MSRMENT DIAG_INF	Read process and diagnostic interrupt data	3.0 ms	–	–

Note

In distributed configurations (under development) it is possible to increase the number of cycles.

Starting up the FM 354

7

Overview

This Chapter introduces you to the user interface for testing and start–up, and provides check lists for starting up the positioning module. The checklists will help you:

- Check all steps until the module is running.
- Prevent malfunctions of the module once it is in operation.

You are guided through start–up of the machine axes.

Chapter overview

7.1 Installation and wiring

Installation information

You can find information about how to install your module:

- In Chapter 3 of this manual
- Inthe manual *S7-300 Programmable Controller, Hardware and Installation*

Wiring information

You can find information about how to wire your module:

- In Chapter 4 of this manual
- In the manual *S7-300 Programmable Controller, Hardware and Installation*

Checklist

The checklist below will help you check important steps in the installation and parameterization of the FM 354 positioning module.

Table 7-1 Installation and wiring checklist

Step	Check	What to do:	OK
1	Slots	Plug the module into one of the suitable slots.	
2	Shielding	Check the shielding of the FM 354 positioning module: • To ensure proper shielding, the module must be screwed down firmly on the rail. • The shielding for shielded lines for digital I/O modules must be connected to the shielding terminal element. • The shielding for the setpoint cable should not be grounded on the drive-unit end.	
3	Limit switches	Check the start/stop limit switches. The limit-switch connections must be connected to the power section. The start/stop limit switches should not be connected to the digital inputs.	
4	Parameterize	Make sure the FM 354 positioning module setup is consistent with the parameterization. Check in particular that: • The attached encoder matches the machine data. • The wiring of the digital I/O modules matches the machine data.	

7.2 Initial values for testing and optimization

Parameterization information

You can find information about parameterization:

- In Chapter 5 of this manual
- In the on-line help in "Parameterize FM 354"

Overview

The following opening display appears in the "Parameterize FM 354" tool:

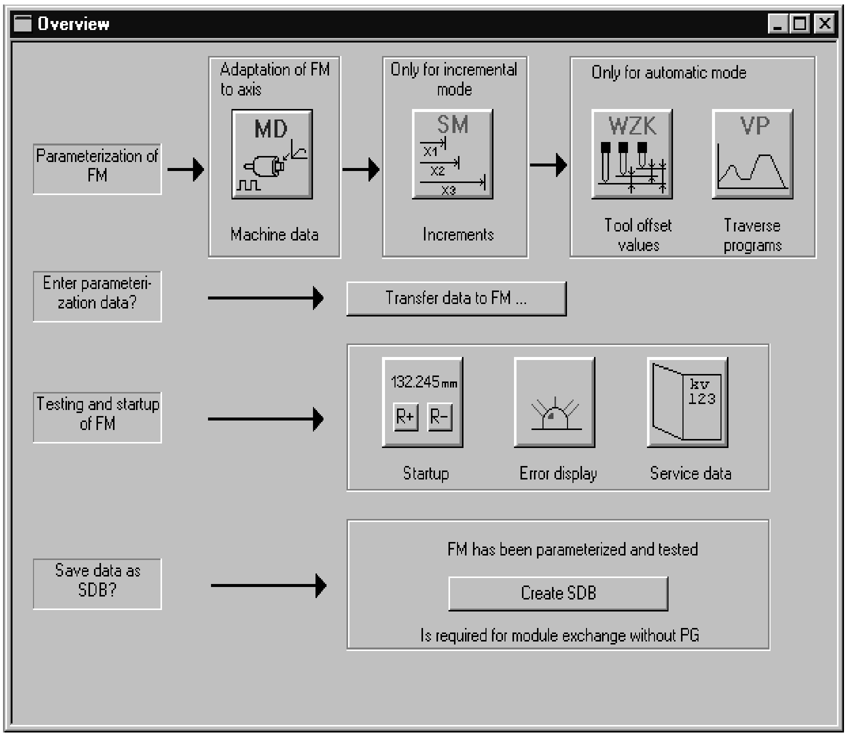


Fig. 7-1 Overview display for parameterization and start–up

You can return to this display at any point during parameterization by selecting the menu **View ▸ Overview**.

As it is written to the FM 354, the DB-MD is checked for the input limits of the individual values and their interdependencies. It is then stored only if all values are allowed. Otherwise data error messages are displayed by way of the MPI. A defective DB will not be retained when the power is turned off.

Checklist

Despite the "acceptance" testing just mentioned, the ultimate responsibility for the accuracy of all machine data lies with the module user. So it is highly advisable to perform startup using the following checklist.

Table 7-2 Parameterization checklist

Step	Check	What to do:	OK
1	Machine data	**Set initial machine data contents** As shown in Table 5-5 machine data are subdivided into configuration data (**K**) and setting data (**E**). K data indicates how the FM 354 is connected to the machine axis or CPU user program, and must therefore already be fully set up before startup begins. E data is intended for changes during startup, and serves to optimize FM 354 response for the technological process of positioning. The values in Table 7-3 are recommended, and sometimes necessary, as initial settings.	
2	Increments	Increments are only needed for the "Relative incremental" mode. For the next part of the startup procedure it is helpful to set up an "Increments" data block (DB-SM) with the following values: Value 1 1 MSR Value 2 10 MSR Value 3 100 MSR Value 4 1,000 MSR Value 5 10,000 MSR with rotary axes: Value 6 1 rotary-axis cycle (MSR) MSR = measurement-system grid	
3	Tool offset data	Tool offset data is needed only for the "Automatic" mode and is not necessary for the startup described here. Generally, it is not needed until you start up the user program on the S7-300 CPU.	
4	Traversing programs	Traversing programs are needed only for the "Automatic" mode and are not necessary for the startup described here. Generally, it is not needed until you start up the user program on the S7-300 CPU.	
5	Create SDB ≥ 1 000	When you have completed **all start–up actions** on the FM 354 and your plant, create, save and load SDB ≥ 1 000 into the CPU/onto the memory card of the CPU. All the parameter data (DBs) of the FM 354 are stored in SDB ≥ 1 000. This SDB allows you to replace the FM 354 module in the event of a fault, and to download the parameters without a programming device/PC.	

Note

The measurement system (MD7) must match the measurement system specified in the other DBs.

The measurement system raster (MSR) is the smallest distance unit in the active system of measurement.

If at some point you have failed to take this precaution:

1. Delete all data blocks (which do not match the measurement system) or clear the memory of the FM 354 completely.
2. Modify the other data blocks on the PG.
3. Reload the data blocks to the FM 354.

Initial contents of MD

The table below shows you what initial contents are recommended or required for the E machine data at startup of the machine axis.

Enter the machine data in the tab windows or in table format in accordance with the following table.

Table 7-3 Initial contents of machine data

MD (E)	Value	Explanation
5	0	FM 354 triggers no process interrupts
16	$-10^9 ... +10^9$ [MSR]	Specified reference-point coordinates
17	0	Absolute encoder readjustment value (absolute encoders only). Entered automatically when the reference point of the FM 354 is set.
21/22	$-10^9/+10^9$ [MSR]	Software limit switches inactive
23[1)]	$v_{max} = 10 ... 5 \cdot 10^8$ (MSR/min)	Specified maximum axis speed
24	1 000 [MSR]	Large PEH target range
25	0	PEH time monitoring switched off
26	10 000 [MSR]	Zero speed range monitoring set to default value (If the “zero speed range” monitoring error occurs during the start–up, as described in Section 7.3, please enter the maximum value of 1 000 000 deviating from the default. You then define the value required for your true drive axis in Section 7.3.6)
27	0	Reference-point shift (incremental encoders only) should be entered for numerical readjustment of the reference point
28	$0.2 \cdot v_{max}$	20% of the maximum speed is the recommended initial value
29	$0.1 \cdot v_{max}$	10% of the maximum speed is the recommended initial value
30/31	0/0	Backlash compensation inactive
38	1 000 [MSR/min/MSR]	Generally applicable position control loop gain
39	0	Following-error monitoring inactive
40/41	1 000/1 000[10^3MSR/s^2]	Very low acceleration values
42	0	Jolt filter switched off

Table 7-3 Initial contents of machine data, continued

MD (E)	Value	Explanation
43[1)]	U_{max} = 1,000...10,000 (mV)	Specified maximum amount of drive setpoint
44	0	Offset value for drive setpoint
45	0	Voltage ramp inactive

1) This pair of values corresponds to the speed category of the drive. It serves as a basis for calculating the K_V factor in the servo, and must therefore be entered correctly.
Recommendation: So far as possible, U_{max} should be set in the range between 8 and 9 V.

7.3 Testing and optimization

Testing and optimization information

Once you have installed, wired and parameterized the unit, you can test and optimize your FM 354 positioning module. Testing and optimization can be performed with the aid of the testing and start–up interface with or without the user program.

You can also test individual modes and their traversing programs, and view and debug them during execution.

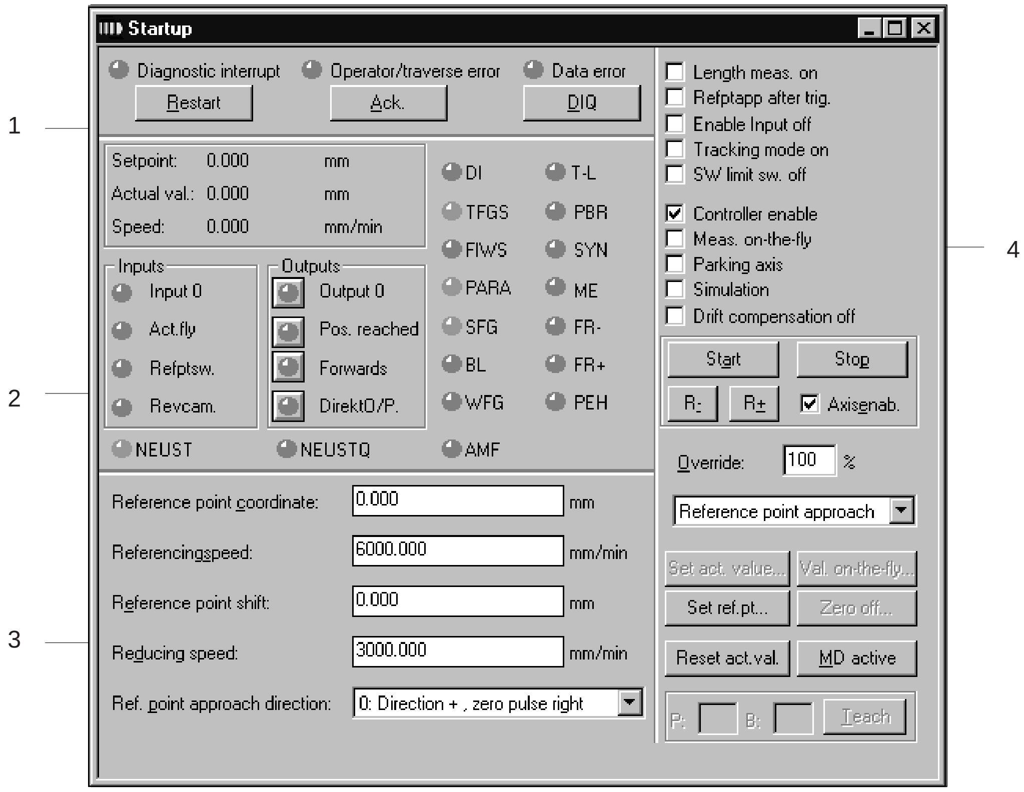
There are two ways of operating the FM:

- **CPU is in “STOP”, test without user program**
- **CPU is in “RUN”, test with user program**

  You can monitor the interface between the FM and the user program. You can also control the program from the start–up user interface when control signal [TFB] (TEST_EN) is enabled in the user program. Example application 3 (see Section 6.7) can be included in the user program for this purpose.

This interface is installed with “Parameterize FM 354”. Once the FM 354 has been parameterized, you can call it up by selecting the menu **Test ▸ Startup** or by selecting from the overview display.

When you call up this menu the following screen appears:



1 – Error field
2 – Status field (e.g. actual values, check-back signals)
3 – Field for mode-specific inputs
4 – Field for input of values/settings/commands and start/stop for movement

The abbreviations for the checkback signals are described in Section 6.2.2.

Fig. 7-2 Startup interface (e.g. for “Reference-point approach” mode)

Note

To start a movement, we recommend the following input sequence:

- Select a mode
- Turn simulation on (if you want an operating case)
- Servo enable
- Enable axis
- Override 1...100%

You can operate the “R+” and “R–” buttons in the “jogging” mode as follows:

1. Select “R+” or “R–” with the mouse
2. Press the space bar

You can operate “Start” and “Stop” with the mouse, or with the space bar if you have already selected the button.

The digital outputs are not set in the “Stop” status of the CPU.

When you operate the following buttons, you will get dialog windows:

- Set actual value...
- set actual value on-the-fly
- set reference point
- zero offset

Warning

If you move the axis directly (without simulation), for safety’s sake make sure you can switch off the hardware if a hazard arises.

Note

If you use the start–up user interface to operate the FM 354 when the CPU is in “STOP”, and then switch the CPU to “RUN” and then immediately switch to the start–up interface in your user program by means of the [TFB] (TEST_EN) signals (e.g. if example application 3 is included in the user program), please note the following:

You must select the mode again from the start–up interface, or close the start–up interface and call it up again.

You can also call up the following screens:

The following display appears when you select **Test ▸ Alarms** :

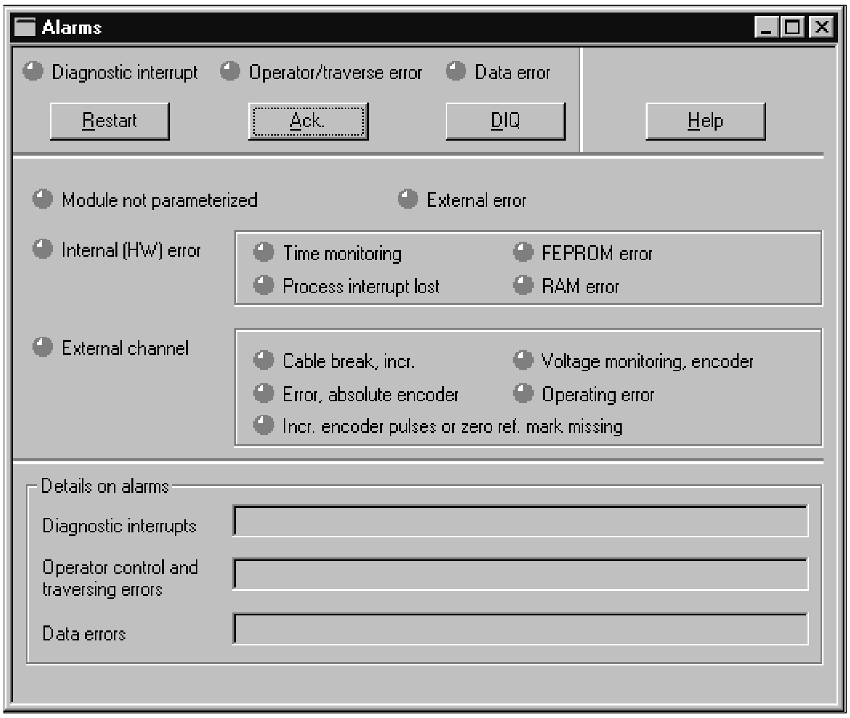


Fig. 7-3 Troubleshooting

The following display appears when you select **Test ▸ Service data** :

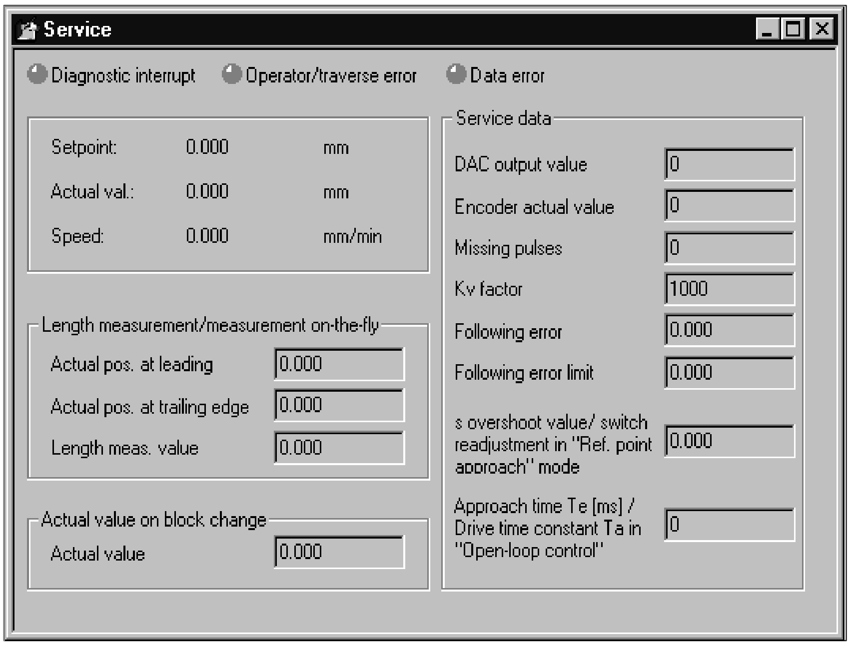


Fig. 7-4 Service data

Checklist

When starting up the machine axis, it is important to perform the following steps in the indicated sequence. Steps 1 to 5 are always necessary; the rest are optional, depending on your own application.

Table 7-4 Checklist - Startup of machine axis

Step	Check	What to do:	Page	OK
1	Activation of machine data	See Section 7.3.1	7-11	
2	Monitoring of drive and encoder switching	See Section 7.3.2	7-12	
3	Basic startup of position controller	See Section 7.3.3	7-16	
4	Optimization of position controller	See Section 7.3.4	7-18	
5	Readjustment of reference-point coordinates	See Section 7.3.5	7-22	
6	Activation of position controller diagnostics	See Section 7.3.6	7-23	
7	Activation of software limit switches, drift compensation and backlash compensation	See Section 7.3.7	7-25	

Note

In order for an axis to start, the start enable checkback signal must have been set.

If there is no start enable, this may be because:

- “Axis enable” is not set
- “Stop” is set
- “Operation in progress” is active

7.3.1 Activating the machine data

Overview

The checkback signal PARA notifies you that a DB-MD has been retained. This machine data is automatically activated at power-up. The module's positioning functions are ready to operate.

If no DB-MD is present as yet on the FM 354 when the control is switched on, the module can only communicate by way of the MPI interface. The control signals are not processed by the FM 354. Once an error-free DB-MD has been transferred, the machine data is automatically activated, PARA is set and the control signals are processed.

If the FM 354 is working with activated machine data, you can transfer a new data block or individual parameters in modified form to the module, and if the entire DB-MD is error-free this new data can then be put into effect by way of the "Activate machine data" function.

The following approaches are possible:

- If only E data have been modified in the machine data record since the last activation, the equipment is activated with module status "Operation in progress" = 0, without interrupting the servo cycle. "SYN" is retained.
- If K data has also been modified in the machine data record since the last activation, activation takes place with module status "Operation in progress" = 0 by way of a cold restart of the servo, just as occurs for a power-up of the module. The instantaneous actual position is still displayed, but encoder pulses from incremental encoders might go unrecorded. "SYN" is reset.
- If the machine data record contains erroneous data at activation time, the function is rejected, with the "Machine data cannot be activated" error message (see Table 11-5, Class 2, No. 21).

7.3.2 Checking the drive and encoder circuits

Overview

The following startup action allows you to monitor the proper actuation of the drive unit and the encoder:

Note

Always be sure to put MD modifications into effect with “Activate machine data.”

Caution

Before triggering any traversing movement, be sure to check that there is enough space for the axis to move in the desired direction.

Drive actuation

You can use the following flow chart to check the actuation of the drive.

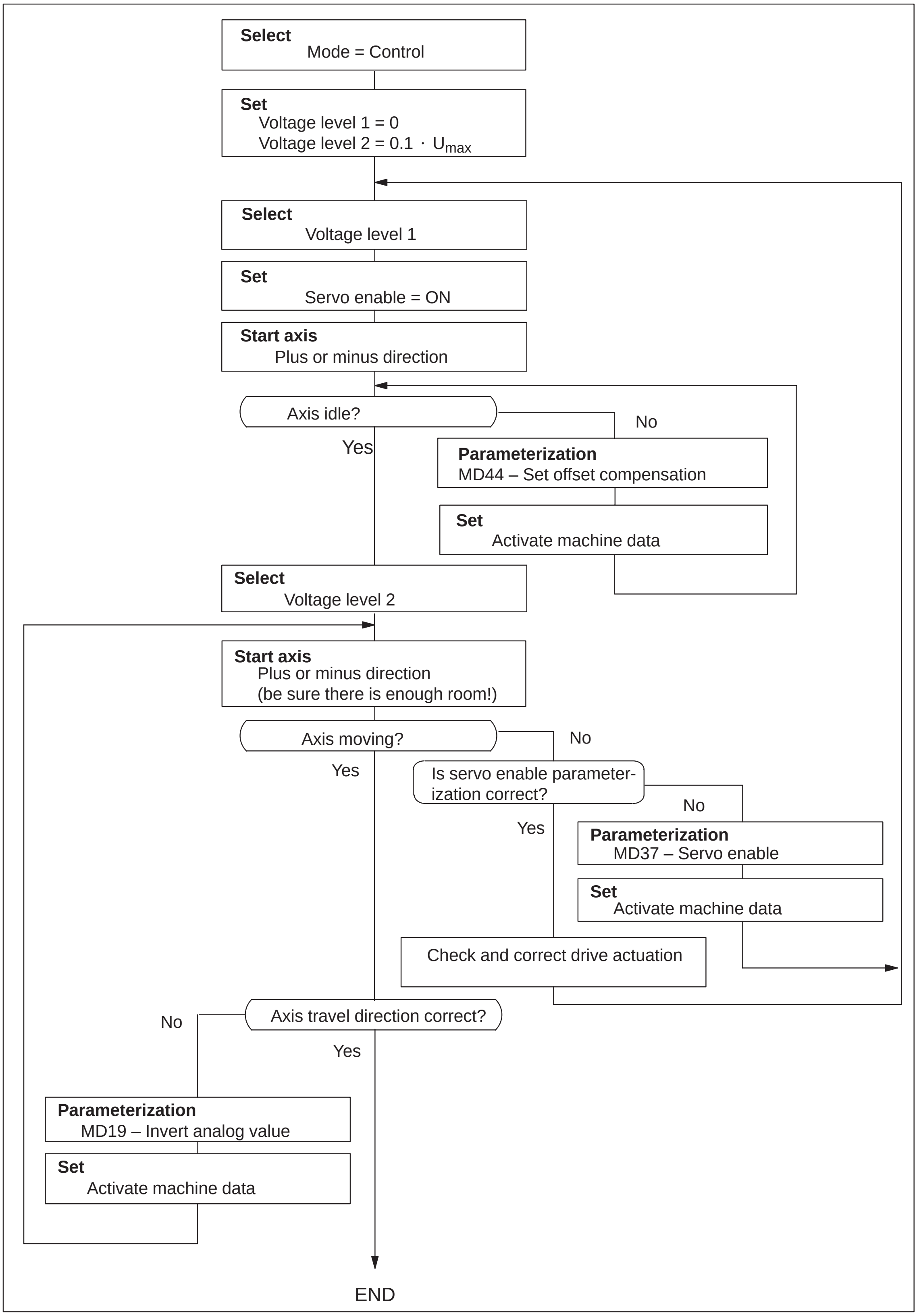


Fig. 7-5 Drive actuation

Encoder actuation and traversing speed

You can use the following flow chart to check the encoder actuation and traversing speed.

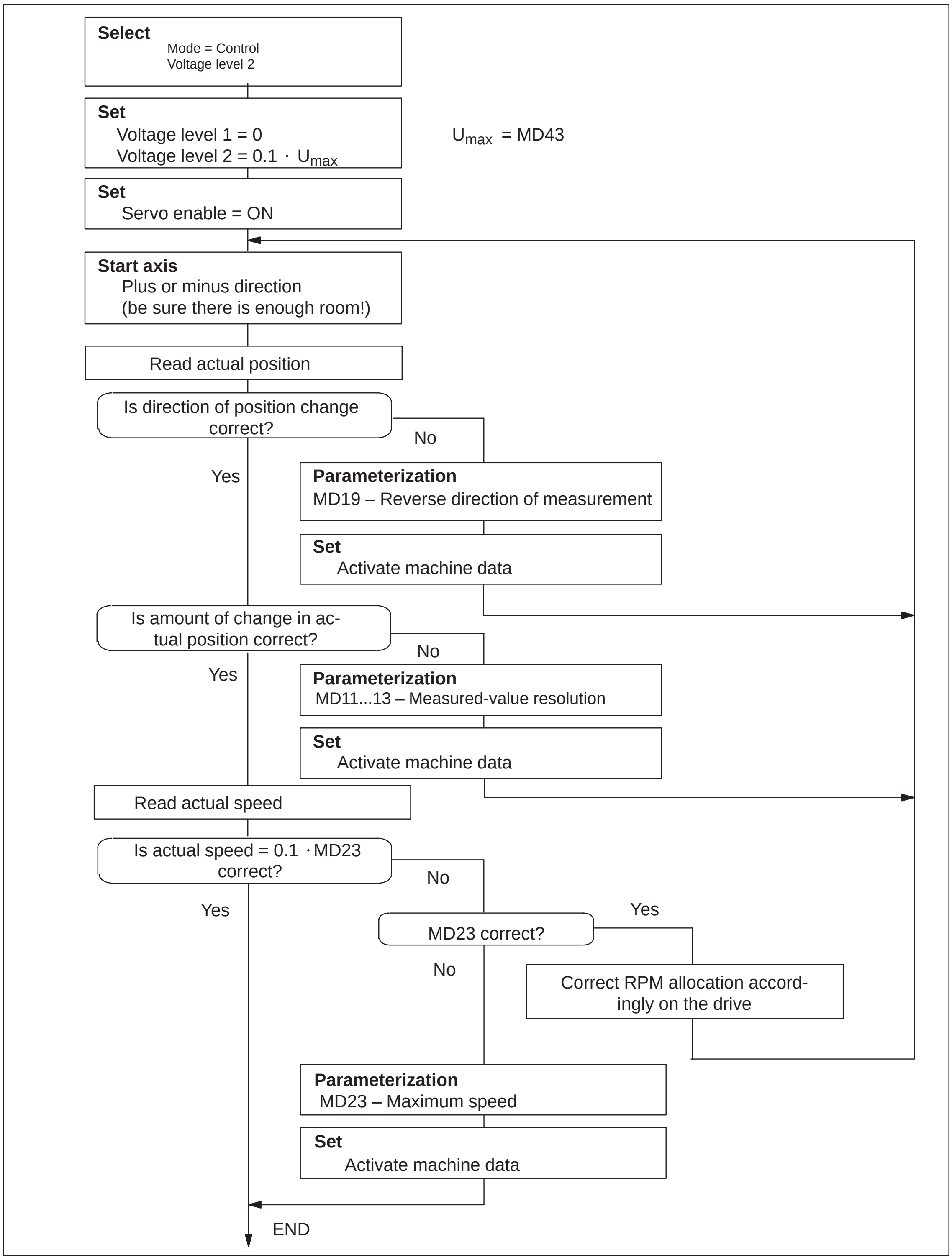


Fig. 7-6 Encoder actuation and traversing speed

Drive transition time and maximum voltage rise

For the following position-controller optimization it is important to know the drive time constant (transition time). In open–loop control mode and on errors with the response "Everything Off" (see Section 11) the voltage value is fed to the drive by way of a ramp defined in MD45. A variety of drives, as well as certain mechanical or technological situations, may require a limitation on the voltage rise. If you do not have a specific value in hand and wish to find a suitable rise value by trial and error, please use the following procedure:

Note

A voltage rise setting will obviously make the axis stop more slowly if an "Everything Off" error response occurs.

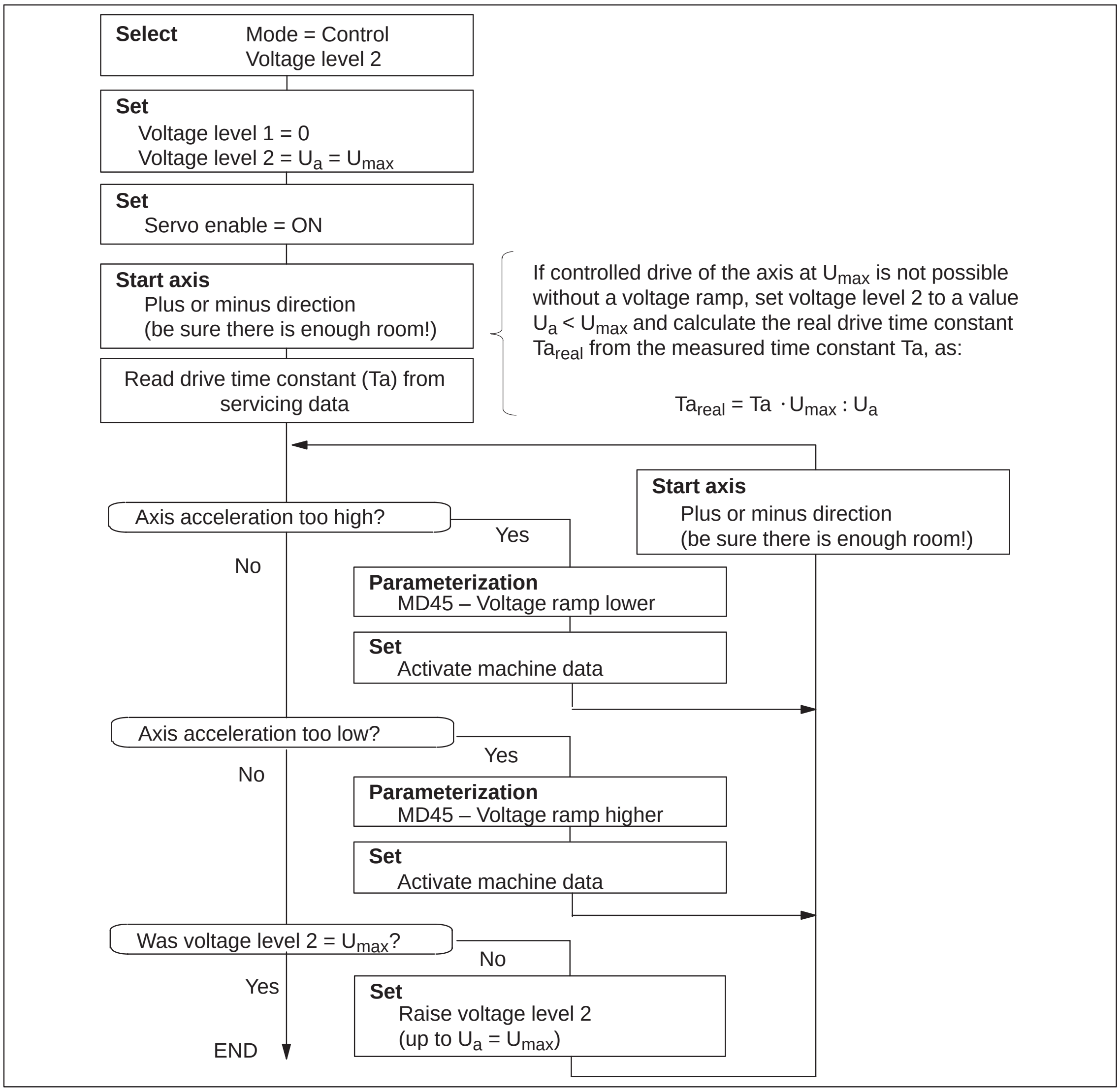


Fig. 7-7 Drive transition time and maximum voltage rise

The axis is now ready for startup of the position controller.

7.3.3 Basic startup of the position controller

Overview

The following startup action allows you to monitor the basic function of the servo - i.e. position control of the axis:

Note

Always be sure to put MD modifications into effect with “Activate machine data.”

Non-release control

You can use the following flow chart to check the non-release control.

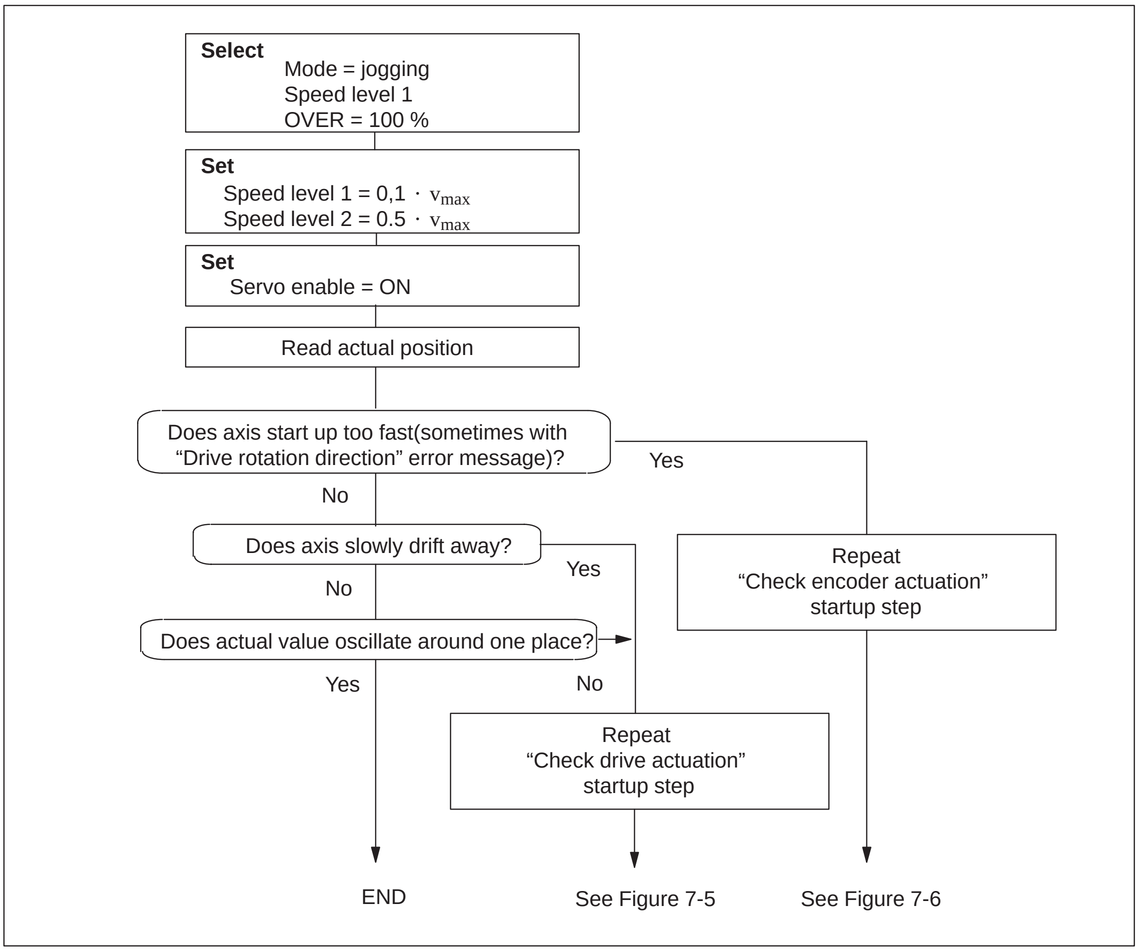


Fig. 7-8 Non-release control

Positioning

Use the following flow chart to check axis travel to a target position.

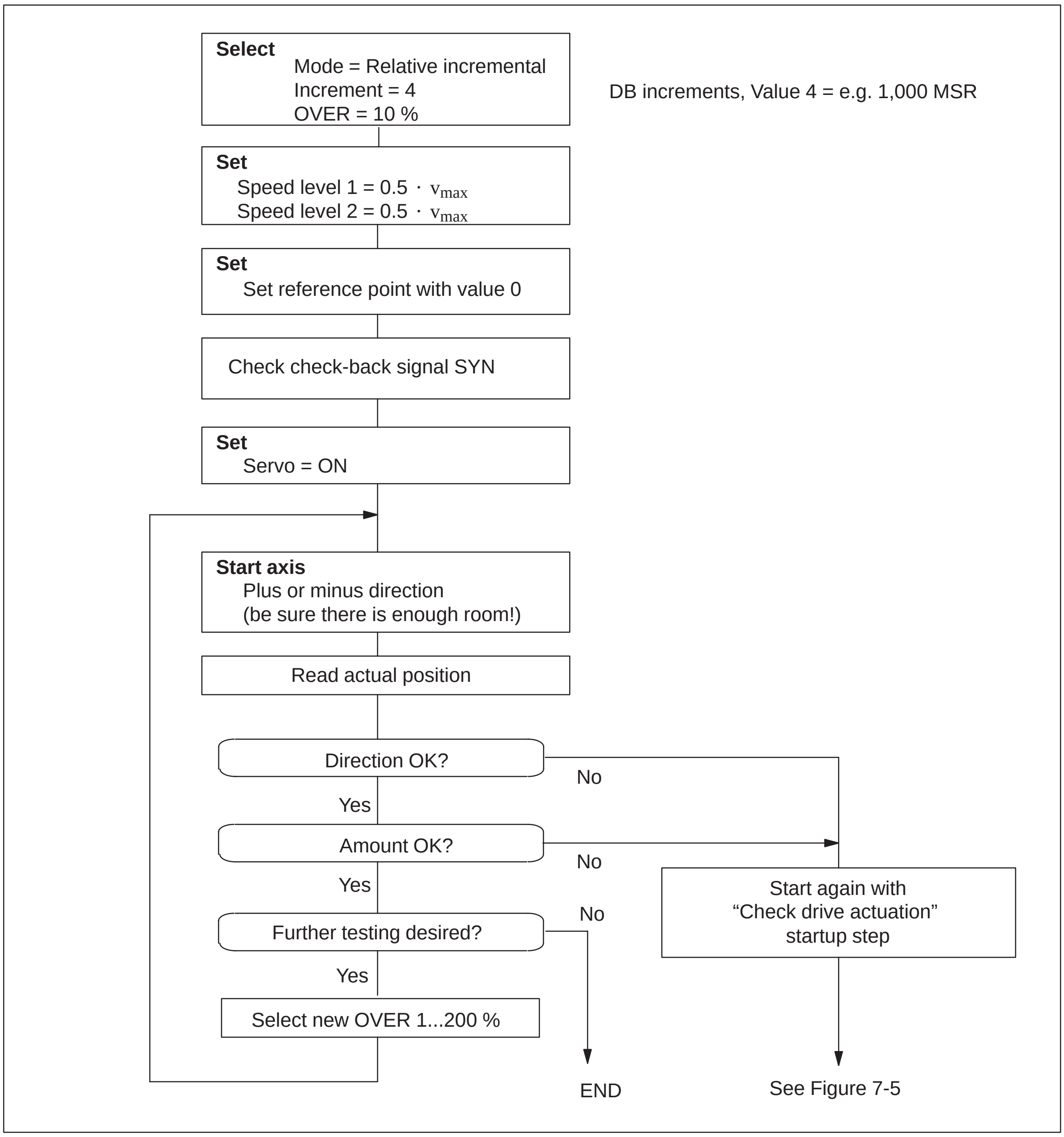


Fig. 7-9 Positioning

7.3.4 Optimizing the position controller

Overview

In principle, the dynamic response of an axis is essentially determined by the dynamic response of the variable-speed drive; there is not sufficient space to discuss this topic here. But this latter dynamic response, in turn, is influenced by the design characteristics of the machinery, such as friction, backlash, torsion and the like. By feeding back the measured displacement, a position controller generally closes the outermost loop of a controller cascade with the following structure:

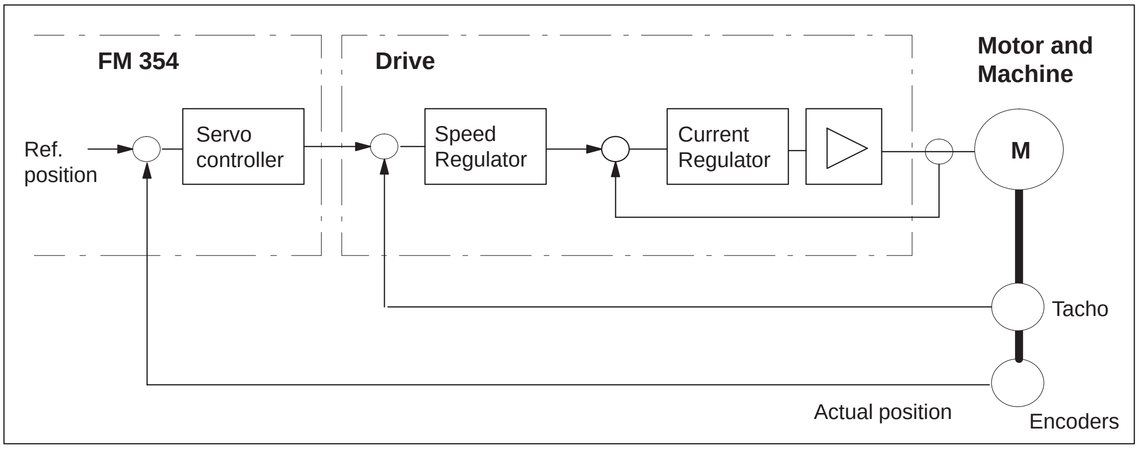


Fig. 7-10 Position control circuit

Procedure

The following instructions are intended as an aid for practical situations.

Position controllers must meet a variety of requirements for various technological applications.

Assessment criteria for the quality of the positioning process can include:

- Good uniformity of traversing movement
- Little or no overshoot at the target point for positioning
- Short positioning time
- A continuous acceleration (soft travel).

In most applications, several of these criteria will be important, so that most of the time the dynamic response of the controller can be optimized only with a number of compromises.

Execute test movements as in Figure 7-11 during the optimization steps described below.

To trigger test movements

You can trigger test movements as follows as you perform optimization:

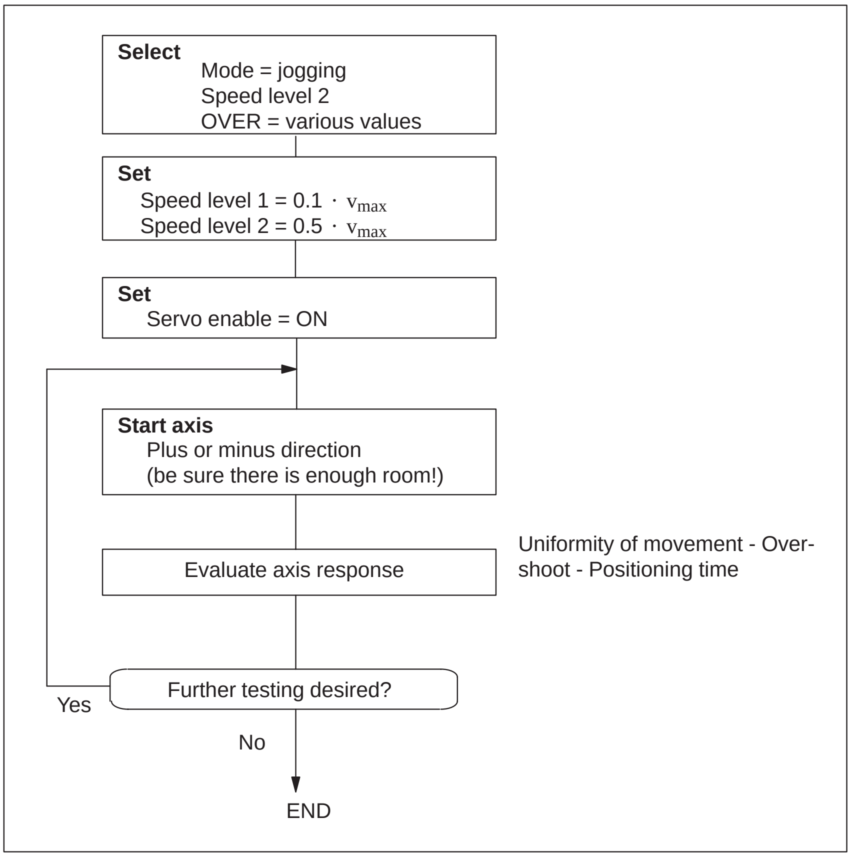


Fig. 7-11 Test movements for optimizing the servo control system

Selecting initial values of response-defining MD

Set the following machine data in accordance with the drive time constant #Ta (Ta_{real}) determined in Section 7.3.2 to the initial values for the optimization steps below, e.g. for an axis in MSR 10^{-3} mm:

- Acceleration, delay

  MD40 = MD41 [mm/s^2] = 30 · MD23 [mm/min] : Ta [ms]

- Jolt time

  MD42 (ms) = 0

- Positioning loop amplification

  MD38 (1/min) = 100,000 : Ta (ms)

The acceleration value that actually acts on the system is reduced by the time response of the position control circuit - i.e. as a function of the K_v value. The maximum acceleration (a) in this setting can be attuned to the drive time constant, and can be estimated as follows:

a_{max} [mm/s^2] = 16 · MD23 [mm/min] : Ta [ms]

The qualitative effect of the parameters on the positioning process appears in the following table:

Table 7-5 Effect of machine data that defines response

	MD38	MD40/41	MD42
Quiet running	small	–	–
Noise immunity	great	–	–
Soft movement reversal	small	great	great
Positioning without overshooting	small	great	great
Fast positioning	great	small	small

Optimization of dynamic response

The following startup actions allow you to optimize the position controller to your requirements. Check all speed ranges, and if applicable give the greatest weight in evaluating the results to the speed that is the most significant for your technology.

Optimization for uniformity of movement

You can make optimization of the position controller considerably easier by analyzing the actuating signal or drive speed (tachometer voltage) with a storage oscillograph. The resulting oscillograms for the transition functions U(t) and v(t), i.e. the oscillation pattern, can be interpreted more easily (see Figure 7-12).

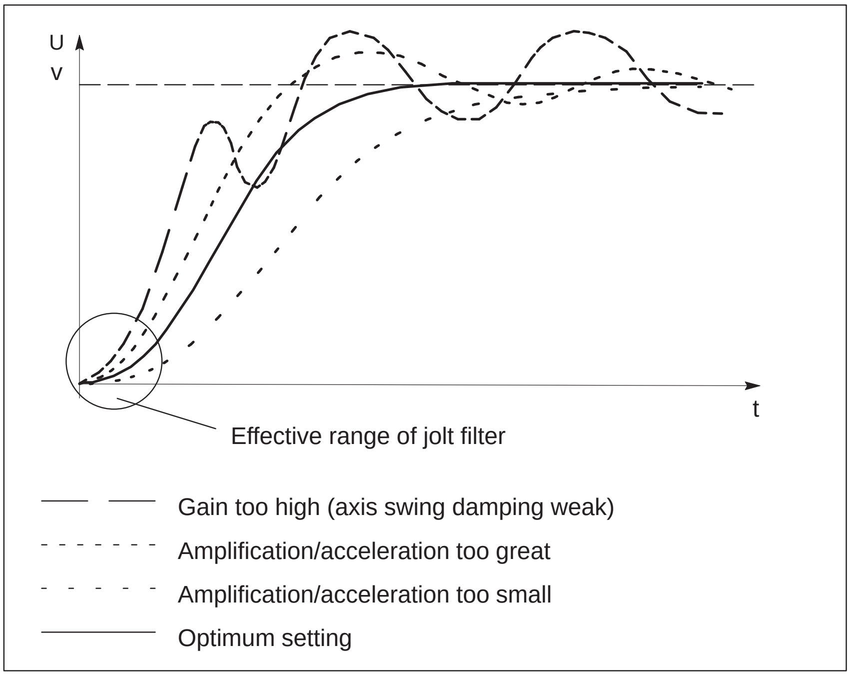


Fig. 7-12 Transition function of the position-control circuit

Optimization for overshoot

Evaluate the overshoot in the target position (s-overshoot in the servicing data).

For suitable machine data changes, see Table 7-5.

Optimization for positioning time

Evaluate the approach time to the target position (approach time Te in the servicing data).

For suitable machine data changes, see Table 7-5.

Optimization for especially soft travel (super-soft)

For particular applications, especially soft travel response of the axis is desirable. By choosing the following output values for the machine data affecting the dynamic response you can produce a very soft movement where the acceleration is controlled exclusively by the jolt filter. The effective maximum acceleration in movement-reversal processes responds proportionally to the difference in speed, and reaches its maximum in the transition from v = 0 to maximum speed (see Figure 7-13).

- Acceleration, delay

  MD40 = MD41 (mm/s^2) = 0

- Jolt time

  MD42 (ms) = 0.5 $\cdot$ Ta (ms)

- Positioning loop amplification

  MD38 (1/min) = 100,000 : Ta (ms)

The maximum value of the actual effective acceleration can be estimated as follows:

a_{max} [mm/s^2] = 16 $\cdot$ MD23 [mm/min] : Ta [ms]

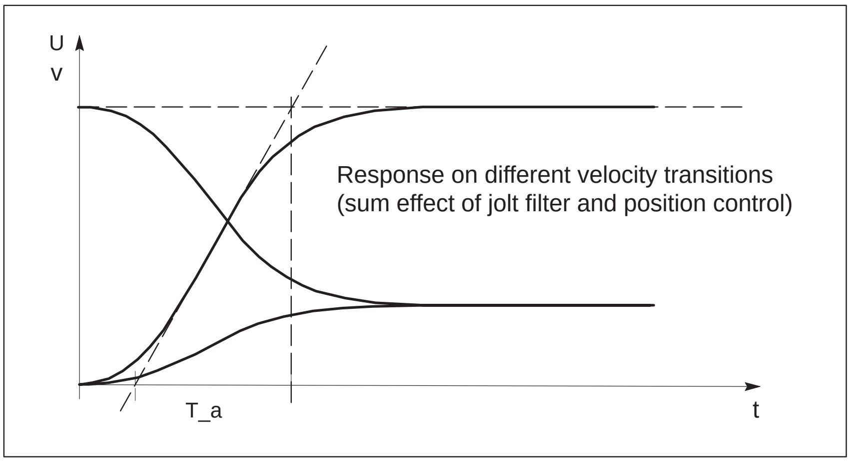


Fig. 7-13 Response on different velocity transitions (sum effect of jolt filter and position control)

Compromise optimization

When optimizing for several of the above criteria, you can determine the machine data from the results of the individual optimizations by a variety of methods:

- Guarantee of all partial results
  - Least determined value of MD38
  - Greatest value for each of MD40, MD41 and MD42
- Prioritization of one optimization criterion

  Set MD38 and MD40-MD42 to the values that match the highest-priority optimization criterion for your application, and again evaluate response as to the remaining criteria.
- Taking the mean of partial results

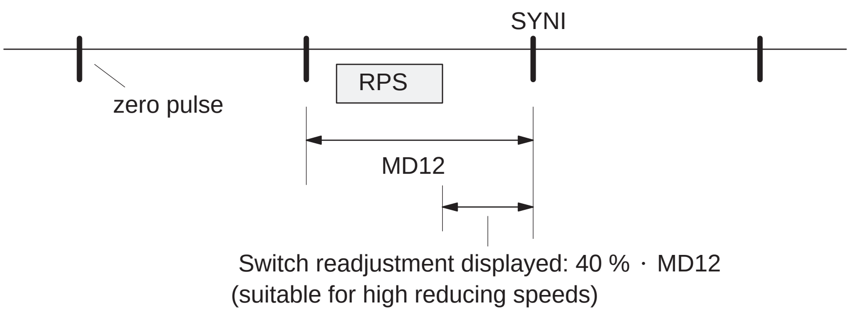
  Set MD38 and MD40-MD42 to the means of the individual partial results, and again evaluate response as to all criteria.

7.3.5 Realigning the reference point coordinates

Axis with incremental encoder

To ensure distinct reproducibility of reference recordings, it is necessary for the synchronizing zero pulse (SYNI) to be a distinct distance away from the reference point switch (RPS). At low reducing speeds, we recommend a distance of from 10% to 90% of the distance of one encoder revolution. For high reducing speeds we recommend 30% to 60%. Check this value in the servicing data report after executing a reference point approach (switch alignment value) and if you find nonconformity to the required value range, make a corresponding adjustment in the relative position allocation between the encoder and the reference point switch.

Example: Positive search direction



Set the referencing velocity to the highest value compatible with your requirements. It is important to be able to decelerate to the reducing velocity across the length of the reference-point switch. If this is not the case, an additional repositioning to the RPS occurs before the search phase of the synchronizing zero pulse begins. Compare the cycle of the executed traversing movements with Section 9.2.3 and optimize the referencing speed (MD28).

Then readjust the reference-point coordinates proper by entering the necessary reference-point shift in the machine data. After the machine data is activated, the new reference-point shift takes effect with the next search for reference.

Axis with absolute encoder (SSI)

In a suitable mode ("jogging", "incremental relative") move to a known point on the axis and execute the Set reference point function with the known position value. The set position and actual position will immediately be set to this value, and the allocation of an absolute value to the absolute encoder (SSI) will be entered in the machine data record (MD17). If you want to archive this value externally, apart from the module's own data memory, perform a readout of the machine data DB and save it to a floppy disk or to the hard disk of your PG.

7.3.6 Activating position controller diagnostics

Overview

Once the position controller has been optimized, activate the position controller diagnostics. If position control is performing improperly or the axis is responding abnormally, this function will trigger error messages.

You can use the following flow chart to start the position controller diagnostics:

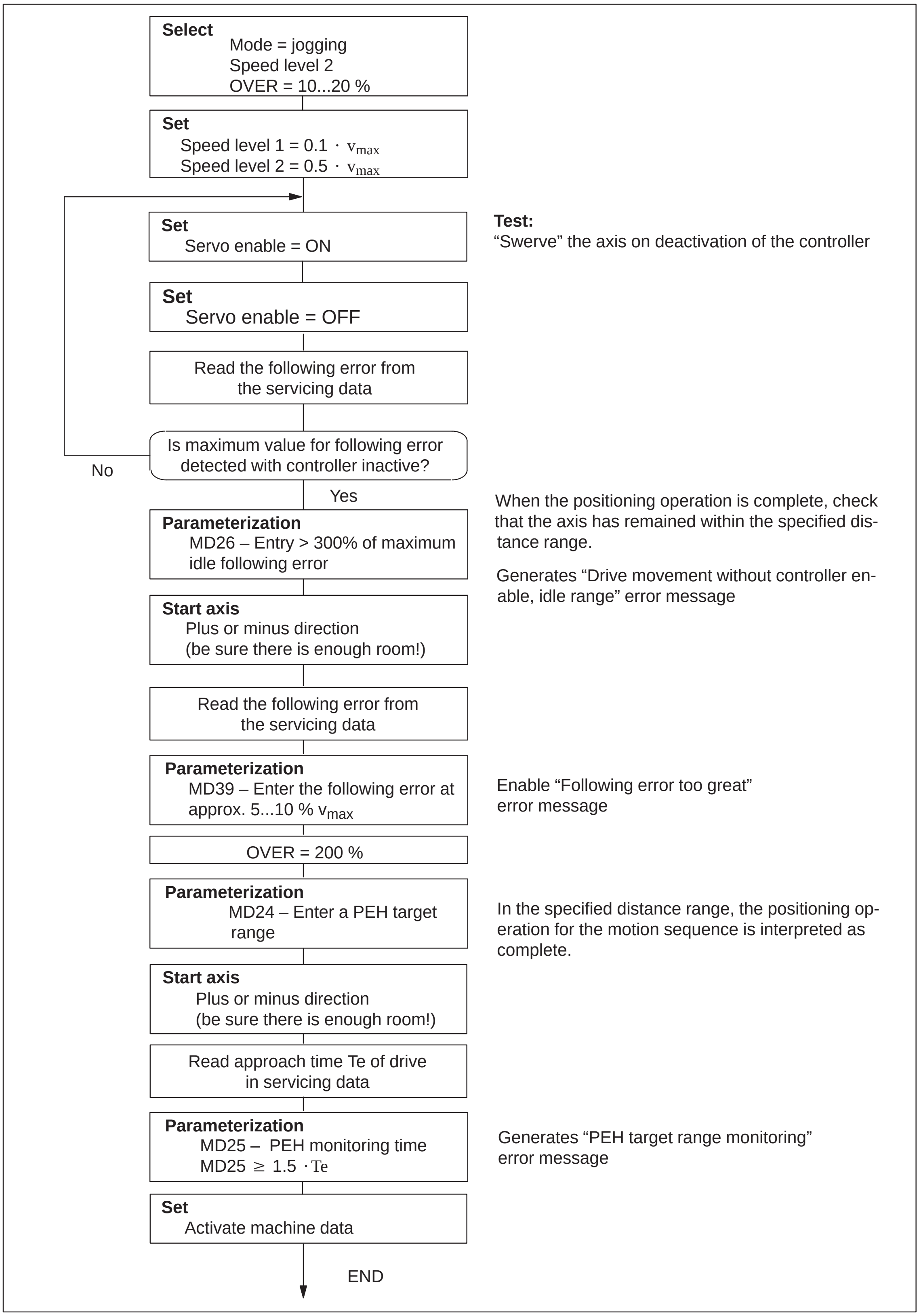


Fig. 7-14 Activation of position controller diagnostics

7.3.7 Activating the software limit switches, drift compensation and backlash compensation

Software limit switches

Carefully run the axis up to its extreme limits as specified for normal operation of the machine. Enter these position actual values into the machine data MD21/MD22 as software limit switches, and activate them.

Note

If you change the reference-point coordinate later or use Set reference point for the absolute encoder, you must redefine the positioning values of the software limit switches.

If you do not need the software limit switches, the input limits -10^9 and 10^9 [MSR] must be entered in MD21/MD22 (for default values, see Table 5-5).

Drift compensation

If you want to use the drift compensation function in addition to the offset compensation already described in Section 7.3.2, activate it in the machine data (please see the function description in Section 9.7, Position control).

Backlash compensation

With indirect position measurement (for example, with an encoder on the motor) the free play of mechanical transmission elements during positioning may cause a position deviation of a machine part (such as a lathe saddle) that is to be positioned but does not lie in the measured-value feedback loop. As a rule, a piece of the distance will be “missing” after a reversal of direction. This backlash amount can be determined as a mean at various axis positions, and entered in the machine data MD30 and MD31.

You can use the following flow chart to determine backlash and activate backlash compensation.

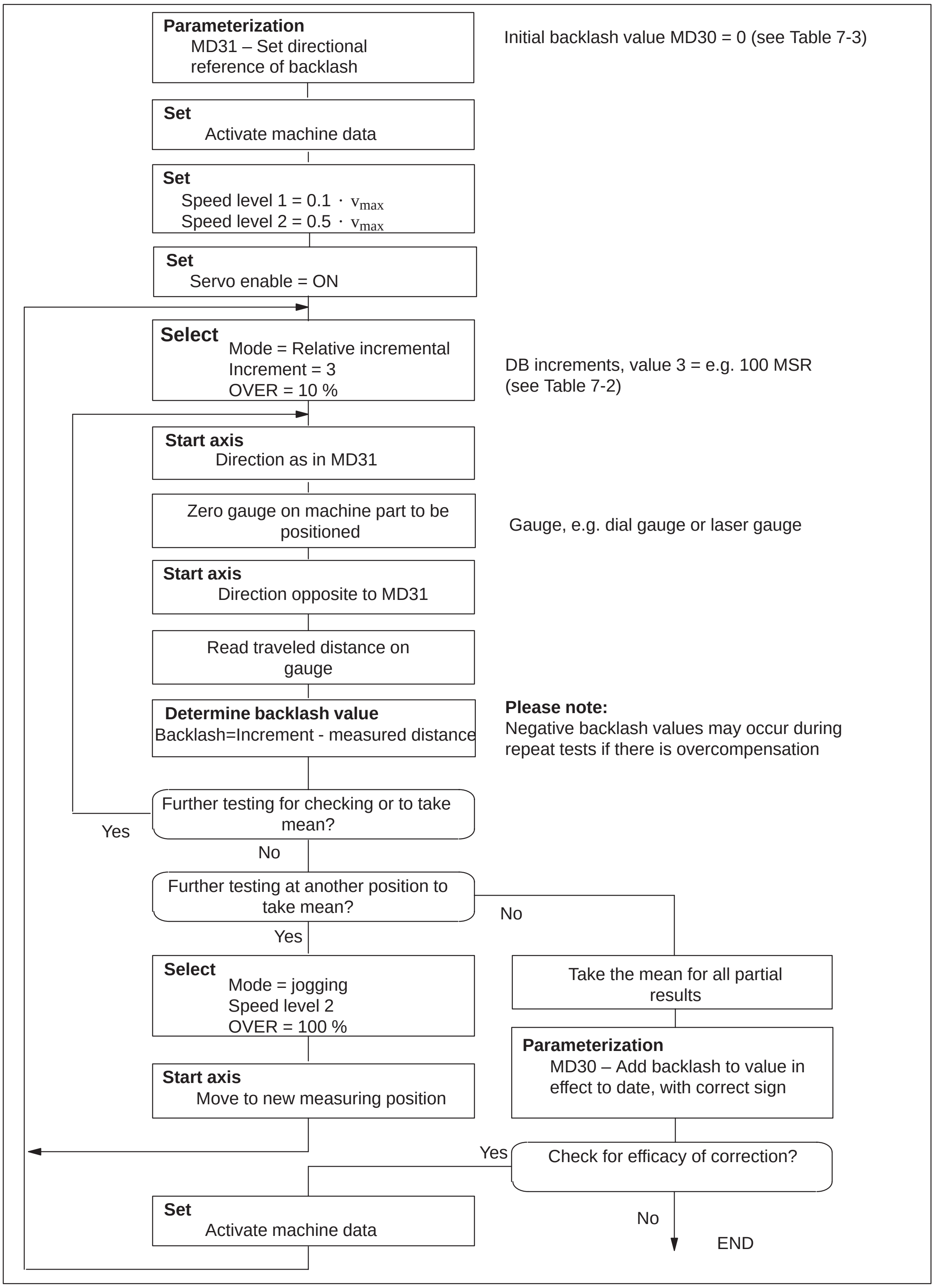


Fig. 7-15 Determination of backlash and activation of backlash compensation

8 Human-machine interface

Summary

In this chapter you'll find an overview of the operator control and monitoring capabilities offered by the FM 354.

For operator control and monitoring of the FM 354, an operator panel can be connected to the CPU via the MPI interface (see Figure 1-2).

The module uses the SIMATIC interface (backplane bus) to communicate with the control panel.

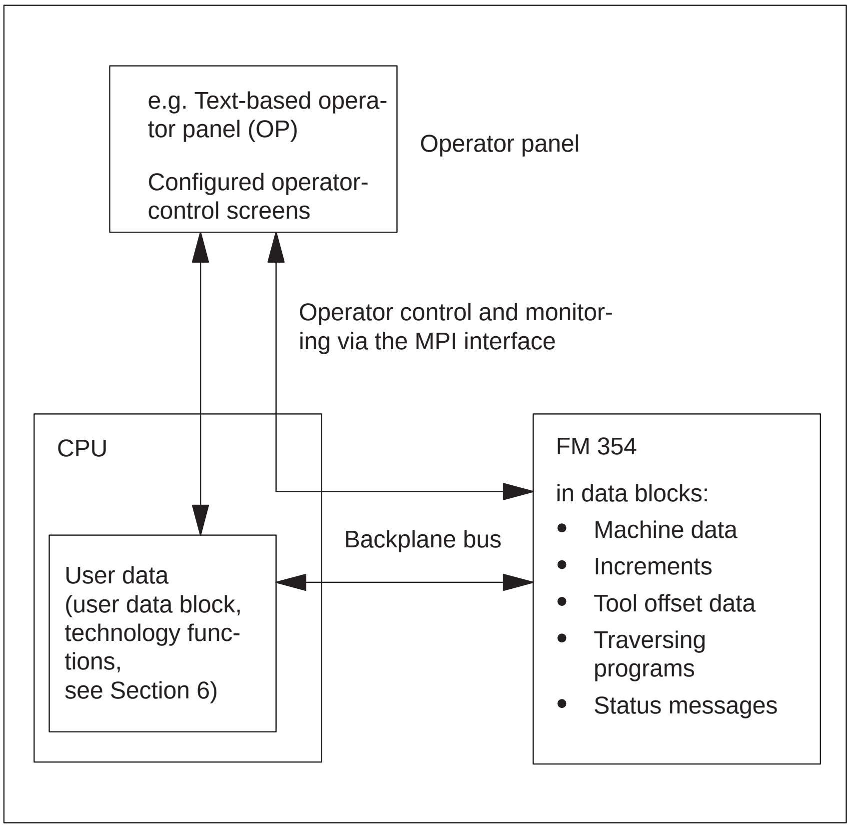


Fig. 8-1 Operator control and monitoring for the FM 354

Operator control and monitoring of FM data/signals on the CPU 314

The data and signals that can be controlled and monitored at the control panel are listed in the user data block. These data or signals must be processed by the user program (for data and signals see Chapter 6 and Section 8.1).

What can I control on the FM 354?

Using the keyboard of the operator panel, you can change the data/signals in the data blocks:

- Machine data DB No. 1200
- Increment sizes DB No. 1230
- Tool offset data DB No. 1220
- Traversing programs DB No. 1001...1199

What can I monitor on the FM 354?

The following data and signals can be displayed on the operator panel display:

- Machine data DB No. 1200
- Increment sizes DB No. 1230
- Tool offset data DB No. 1220
- Traversing programs DB No. 1001...1199
- Status messages DB No. 1000 (DB-SS) etc.
  - Operating data, such as actual values
  - Active NC blocks
  - Linear measurements
  - Actual value block change
  - Check-back signals and error conditions
  - Service data

The configuration package includes a pre-configured interface for the COROS OP 07 and OP 17 operator panels.

Chapter overview

8.1 Standard HMI (human–machine interface) for the OP 07 and the OP 17

Overview

This Section describes a preconfigured user interface, which you will need to change according to your project (e. g. FM addresses, DB no.), for the following COROS equipment (operator panels):

- OP 07
- OP 17

The tool to be used for this is the configuring tool “ProTool/Lite” V3.0. You can use it to modify, add or delete screens.

The user interface is addressed to user DB No.1 in the CPU (target system 1, address = 2) and to the DB-SS of the FM 354 (target system 2, address = 3).

The text field “FM user name” represented in the images can be renamed to a text of your choice.

You can print out the entire configuration using “ProTool/Lite” V3.0. This provides you with detailed screen descriptions.

You will find the preconfigured user interface in the following directory:

SIEMENS\STEP7\EXAMPLES\S7OP_BSP

DB-SS

This data block for status messages (DB 1000) contains the control/checkback signals, as well as the system data of the FM 354. The data of the DB-SS can only be read.

Monitoring

The data for monitoring can be read and displayed directly in the BD-SS as well as in the corresponding parameterized DBs of the FM 354.

Operator control

For operator control, the data and signals (including memory bits and values) are written to the user DB of the user program.

User program

Your user program must analyze the signals (only those which are relevant to its applications). User-specific interlocks can be incorporated and the data/signals are to be transmitted to the FM 354 by way of the FCs.

8.1.1 Standard user interface for the OP 07

User interface of the OP 07

The following illustration provides you with an overview of the OP 07 user interface (menu tree).

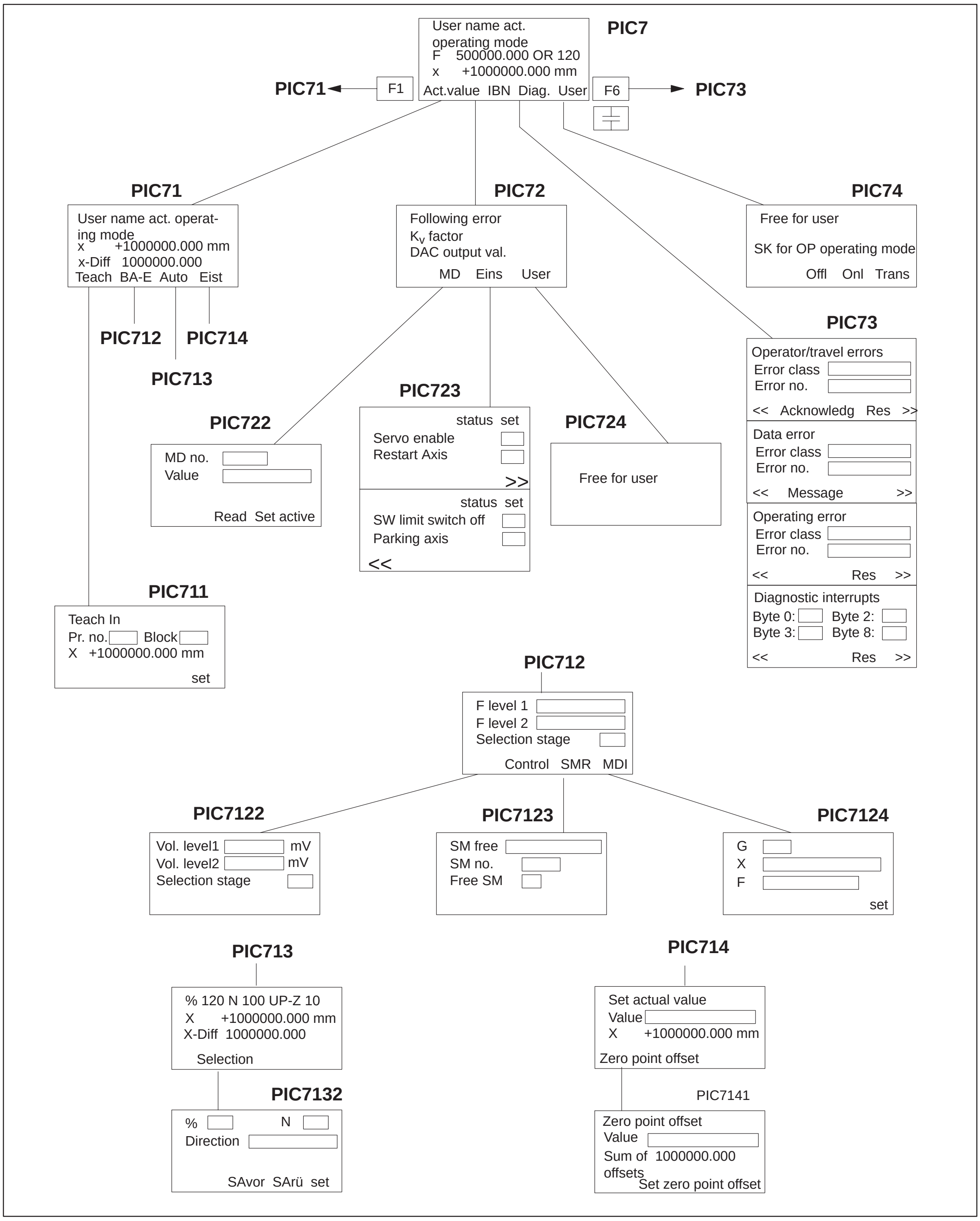


Fig. 8-2 Menu tree of the OP 07 user interface

Figure 8-2 describes the functions of the global function keys for the user interface of the OP 07.

ESC key You can use this key to call up the previous screen of the higher levels.

>> << **Soft keys** You can use these keys to call up the next inset screen within the same screen (same screen number).

F1 **Function key** You can use this key to skip from any point in the menu tree to the actual value display main screen (PIC71).

F6 **Function key** You can use this key to skip from any point in the menu tree to the diagnostic main screen (PIC73).

Note

The screens of the user interface (see Figure 8-2 and description of each screen, Table 8-1) contain display fields and input/output fields. These fields contain values of configured variables.

- The display fields are addressed to the DB-SS (“Control 2,” DB1000) and are read directly from the FM 354.
- The input/output fields are addressed to the user DB No.:1 (Control 1-CPU).
  - Transmission of these values occurs from the OP 07 to the CPU into the user DB. These values (if needed) must be transmitted to the FM 354 by the user program.
  - If certain values or control signals can be written only under the right conditions (e.g. if axis is in HOLD mode or selection of a certain operating mode is required), then the user program must ensure, by analyzing the response signals, that these conditions are met.

Table 8-1 below describes the individual screens of the user interface.

Table 8-1 Description of the screens in the user interface

Screen name	Screen No.	Description
Main screen	PIC7	This screen is displayed to you after the OP 07 is enabled. The FM354 values are display values. You can use the soft keys to call up the following main screens: • Softkey “Istw” → PIC71 • Softkey “IBN” → PIC72 • Softkey “Diag” → PIC73 • Softkey “Anw” → PIC74
Main screen, actual value display	PIC71	The values on the screen are display values. You can use the soft keys to call up the following screens: • Softkey “Teach” → PIC711 • Softkey “BA-E” → PIC712 • Softkey “Auto” → PIC713 • Softkey “Eist” → PIC714
Main screen, startup	PIC72	The FM 354 values (service data) are display values. You can use the soft keys to call up the following screens: • Softkey “MD” → PIC722 • Softkey “Eins” → PIC723 • Softkey “Anw” → PIC724
Main screen, diagnostics	PIC73	This screen shows you the diagnostic interrupts and error messages for the FM 354 in four lower-level screens. The following are displayed: • Error class and error number for operator control and guidance errors • The bit numbers of the single diagnostic interrupt bytes You can use the soft keys “Quit” and “Res” to acknowledge the errors. See Section 11.3 for diagnostic interrupts and error messages
User screens	PIC74 and PIC724	You can use these screens to configure your own contents.
Setting data for operating mode	PIC712	This screen contains input/output fields for the velocity levels or a text field for selecting velocity level (selections 1 or 2 possible). You can use the soft keys to call up the following screens: • Softkey “Steu” → PIC7122 • Softkey “SMR” → PIC7123 • Softkey “MDI”→ PIC7124
Setting data for the “Automatic” operating mode	PIC713	This screen contains only display fields. You can use the “Anwahl” soft key to call up screen PIC 7132.
Set actual value	PIC714	The actual value display is a display field. The value for set an actual value appears in an input/output field. You can use the soft key “NPVer” to call up the screen PIC7141. You can use the soft keys “Iwrü” and “IWset” to execute functions.

Table 8-1 Description of the screens in the user interface, continued

Screen name	Screen No.	Description
Machine data	PIC722	This screen contains input/output fields. Input of values is password protected. The values entered are in the user DB. You can use the soft keys to set bits in the user DB: • Soft key “lesen” – You can use this soft key to insert a memory bit (in the user DB) which causes the user program to read the item of machine data whose number has been entered. • Soft key “set” – You can use this soft key to set a bit which causes the user program transfer to the FM 354 the number entered under value for the item of machine data under MD No. • Soft key “aktiv” – You can use this soft key (activate MD) to set a memory bit (“activate MD” bit in the user DB) which is transmitted from the user program to the FM 354. Each bit in the user DB which you set in this screen must be reset by the user program following execution of the function.
Settings for startup	PIC723	The fields of this screen which are identified by an “x” are display fields. If the bit is set, you will see an x. If the bit is not set, the field will be blank. The other fields are text entry fields in which you can toggle between “x” and “ ”. You can use the soft keys “<<” and “>>” to change between the inset screens of this screen.
Set data for the “Control” operating mode	PIC7122	This screen contains input/output fields for the control levels or a text field for control level selection (selections 1 or 2 possible).
Set data for the “incremental mode, relative” operating mode	PIC7123	This screen contains input/output fields. The fields “SM-No.” and “free SM” are addressed to the same address in the user DB. You can use the SM No. to enter the values 1 to 100. The “free SM” field is a text field and you can toggle back and forth between “x” and “”. If you select “x”, this corresponds to the value 254 for free SM. The value of the free SM is found in the user DB.
Set data for the “MDI” operating mode	PIC7124	This screen contains input/output fields. The MDI data record must be preassigned by the user program with the corresponding bits (G, X, F bits are set) and possibly, with values for G, X and F. The entry field behind G is a text field. There, you can select between the values 90 and 91. Using the soft key “set” you can set a memory bit (in the user DB), which must be analyzed by the user program. With modification of this bit, the MDI block must be transmitted from the user DB to the FM 354, and the bit must be reset.
Program selection	PIC132	This screen contains input/output fields. This field for direction is a text field. It is upward and downward selectable. You can use the soft keys to set bits in the user DB: • Soft key “SAvor” - You can also use this soft key (automatic block search forward) or “SArü” (automatic block search backward) to set the corresponding bits in the user DB. • Soft key “set” - you can use this soft key to set a memory bit (in the user DB).

Table 8-1 Description of the screens in the user interface, continued

Screen name	Screen No.	Description
Teach In	PIC711	This screen contains input/output fields. You can use the soft key “set” to set a bit in the user DB. The actual value display (X) is a display field.
Zero offset	PIC7141	This screen features an input/output field. The sum of the offsets (Sumn Ver) is a display field. You can use the soft key “NPVset” to execute a function.

8.1.2 Standard user interface for the OP 17

User interface of the OP 17

The following illustration provides you with an overview of the OP 17 user interface (menu tree).

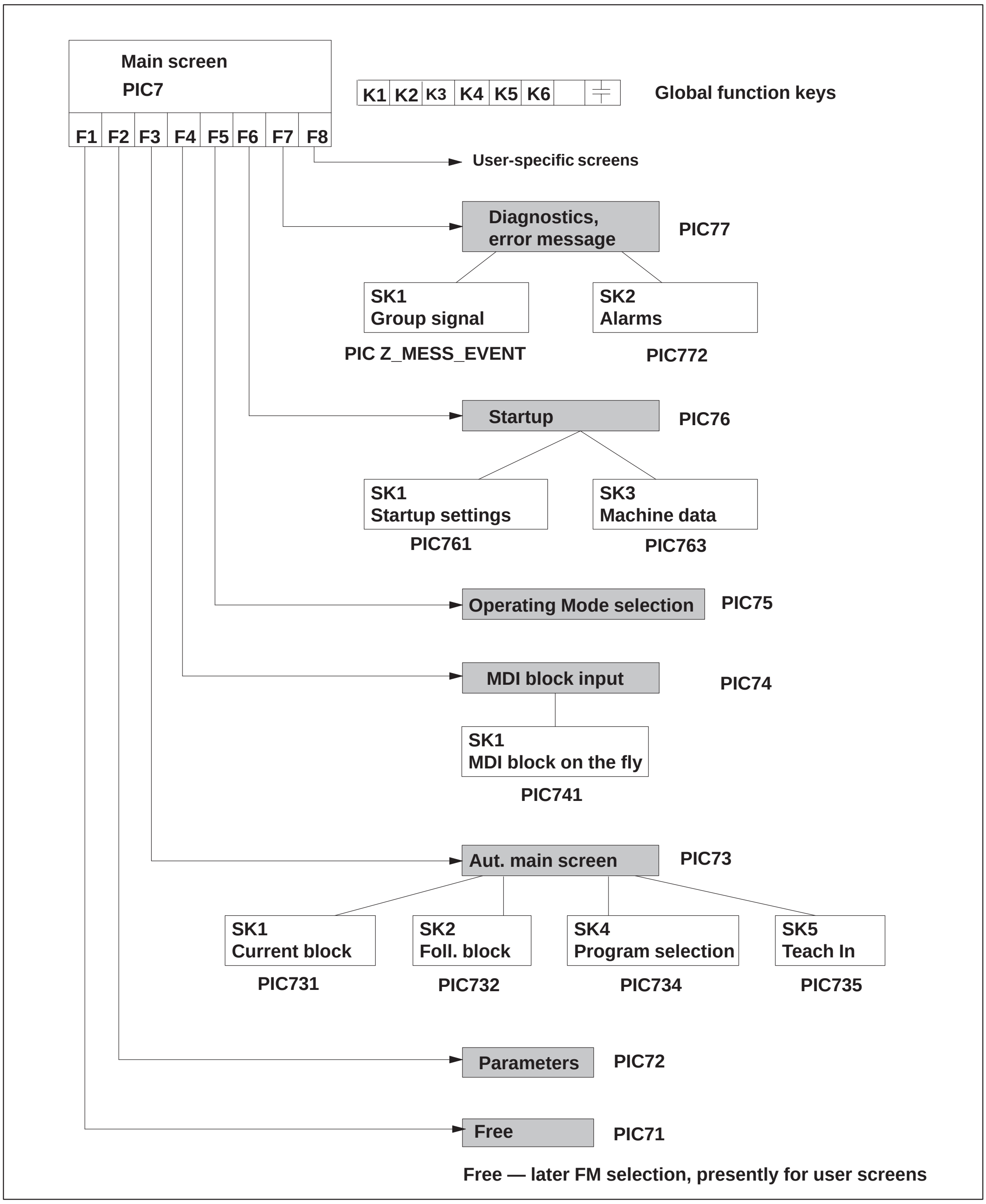


Fig. 8-3 Menu tree of the OP 17 user interface

Figure 8-3 describes the functions of the global function keys for the user interface of the OP 17.

	ESC key	You can use this key to call up the previous screen of the higher level (the table of contents in the main screen).
K1	**Function key**	You can use this key to jump from any point on the menu tree to the main screen (PIC7).
K2	**Function key**	You can use this key to jump from any point on the menu tree to the diagnostics, error message screen (PIC77).
K3	**Function key**	You can use this key to jump from any point on the menu tree to the operating mode selection screen (PIC75).
K4	**Function key**	OP 17 “Offline” operating mode selection
K5	**Function key**	OP17 “Online” (normal) operating mode selection
K6	**Function key**	OP 17 “Transfer” operating mode selection
F1 ... F8		F1 to F8 (local soft keys)

Note

The screens of the user interface (see Figure 8-3 and description of the individual screens) contain display fields and input/output fields. These fields contain values of configured variables.

- The display fields are addressed to the DB-SS (“Control 2,” DB1000) and are read directly from the FM 354.
- The input/output fields are addressed to user DB No.: 1 (control 1-CPU).
  - Transmission of these values occurs from the OP 17 to the CPU into the user DB. These values (if needed) must be transmitted to the FM 354 by the user program.
  - If certain values or control signals can be written only under the right conditions (e.g. if axis is in HOLD mode or selection of a certain operating mode is required), then the user program must ensure, by analyzing the response signals, that these conditions are met.

The pending errors are displayed in the errors line. More detailed error information is provided on the screens “Diagnostics, Troubleshooting” and “Interrupt messages.”

Description of the individual screens

The following illustration shows the screen layout of the preconfigured interface.

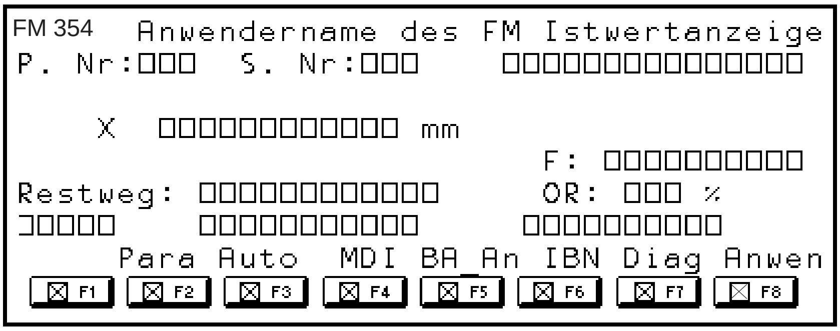


Fig. 8-4 Screen layout of the preconfigured interface

The illustrations shown below (Fig. 8-5 to Fig. 8-19) provide you with the screen content of the configured screens.

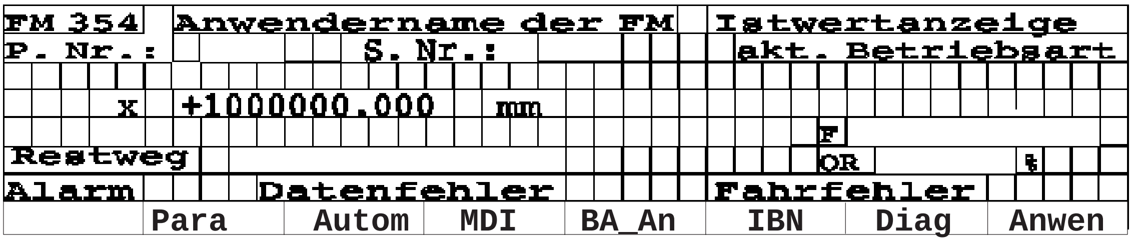


Fig. 8-5 Main screen, PIC7

This screen is displayed to you after the OP 17 is enabled. The FM354 values are display values. The soft keys (F1 - F8) are used to request the continuation screens. The first and last soft keys (User) are kept free for the user to insert his/her own screens (e.g. including additional FMs).

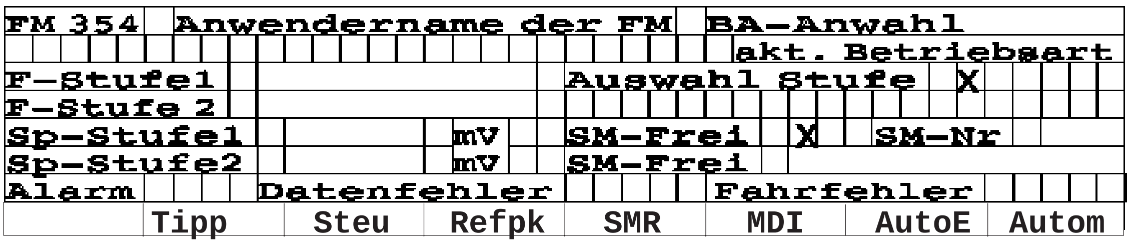


Fig. 8-6 Operating mode selection, PIC75

This screen permits you to enter the operating mode, velocity levels or control levels or the incremental value.

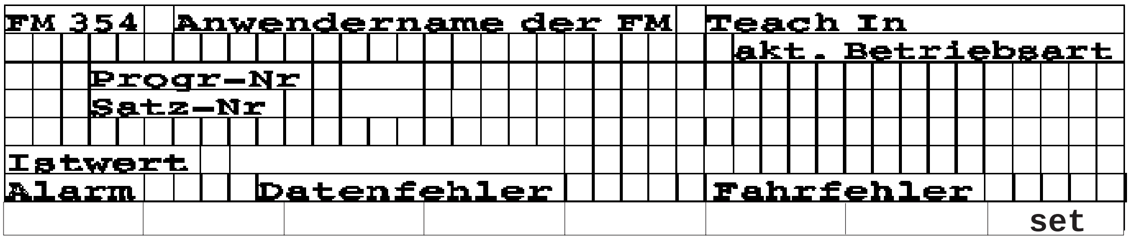


Fig. 8-7 Teach In PIC735

This illustration displays the actual value for you. You can enter the values for Teach In.

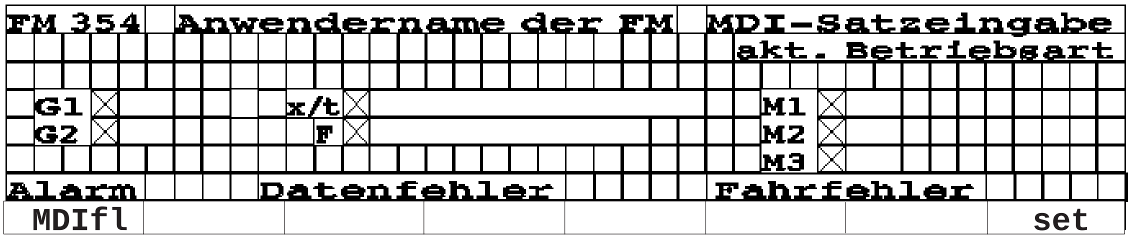


Fig. 8-8 MDI block entry PIC74

The fields identified by an X are text fields and can be toggled back and forth between “X” and “ ”. You can enter the values of the MDI block.

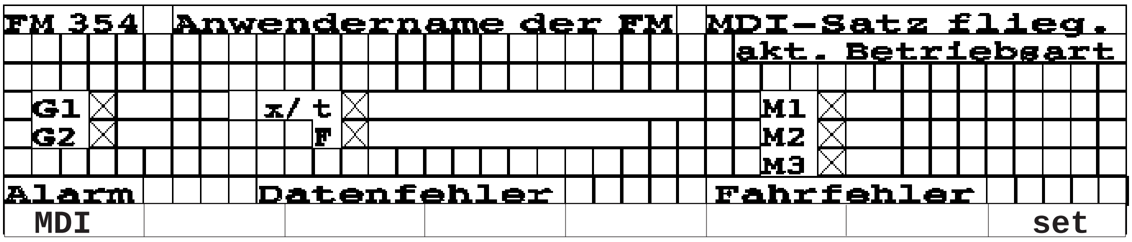


Fig. 8-9 MDI block on the fly PIC741

Handling of the MDI block on the fly corresponds to that of the MDI block.

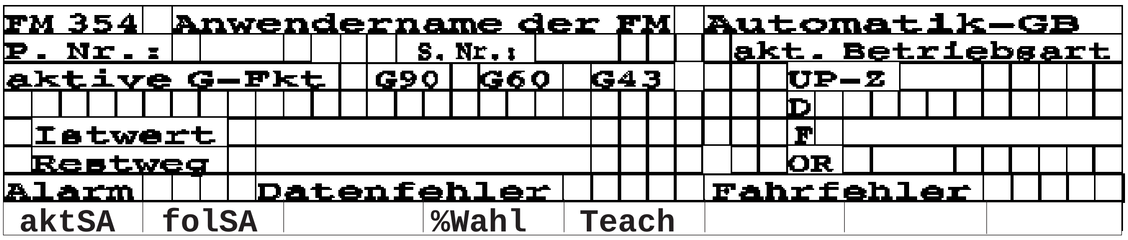


Fig. 8-10 Automatic main screen PIC73

This screen contains only display fields.

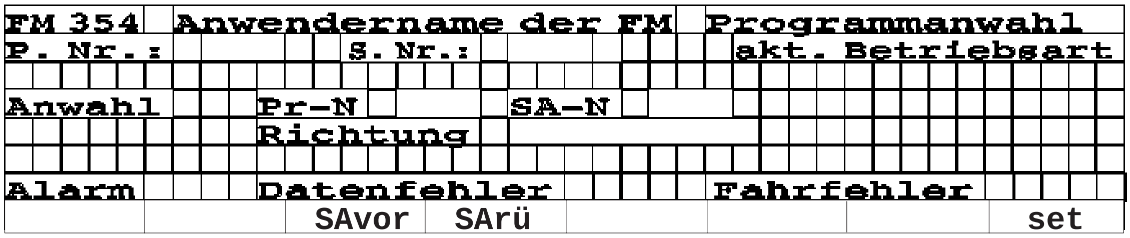


Fig. 8-11 Program selection PIC734

This screen features input/output fields. It is upward and downward selectable.

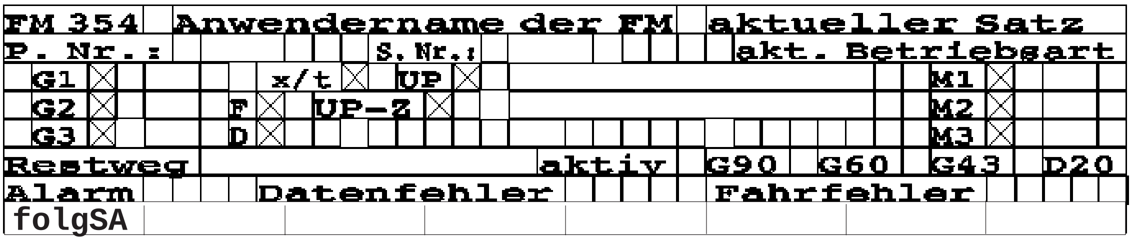


Fig. 8-12 Current block PIC731

This screen contains only display fields.

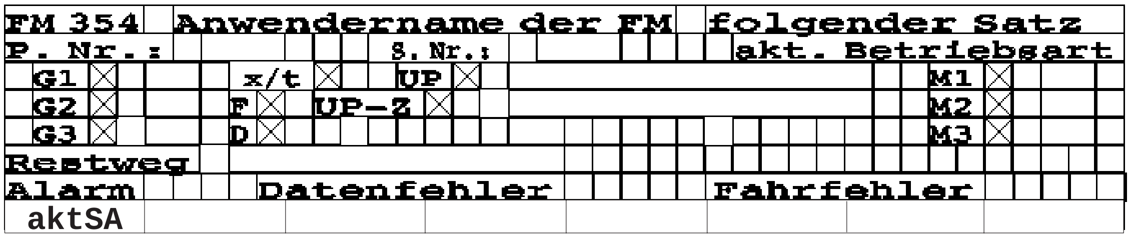


Fig. 8-13 Continuation block PIC 732

This screen contains only display fields.

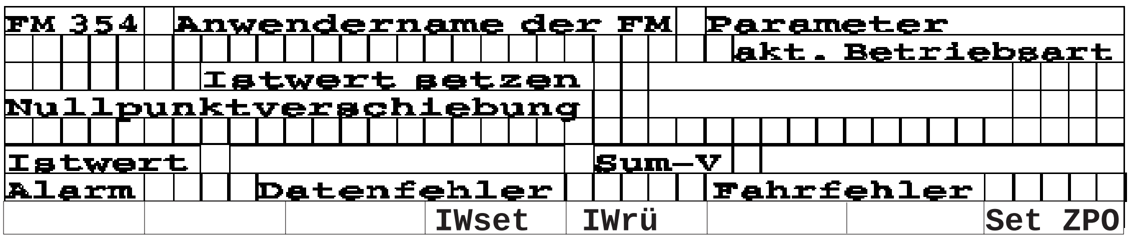


Fig. 8-14 Parameters, PIC72

The sum of the offsets and the actual value display are display fields.

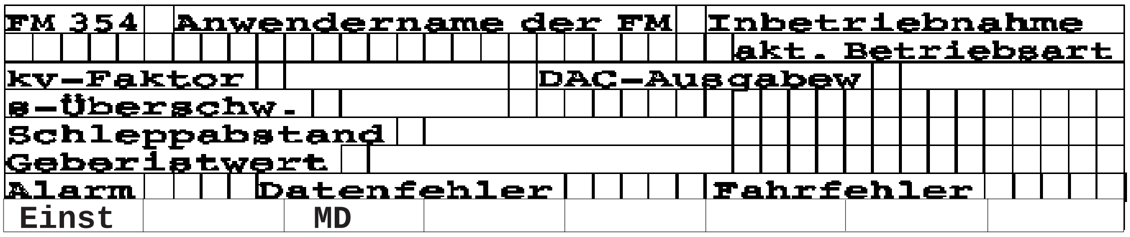


Fig. 8-15 Startup PIC76

The FM 354 values (service data) are display values.

Fig. 8-16 Startup settings PIC761

The fields in which an “X” is entered are display fields.

Fig. 8-17 Machine data PIC763

Value input is password protected.

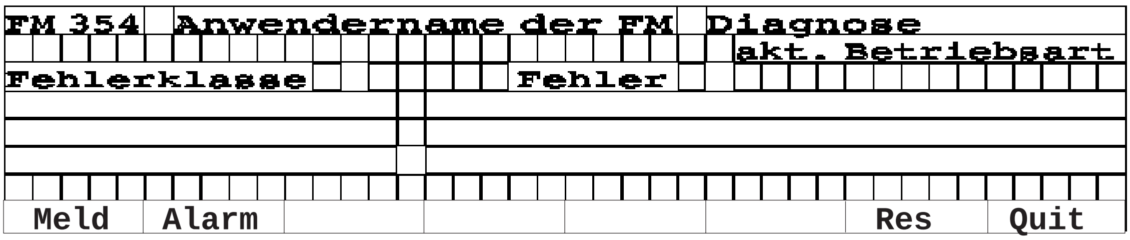


Fig. 8-18 Diagnostics, error message PIC77

This screen displays the FM 354 errors. The fields are display fields.

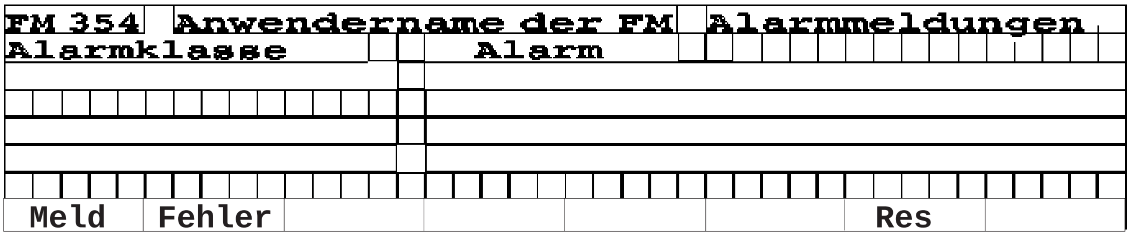


Fig. 8-19 Interrupt messages PIC772

This screen displays the FM 354 errors. The fields are display fields.

The screen “PICZ_MESS_EVENT” contains factory-supplied default “Pro-Tool/Lite” screens for the OP17.

8.2 Analysis of the user DB by the user program for operator control

Overview

The following table describes for you which functions must be executed by the user program. Execution of these functions is triggered by setting/deleting of certain memory bits of the operator panel or by certain events within the FM 354 (e.g. error messages).

Table 8-2 Analysis of the user DB by the user program

OP 07/17	Triggered by...	User program			See PIC...	
Byte.Bit event		Set in user DB (Byte.Bit)	Function	Delete Byte.Bit	OP 07	OP 17
	FM 354	390.13 390.14 390.15	Diagnostic interrupt Data errors Operator/travel errors		–	7
390.9 = 1 390.10 = 1 42.14 = 1	SK “IWset” SK “NPVset” SK “IWrü”		Transfer data for “set actual value” from user DB to the FM Transfer data for “Zero point offset” from user DB to the FM Transfer “Remove setting actual value” flag to the FM	390.9 390.10 42.14	714 7141 714	72
42.10 = 1 42.11 = 1 390.3 = 1	SK “SAvor” SK “SArü” SK “set”		Transfer “Automatic block search, forward” memory bit to the FM Transfer the “Automatic block search, backward” memory bit to the FM Transfer data for “Program selection” from user DB to the FM	42.10 42.11 390.3	7132	734
390.4 = 1	SK “set”		Transfer data for “Teach In” from the user DB to the FM	390.4	711	735
390.2 = 1	SK “set”		Transfer data for “MDI block entry” from the user DB to the FM	390.2	7124	74
390.8 = 1	SK “set”		Transfer data for “MDI block on the fly” from the user DB to the FM	390.8	–	741
40.0 40.6	TF “servo enable” TF “park. axis”		In case of modification, transfer “Servo enable” yes/no to the FM In case of modification, transfer “Parking axis” yes/no to the FM		723	761

SK = Soft key, TF = Text field

Table 8-2 Analysis of the user DB by the user program, continued

OP 07/17	Triggered by...	User program			See PIC...	
Byte.Bit event		**Set in user DB (Byte.Bit)**	**Function**	**Delete Byte.Bit**	**OP 07**	**OP 17**
406.6 = 1	SK “Jog”		Transfer data for “Jogging” mode and the “Jogging” mode to the FM	406.6	–	75
406.0 = 1	SK “Control”		Transfer data for the “Control” mode and the “Control” mode to the FM	406.0		
406.1 = 1	SK “Refpt”		Transfer the “Reference point approach” mode to the FM	406.1		
406.2 = 1	SK “SMR”		Transfer data for “Incremental relative” mode and the incremental relative mode to the FM	406.2		
406.3 = 1	SK “MDI”		Transfer “MDI” mode to the FM	406.3		
406.4 = 1	SK “AutoE”		Transfer “Automatic single block” mode to the FM	406.4		
406.5 = 1	SK “Autom”		Transfer “Automatic” operating mode to the FM	406.5		
40.14	TF “software limit switch off”		In case of modification, transfer “Software limit switch disable” yes/no to the FM		723	761
42.13 = 1	TF “Restart axis”		Transfer “Restart axis” memory bit to the FM	42.13	723	
42.9 = 1	TF “Delete distance to go”		Transfer “Delete distance to go” memory bit to the FM	42.9	–	
390.1 = 1	SK “read”		Read MD No. from the user DB, retrieve its value from the FM and enter into the user DB	390.1	722	763
42.8 = 1	SK “active”		Transfer “activate MD” to the FM	42.8		
390.0 = 1	SK “set”		Transfer MD No. and its value from the user DB to the FM	390.0		
406.15 = 1	SK “Res”		Error acknowledgment “Res” in FM 354 (diagnostic interrupt)	406.15 390.13	73	77
406.14 = 1	SK “Ack”		Error acknowledgment “Quit” in the FM 354 (data errors, operator/travel errors)	406.14 390.14 390.15		

SK = Soft key, TF = Text field

Variables in the user DB

The following table contains the variables which are entered into the user DB.

See Section 6.6 for the structure of the user DB.

Table 8-3 Variables for user DB

Absolute address	Variable type	Significance	Job no.
23	BYTE	Velocity or voltage level 1, 2 [BP]	–
 40.0 40.6 41.6	16 BOOL	Single functions Servo enable Parking axis Deactivate software end position monitoring	10
 43.0 43.1 43.2 43.3 43.5 43.6	16 BOOL	Single commands Activate machine data Delete distance to go Automatic block search forward Automatic block search backward Restart Undo set actual value	11
44	DINT	Zero offset	12
48	DINT	Set actual value	13
86	DWORD	Increment for incremental dimensions	3
90	DWORD	Velocity level 1	1
94	DWORD	Velocity level 2	
98	DWORD	Voltage level 1	2
102	DWORD	Voltage level 2	
106	STRUCT NC block	MDI block	6
152	STRUCT NC block	MDI block on the fly	16
172	BYTE	Program selection – program number	17
173	BYTE	Program selection – block number	
174	BYTE	Program selection – direction	
180	BYTE	Teach In – program number	19
181	BYTE	Teach In – block number	

Table 8-3 Variables for user DB, continued

Absolute address	Variable type	Significance	Job no.
	16 BOOL	Function bits for the user program	–
390.0		Write MD	
390.1		Read MD	
390.2		Transfer MDI block	
390.3		Transfer program selection	
390.4		Transfer Teach In	
390.5		Transfer increment	
390.6		Transfer velocity levels	
390.7		Transfer voltage levels	
391.0		Transfer MDI block on–the–fly	
391.1		Transfer set actual value	
391.2		Transfer zero offset	
391.5		Diagnostic interrupt	
391.6		Data error	
391.7		Operator/travel error	
392	WORD	MD No.	–
394	DINT	MD value	–
398	BYTE	SM No.	–
	16 BOOL	Mode selection/change input to corresponding mode	–
406.0		Open–loop control	
406.1		Reference point approach	
406.2		Incremental relative	
406.3		MDI	
406.4		Automatic single block	
406.5		Automatic	
406.6		Jogging	
407.6		Acknowledge error (“Quit” softkey)	
407.7		Acknowledge diagnostic interrupt (“Res”softkey)	

8.3 Data block for status messages (DB-SS)

Overview

The following table contains the parameters/data which are readable during operation.

Table 8-4 Parameters/data of DB-SS

Byte	Variable type	Value	Significance of the variables	Comment
0...35			DB header	
36...59			Internal header information	
Offset[1)]	**Variable type**	**Value**	**Significance of the variables**	**Comment**
24	8 x BOOL		Control signals	Byte 0
25	8 x BOOL		Control signals	Byte 1
26	2 x BYTE		Control signals	Byte 2, 3
28	2 x BYTE		Control signals	Byte 4, 5
30	8 x BOOL		Checkback signals	Byte 0
31	8 x BOOL		Checkback signals	Byte 1
32	BYTE		Checkback signals	Byte 2
33	8 x BOOL		Checkback signals	Byte 3
34	BYTE		Checkback signals	Byte 4
35	8 x BOOL		Checkback signals	Byte 5
36	12 x BYTE		Reserved	
48	DWORD		Velocity level 1	
52	DWORD		Velocity level 2	
56	DWORD		Control level 1	
60	DWORD		Control level 2	
64	DWORD		Setpoint for incremental value	
68	STRUCT	MDI block structure	MDI block	
88	16 x BOOL		Single functions	
90	16 x BOOL		Single commands	
92	DINT		Zero offset	
96	DINT		Set actual value	
100	DINT		Set actual value on the fly	
104	16 x BOOL		Digital inputs/outputs	
106	STRUCT	MDI block structure	MDI block on the fly	
126	BYTE		Program selection	Program number

1) A variable in the S7 protocol is addressed by the DB No. and, depending on data format, by the DBB, DBW and DBD No. (offset in DB), as well.

Table 8-4 Parameters/data of DB-SS, continued

Offset[1]	Variable type	Value	Significance of the variables	Comment
127	BYTE		Program selection	Block number
128	2 x BYTE		Program selection	Direction, free
130	4 x BYTE		Request application data	Application data 1-4
134	BYTE		Teach In	Prog. no.
135	BYTE		Teach In	Block number
136	DINT		Reference coordinate	
140	4 x DINT		Free	
156	DINT		Actual position	Basic operating data
160	DINT		Actual velocity	Basic operating data
164	DINT		Residual travel	Basic operating data
168	DINT		Target position	Basic operating data
172	DINT		Sum of current tool offset	Basic operating data
176	DINT		Rotational speed, rotary axis	Basic operating data
180	DINT		Free	
184	DINT		Free	
188	STRUCT	NC block structure	Active NC block	
208	STRUCT	NC block structure	Next NC block	
228	DINT		Code application 1	Application data
232	DINT		Code application 2	Application data
236	DINT		Code application 3	Application data
240	DINT		Code application 4	Application data
244	DINT		Actual position on leading edge	Length measurement/in-process measurement
248	DINT		Actual position on trailing edge	Length measurement
252	DINT		Length measurement value	Length measurement
256	DINT		Actual value at external block change	
260	DINT		DAC value/frequency	Servicing data
264	DINT		Encoder actual value	Servicing data
268	DINT		Error pulses	Servicing data
272	DINT		K_V-factor	Servicing data
276	DINT		Following error	Servicing data
280	DINT		Following error limit	Servicing data
284	DINT		s Overshoot/Switch readjustment in Reference Point Approach mode	Servicing data

1) A variable in the S7 protocol is addressed by the DB No. and, depending on data format, by the DBB, DBW and DBD No. (offset in DB), as well.

Table 8-4 Parameters/data of DB-SS, continued

Offset[1)]	Variable type	Value	Significance of the variables	Comment
288	DINT		Approach time Te (ms)/drive constant Ta (ms) in OL Control mode	Servicing data
292	8 x DINT		Free	
324	BYTE		Override	Additional operating data
325	BYTE		Part program no.	
326	BYTE		Part program block no.	Additional operating data
327	BYTE		No. of callup subroutine loops	Additional operating data
328	BYTE		G90/91 Active	Additional operating data
329	BYTE		G60/64 Active	Additional operating data
330	BYTE		G43/44 Active	Additional operating data
331	BYTE		Active D No.	Additional operating data
332 332.1 332.2 332.3	8 x BOOL		Status messages 1 • Speed limitation to limit value from MD • Limitation to $\pm$10 V • Limitation of minimum acceleration or minimum deceleration in effect	Additional operating data
333	8 x BOOL		Free	
334	2 x BYTE		Free	
336	4 x 8 x BOOL		Diagnostics, system-specific	
340	4 x BYTE		Diagnostics, channel-specific	Identifier
344	2 x 8 x BOOL		Diagnostics, channel-specific	Channel error
346	4 x 8 x BOOL		Free	
350	2 x BYTE		Error number DS 162	Operator/travel error
352	BYTE		Free	
353	BYTE		Free	
354	2 x BYTE		Error number DS 163	Data error
356	BYTE		Free	
357	BYTE		Free	
358	2 x BYTE		Error number DS 164	Operator control error
360	BYTE		Free	
361	BYTE		Free	
362	32 x BOOL		Process interrupt	

1) A variable in the S7 protocol is addressed by the DB No. and, depending on data format, by the DBB, DBW and DBD No. (offset in DB), as well.

The control and checkback signals in Table 8-4 can be the following signals:

Byte \ Bit	7	6	5	4	3	2	1	0
Control signals:								
20					BFQ/FSQ		TFB	
21	AF	SA	EFG	QMF	R+	R–	STP	ST
22	operating mode							
23	BP							
24	OVERR							
25								
Checkback signals:								
28	PARA			DF	BF/FS		TFGS	
29		PBR	T-L			WFG	BL	SFG
30	BAR							
31	PEH		FIWS		FR+	FR–	ME	SYN
32	MNR							
33				AMF				

The following table describes the control and checkback signals in German and English.

Table 8-5 Control and checkback signals

German	English	Significance
Control signals		
BP	MODE PA-RAMETER	Operating mode parameters Velocity levels 1 and 2 Frequency levels 1 and 2 Increment selection 1...100, 254
BAoperat-ing mode	MODE	Operating mode Code Jogging 01 Open–loop control 02 Reference point approach 03 Incremental relative 04 MDI 06 Automatic 08 Automatic single block 09
R+	DIR_P	Direction plus
R–	DIR_M	Direction minus
STP	STOP	Stop
ST	START	Start
OVERR	OVERRIDE	Override
AF	DRV_EN	Drive enable

Table 8-5 Control and checkback signals, continued

German	English	Significance
SA	SKIP_BLK	Enable bit for block skip
EFG	READ_EN	Read enable
QMF	ACK_MF	Acknowledgment M function
BFQ/FSQ	OT_ERR_A	Acknowledgment operator control and guidance error
TFB	TEST_EN	Switchover, P-BUS port
Checkback signals		
MNR	NUM_MF	M function number
BL	WORKING	Program running
SFG	START_EN	Start enable
BF/FS	OT_ERR	Travel/operator error
BAR	MODE	Active operating mode
AMF	STR_MF	Modify M function
PBR	PR_BACK	Program scanning backward
T-L	DT_RUN	Dwell time running
PEH	POS_ROD	Position reached and stopped
FR+	GO_P	Travel Plus
FR–	GO_M	Travel Minus
ME	MSR_DONE	Measurement done
SYN	SYNC	synchronized
DF	DATA_ERR	Data error
FIWS	FAVEL	Set actual value on the fly ready
TFGS	TST_STAT	Switchover, P-BUS port done
WFG	WAIT_EN	Wait for external enable
PARA	PARA	parameterized

■

9 Description of Functions

Summary

This chapter describes the functions of the FM 354.

By calling up the appropriate functions (FCs) you can activate these functions by way of the user program.

Chapter overview

9.1 Control/checkback signals

Overview

FC MODE_WR transfers the control signals from the user DB to the FM and transfers the checkback signals from the FM to the user DB.

Byte \ Bit	7	6	5	4	3	2	1	0
Control signals:								
20					BFQ/FSQ		TFB	
21	AF	SA	EFG	QMF	R+	R–	STP	ST
22	BA							
23	BP							
24	OVERR							
25								
Checkback signals								
28	PARA			DF	BF/FS		TFGS	
29		PBR	T-L			WFG	BL	SFG
30	BAR							
31	PEH		FIWS		FR+	FR–	ME	SYN
32	MNR							
33				AMF				

9.1.1 Control signals

Overview

The axis is operated and controlled by means of control signals.

Table 9-1 describes the control signals and their functions.

Table 9-1 Control signals

Symbol		Name	Function
English	**German**		
TEST_EN	TFB	Sw./over P-bus interface	Interrupts communication with the user program, and switches over the P bus interface for operation with the start–up user interface.
OT_ERR_A	BFQ/ FSQ	Acknowledge operato/ travel error	... resets an error message. Before acknowledging the error, correct its cause.
START	ST	Start	... starts movement in Automatic, MDI and Reference-point approach modes.
STOP	STP	Stop	... interrupts movement or processing of the program. ... cancels reference point approach.
DIR_M	R–	Direction minus	... moves axis in negative direction. • In Jogging and Control modes, moves axis in negative direction (level-dependent). • Starts movement in negative direction in Incremental relative and Reference-point approach modes. • Specifies direction of movement for rotary axes in MDI and Automatic modes.
DIR_P	R+	Direction plus	... moves axis in positive direction. • In Jogging and Control modes, moves axis in positive direction (level-dependent). • Starts movement in positive direction in Incremental relative and Reference-point approach modes. • Specifies direction of movement for rotary axes in MDI and Automatic modes.
ACK_MF	QMF	Acknowledge M function	... only “acknowledge–driven” during M function output (see machine data list in Table 5-5, MD32). ... acknowledges receipt of M function. Program sequence can be continued.
READ_EN	EFG	Read–in enable	... prevents read-in (processing) of the next block. ... has effect only in Automatic mode. The read–in enable is required in order to read in the next traversing block during program execution.
SKIP_BLK	SA	Skip block	... skips identified blocks in the program. ... has effect only in Automatic mode.

Table 9-1 Control signals, continued

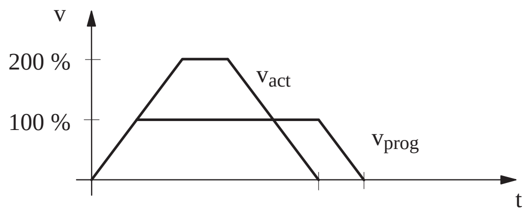
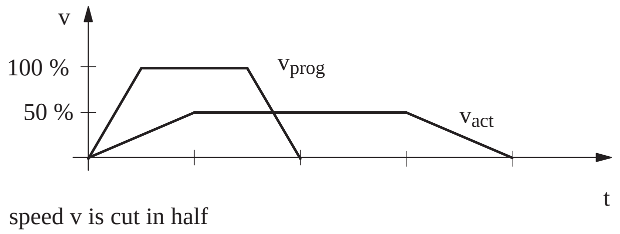
Symbol		Name	Function
English	**German**		
DRV_EN	AF	Drive en- able	... enables movement. When the signal is reset, a **rapid deceleration** of the movement takes place. On MD 37.15 = 0 program execution, or the movement, is canceled and the residual distance is deleted. On MD 37.15 = 1 machining resumes after emergency stop • Rapid deceleration of the movement. • On axis standstill FR+ or FR– = 0; BL = 1 – if the drive remains activated and the servo enable active, the axis is held in position control mode. – if the drive is deactivated, the user must activate “follow–up”. This deactivates zero speed control (the axis can be pushed away). • If an error occurs in this state (e.g. if the user starts a movement without a start enable, etc.), an error response is initiated, e.g. the residual distance is deleted, BL = 0, (a new path default must be defined).
MODE	BA	Operat- ing mode	Operating mode (see Section 9.2) Code Jogging 01 Open–loop control 02 Reference point approach 03 Incremental relative 04 MDI 06 Automatic 08 Automatic single block 09
MODE PA- RAMETER	BP	Mode pa- rameter	... selects speed levels in Jogging mode. ... selects voltage levels in Control mode. ... selects increment in Incremental relative mode (value 1...100 or 254).
OVERRIDE	OVERR	Override	... affects response of traversing movement. Range: 0-255% ... override has no effect in Control mode • Velocity override Range: 0-255% Speed adjusted by percentage **Example:** Override doubled from 100% to 200%  – speed v is doubled – acceleration and deceleration values are not affected $v_{act} = \frac{v_{prog} \cdot Override}{100}$ The positioning time is not cut in half.

Table 9-1 Control signals, continued

Symbol		Name	Function
English	German		
OVERRIDE	OVERR	Override	• Time override If you parameterize the “time override” function in MD37, there are two ranges: – range 100-255%: speed override operates as described above – range 0-100%: time override operative Speed, acceleration and deceleration are changed in such a way that the time necessary for the traversing movement is directly correlated with the override value. **Example:** Cut override in half, from 100% to 50%  – speed v is cut in half – acceleration and deceleration are quartered $v_{act} = \frac{v_{prog} \cdot Override}{100}$ $a_{act} = \frac{a \cdot Override^2}{100^2}$ $t_{act} = \frac{t \cdot 100}{Override}$ Positioning time is doubled. Taking the override into account as a time override presupposes the following additional condition: If a traversing movement consists of multiple positioning blocks with block change on-the-fly (the axis does not stop between blocks), changing the override value affects only the speed. Acceleration and deceleration are additionally affected only after the axis comes to a stop (e.g. reversal of direction). **Note:** Time override has effect only in the MDI and Automatic modes.

Note

For further functions, **settings and commands** concerning open–loop control, see Section 9.3.2 and Section 9.3.3.

9.1.2 Checkback signals

Overview

The checkback signals indicate the processing status of the axis and report it to the user program.

Table 9-2 describes the checkback signals and their functions.

Table 9-2 Checkback signals

Symbol		Significance	Function
English	**German**		
TST_STAT	TFGS	Sw./over P bus interface complete	Communication with the user program is not possible, since the P bus interface has been switched over for operation with the start–up tool.
OT_ERR	BF/FS	Operator/travel error	... signaled to the user if an operator-control error or travel error is pending (e.g. unallowed control signal has been set, (R+) and (R–) set simultaneously) An error message causes the movement to be canceled. see Chapter 11
DATA_ERR	DF	Data error	... is reported to the user when a data error occurs. see Chapter 11
PARA	PARA	Parameterize	... module parameterized. All machine data applicable for control of an axis are present on the module.
START_EN	SFG	Start enable	... signals that the FM 354 is ready for positioning and output. • “Start enable” is set: – if no static stop or error is pending and the drive enable is pending – if the mode setting and mode checkback match (after mode change) – if no axis functions (including M output, dwell time) are active, or after functions have been completed – for further processing of a function interrupted with unprogrammed stop – in Automatic mode, after M0, M2, M30 or at end of block in Automatic single-block mode • “Start enable” is deleted: – if a function has been started and is active, or – if a start condition is active (stat.) – if there is an error and an unprogrammed stop – in follow–up mode • Without Enable Start, none of the functions that can be operated with Travel Plus, Travel Minus and Start can be executed.

Table 9-2 Checkback signals, continued

Symbol		Significance	Function
English	German		
WORKING	BL	Processing in progress	... indicates that a function has been started with Start or Travel Plus/Minus, and is active. • "Processing in progress" is set with: – "Jogging", "Control" mode during the movement up to standstill after cancelation of R+, R– – Reference-point approach mode, during approach until reference point is reached – "MDI", "Incremental relative mode", during the positioning process or while functions of the MDI block are being processed – Automatic mode, during processing of a traversing program until the end of the program. • "Processing in progress" is deleted: – by errors and restarts – by mode changes. – after axis standstill
WAIT_EN	WFG	Wait for external enable	... takes effect only if a digital input has been parameterized by means of MD34 (see Section 9.8.1). Set: if the enable input has not yet been set or has been reset when a movement has been activated.
DT_RUN	T-L	Delay time running	... only active in Automatic and MDI mode. As soon as a traversing block with a dwell time has been processed, (T-L) is output during the programmed time period.
PR_BACK	PBR	Processing in reverse	... is set after a Start in Automatic mode if a program is being processed in reverse.
MODE	BAR	Active mode	The selected mode is not fed back until it is internally active. For a mode change, for example, a movement must be stopped before another mode can become active (does not apply to switching between Automatic and Automatic single-block modes).
SYNC	SYN	Synchronism	... module is synchronized (see Section 9.6.3) Required for axis motion in modes: • Incremental Relative • MDI • Automatic
MSR_DONE	ME	End of meas.	... signals an executed measurement (see Section 9.3.10)
GO_P GO_M	FR+ FR–	Travel plus Travel Minus	... means the axis is traveling in the direction of increasing actual values or in the direction of voltage output "+" in OL control mode. ... means the axis is traveling in the direction of decreasing actual values or in the direction of voltage output "–" in OL control mode. • As soon as an active travel movement is pending, the messages (FR+) or (FR–) are output depending on the traversing direction. They can only be pending as alternatives. • "Travel Plus" or "Travel Minus" is actuated at the start of the acceleration phase and remains active until the axis comes to a standstill or the POS_ROD target area has been reached.

Table 9-2 Checkback signals, continued

Symbol		Significance	Function
English	German		
FAVEL	FIWS	Set actual value on-the-fly complete	... set Actual value on-the-fly is executed. The signal is reset when "Set actual value on-the-fly" is activated (see Section 9.3.6).
NUM_MF	MNR	M function number	M command 0...99
STR_MF	AMF	Change M function	... is indicated simultaneously with the M function number. • If M functions are programmed in a traversing block, their output is signaled by setting "Change M function." • "Change M function" remains pending until: – the specified time has expired, for time-controlled M functions – the user has acknowledged, for acknowledgment-controlled M functions.
POS_ROD	PEH	Position reached, Stop ("PEH")	• When the preset target position is reached correctly, (PEH) is actuated, and remains in effect until the next axis movement. • "Target position reached correctly" means that during approach of the actual value to target position, a defined tolerance (PEH tolerance) must not be exceeded during a defined time (PEH time watchdog). If this is not the case, an error is signaled and positioning is interrupted. • (PEH) is actuated only in the following modes and cases: – Reference-point approach: If the reference point has been reached in full (including reference-point shift). – "MDI", "Incremental relative": If the preset position has been reached. – Automatic: If a traversing block has been positioned in full and the axis remains motionless until the next traversing movement. • It is not set if no synchronization is available yet.

9.1.3 General handling information

Overview

Before data/settings can be transferred to the FM 354, an operating mode must be active (e.g. “Jogging” mode = 1 and MODE = 1). That means that communication with the FM 354 has been initiated and the FM 354 has access to valid machine data.

Operating modes (codes)	Relevant control signals	Relevant checkback signals	Required data/ settings job no.
Jogging (01)	[R+], [R–], [STP], [AF], [OVERR], [BP] = 1 or 2	[BL], [SFG], [FR+], [FR–], [SYN], [WFG]	1, 10 (servo enable)
Control (02)	[R+], [R–], [STP], [AF], [BP] = 1 or 2	[BL], [SFG], [FR+], [FR–], [WFG]	2
Reference point approach (03)	[R+], [R–], [ST], [STP], [AF], [OVERR]	[BL], [SFG], [FR+], [FR–], [WFG], [SYN], [PEH]	10 (servo enable)
Incremental relative (04)	[R+], [R–], [STP], [AF], [OVERR], [BP] = 1...100 for increment table or 254	[BL], [SFG], [FR+], [FR–], [WFG], [SYN], [PEH]	1, 10 (servo enable), 3 (only if BP = 254, if BP = 1...100 the appropriate increments must be parameterized)
MDI (06)	[ST], [STP], [AF], [QMF], [OVERR]	[BL], [SFG], [FR+], [FR–], [WFG], [SYN], [PEH], [AMF], [MNR], [T-L]	6, 10 (servo enable)
Automatic (08) Automatic single block (09)	[ST], [SA], [EFG], [STP], [AF], [QMF], [OVERR]	[BL], [SFG], [FR+], [FR–], [WFG], [SYN], [PEH], [AMF], [T-L], [PBR], [MNR]	17 (assuming the corresponding traversing program was parameterized), 10 (servo enable)

Error condition:

- Message via BF/FS – acknowledgement with BFQ/FSQ
- Message via DF – acknowledgement on next correct data transfer
- Message via diagnostic interrupt – acknowledgement on “Restart” (job no. 11)

Hints to the user

Here are a few hints for starting a movement and about the response of the FM 354 to a change of the status of the S7-300 CPU:

It is assumed that the FM 354 has been parameterized correctly.

- First a mode must be set. The servo enable must subsequently be set in order to prevent the axis from “running away,”.
- Before starting a movement in a mode, first transfer the appropriate reference data (e.g. speed levels); the override must be > 0.

- It is only possible to start the movement when the start enable is set and the enable input is set (if parameterized).

  Enable Start is set if:

  – No error occurred

  – Mode is active

  – No Stop is called

  – Drive enable is set

- A static Stop signal prevents all movements or block processing.
- Response of the FM 354 to transition of the S7-300 CPU from RUN to STOP state:

  – As described for Restart (see Section 9.3.3)

  – The digital outputs are switched off

  – Interface to the user program is switched off

- Response of the FM 354 to transition of the S7-300 CPU from STOP to RUN state:

  A cold restart of the module is executed.

Module control

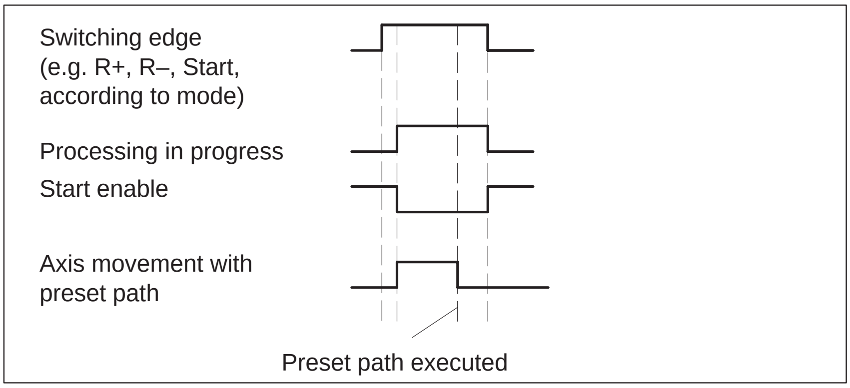
The following table lists the control signals used to start a movement.

Prerequisite: Drive enable [AF] = 1, Stop [STP] = 0,
Start enable [SFG] = 1

Mode	Parameters	Command / Signal state	Activation of movement
Jogging (mode = 01)	Velocity level BP = 1 = level 1 BP = 2 = level 2	R+, R– / Level	R+ or R– with “Level” = 1 (R+ and R– simultaneously → error)
Open–loop control (mode = 02)	Frequency level BP = 1 = level 1 BP = 2 = level 2	R+, R– / Level	R+ or R– with “Level” = 1 (R+ and R– simultaneously → error)
Reference point approach (mode = 03)	–	Start, R+, R– / Edge	Direction as in MD R+ or R– = 0/1 or Start = 0/1 (speed as in MD)
Incremental relative (mode = 04)	BP = 1...100 BP = 254	R+, R– / Edge	R+ = 0/1 or R = 0/1 (speed level 1)
MDI (mode = 06)	–	Start / Edge	Start = 0/1 (R+, R– relevant only for rotary axis with absolute measure specified for direction selection)
Automatic (mode = 08)	–	Start / Edge	Start = 0/1 (according to program presetting)
Automatic single block (mode = 09)	–	Start / Edge	Start = 0/1

Stat. pending start condition

“Processing in progress” remains active after the end of machining and there is no start enable as long as the start condition is not reset.



The following table lists the control signals used to interrupt/terminate a movement.

Mode	Interrupt movement	Continue movement	Interrupt/ end movement, Stop
Jogging (mode = 01)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1	R+ or R– with “Level” = 0 or mode change Drive enable = 0[2]
Control (mode = 02)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1	R+ or R– with “Level” = 0 or mode change Drive enable = 0[2]
Reference point approach (mode = 03)	–	–	Stop = 0/1 or ref. received or mode change or enable input[1] = 0 Drive enable = 0[2]
Incremental relative (mode = 04)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1, with R+ or R–	Position reached or mode change Drive enable = 0[2]
MDI (mode = 06)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1, with Start = 0/1	Position reached or “block” processed or mode change Drive enable = 0[2]
Automatic (mode = 08)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1, with Start = 0/1	Program end or mode change New program selected after stop Drive enable = 0[2]
Automatic single block (mode = 09)	Stop = 1 or Enable input[1] = 0	Stop = 0 or Enable input[1] = 1, with Start = 0/1	Program end or mode change New program selected after stop Drive enable = 0[2]

1) **Prerequisite:** Digital input defined in MD34; see Section 9.8.1
2) if MD37.15 not defined, see Table 9-1 Control signal [AF]

9.2 Operating modes

Overview

The following modes are available on the FM 354:

- Jogging (ER) Code 01
- Open–loop control (STE) Code 02
- Reference point approach (REF) Code 03
- Incremental relative (SMR) Code 04
- MDI (Manual Data Input) Code 06
- Automatic (A) Code 08
- Automatic single block (AE) Code 09

Selecting the mode

FC 2 MODE_WR is called up in order to transfer the operating mode (code), which the user program entered in the user data block, to the FM 354.

The axis is controlled by enabling and disabling appropriate control signals.

Checkback signal for mode

When the specification is allowed, the FM 354 feeds back the specified mode to the user program. If this checkback mode matches the specified one, the mode is active.

Changing modes

Changing modes triggers an internal stop.

If a mode change is attempted while a traversing movement is in progress, the modes are not switched until the axis comes to a stop. The mode check-back is performed after the movement in the old mode is completed.

This does not apply to changes between Automatic and Automatic Single-Block mode.

9.2.1 Jogging

Overview

In Jogging mode, axis traversing movements are specified by way of the direction keys (R+ or R–) and by speed.

Velocity

Before the axis can be moved, the velocities must first be transferred to the FM 354 with **job no. 1**.

You can choose between two mutually independent velocities (level 1 and level 2) with the mode parameter (BP).

The velocity can also be controlled using the override and can be changed during the movement.

Name	Lower input limit	Upper input limit	Unit
Speed	10	500 000 000	MSR/min

MSR stands for measurement system raster (see Section 5.3.1)

Handling by the user

The table below gives you an overview of how to handle this mode.

Triggering of movement, Direction (R)	Level selection	Speed
R+ or R– “level-controlled”	BP = 1	Value for speed level 1
	BP = 2	Value for speed level 2

Note

Please see also Section 9.1.3!

Control actions

Preconditions:

- The FM 354 has been parameterized.
- The mode has been selected and confirmed
- Drive enable [AF] = 1 (control signal, FC MODE_WR)
- Stop [STP] = 0 (control signal, FC MODE_WR)
- Servo enable (RF) = 1 (FC MODE_WR, job no. 10)
- Velocity levels have been transferred (FC MODE_WR, job no. 1)

Table 9-3 Control actions for “Jogging” mode (examples)

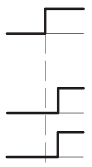
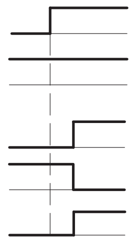
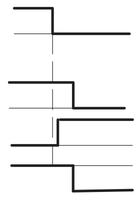
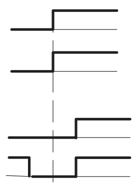
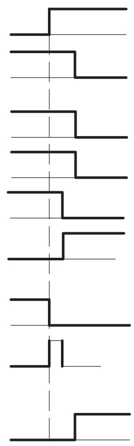
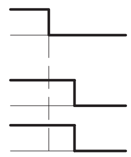
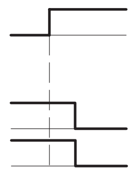
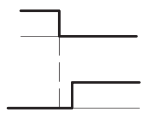
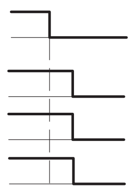
Signal name	Level	Explanation
Control action 1, enable “Jogging” mode		
Control signal: Mode [BA] **Checkback signals:** Active mode [BAR] Start enable [SFG]		The user initiates a [BA] command. The module returns [BAR] and [SFG].
Control action 2, move axis – positive direction		
Control signals: Direction plus [R+] Drive enable [AF] **Checkback signals:** Travel plus [FR+] Start enable [SFG] Processing in progress [BL]		When [SFG] and [AF] are active, [R+] is actuated. The axis cancels the [SFG] and outputs messages [BL] and [FR+]
Control action 3, deactivate axis – positive direction		
Control signal: Direction plus [R+] **Checkback signals:** Travel plus [FR+] Start enable [SFG] Processing in progress [BL]		[R+] is canceled When the axis has come to a standstill by way of the deceleration ramp, the [BL] and [FR+] messages are canceled and [SFG] is activated. Before the axis comes to a standstill, it is possible to define a new direction “through start”.
Control action 4, move axis – negative direction		
Control signals: Direction minus [R–] Velocity level [BP] **Checkback signals:** Travel minus [FR–] Processing in progress [BL]		[R–] is actuated in combination with velocity level 2. The axis travels at velocity level 2, and returns [BL] and [FR–]. The [SFG] signal is canceled.
Control action 5, switch over set–up velocity		
Control signal: Velocity level [BP]		A switchover from [level 2 to level 1] causes a dynamic transition between velocity levels 1 and 2.

Table 9-3 Control actions for “Jogging” mode (examples), continued

Signal name	Level	Explanation
Control action 6, ambiguous direction command (special situation)		
Control signals: Direction plus [R+] Direction minus [R–] **Checkback signals:** Traverse minus [FR–] Processing in progress [BL] Start enable [SFG] Operator/travel error [BF/FS] **Control signals:** Direction minus [R–] Error acknowledgement [BFQ/FSQ] **Checkback signal:** Start enable [SFG]		[R+] is actuated while the axis is traversing with [R–]. The ambiguous direction command causes the axis to stop and [BF/FS] to be output. [FR–] and [BL] are reset. Only when [R+] is canceled and the error is acknowledged [BFQ/FSQ] is [SFG] actuated again and a new direction command can be initiated.
Control action 7, cancel drive enable (special situation)		
Control signal: Drive enable [AF] **Checkback signals:** Traverse minus [FR–] Processing in progress [BL]		[AF] is deactivated during the traversing movement. The axis is stopped abruptly. [FR–] and [BL] are canceled.
Control action 8, reset during axis motion (special situation)		
Single command “Restart ”, job no. 11 **Checkback signals:** Travel plus [FR+] Processing in progress [BL]		Restart is defined during the traversing movement. The axis is stopped abruptly. [FR+] and [BL] are reset. If incremental encoders are used, resynchronization is necessary. (SYN is cleared)
Control action 9, change direction		
Control signal: Direction plus [R+] **Checkback signal:** Start enable [SFG]		Only when [R+] is canceled is [SFG] reactivated.
Control action 10, change mode		
Control signal: Mode [BA] **Checkback signal:** Active mode [BAR] Travel plus [FR+] Processing in progress [BL]		A new [BA] 1 is preselected during the traversing movement. The axis is stopped by way of the deceleration ramp. [FR+] and [BL] are reset.

9.2.2 Open–loop control

Overview

In Control mode, voltages of various magnitudes are specified and then used to perform a controlled movement. The direction of movement is determined by way of direction keys (R+ or R–).

The actual value of the axis is updated at the same time.

Note

A closed-loop controller, which may have been activated by a servo enable, will be interrupted while the voltage is being output. After the Jogging signals R+ or R– have died off, control is referred to the new actual value, and reinstated after the axis comes to a stop, if the controller enable is still active when the axis stops.

Voltage values

The voltage is defined with **job no. 2**.

You can choose between two mutually independent voltage values (level 1 and level 2) with the mode parameter (BP).

Name	Lower input limit	Upper input limit	Unit
Volts	0	10 000	mV

The values for the voltage levels can be changed during movement.

Handling by the user

The table below gives you an overview of how to handle this mode.

Triggering of movement, Direction (R)	Level selection	Speed
R+ or R– “level-controlled”	BP = 0	Value for voltage level 1
	BP = 1	Value for voltage level 2

Note

Please see also Section 9.1.3!

Control actions

The control and checkback signals are handled in the same way as in “Jogging” mode.

9.2.3 Reference point approach

Overview

In Reference-point approach mode, the direction keys (R+ or R–) or Start are used to position the axis to a point (reference-point coordinate MD16) specified in the machine data.

The axis is thus synchronized (see Section 9.6.3).

The override is set to 100% for the reducing speed.

An active zero offset or Set actual value is reset.

Machine data

The following table lists the machine data that is of significance for reference-point approach:

MD	Designa-tion	Value/Meaning	Comments/ Unit
16	Reference-point coor-dinate	–1,000,000,000...+1,000,000,000	(MSR)
18	Type of ref-erence-point ap-proach	0 = direction +, zero pulse right 1 = direction +, zero pulse left 2 = direction –, zero pulse right 3 = direction –, zero pulse left 4 = direction +, RPS center 5 = direction –, RPS center 8 = direction +, RPS edge 9 = direction –, RPS edge	
27	Reference-point shift	–1,000,000,000...+1,000,000,000	(MSR)
28	Referencing speed	10...v_{max} (MD23)	(MSR/min)
29	Reducing speed	10...x (see Machine data list, Table 5-5)	(MSR/min)
34	Inputs	5 = reference point switch for reference point approach 6 = reversing switch for reference point approach	Assigned de-pending on input

MSR stands for measurement system raster (see Section 5.3.1)

Handling by the user

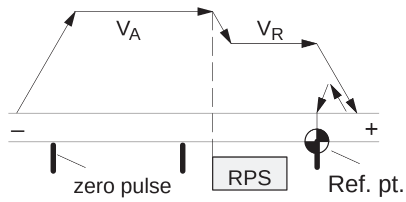
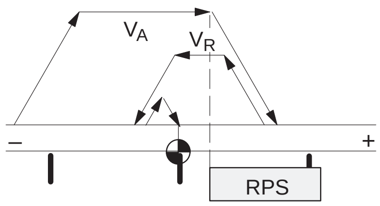
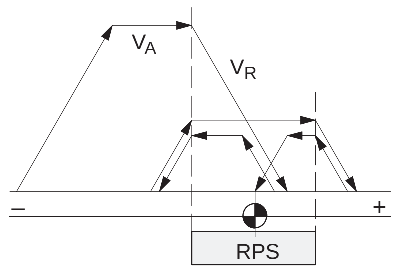
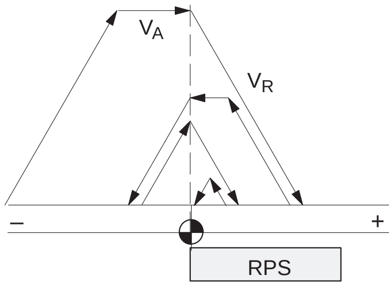
When an absolute encoder is used, only the reference point coordinate defined as a fixed point on the axis is approached in Reference-point approach mode.

When an incremental encoder is used, the user has two options for recording the reference point:

- with connected reference-point switch (RPS)
- without connected reference-point switch (RPS).

With reference point switch (RPS)

It is necessary to connect the reference point switch (RPS) to a digital input and parameterize it in MD34.

Triggering of movement, Direction for synchronization (R)	Type of reference-point approach	Sequence of motions (reference point offset = 0) V_A – referencing velocity V_R – reducing velocity
R+ (“edge-controlled”) or Start	1st situation zero pulse to right of RPS	
	2nd situation zero pulse to left of RPS	
	3rd situation RPS centered (no zero pulse necessary)	
	4th situation RPS edge (no zero pulse necessary)	
R– (“edge-controlled”) or Start	1st situation see above	equals R+ 2nd situation mirrored
	2nd situation see above	equals R+ 1st situation mirrored
	3rd situation RPS centered (no zero pulse necessary)	equals R+ 3rd situation mirrored
	4th situation RPS edge (no zero pulse necessary)	equals R+ 4th situation mirrored

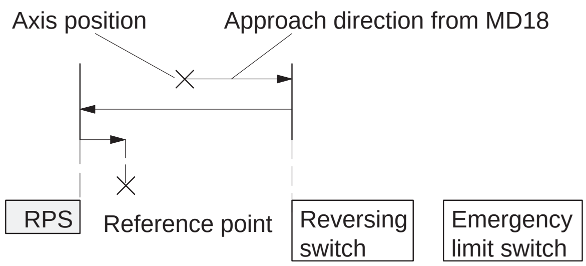
When crossing the RPS, a signal length of $\Delta t \geq 2 \cdot$ FM cycle must be assured!

Using a reversing switch

If it is possible for the axis to be “behind” the reference point switch when you start reference point approach, a reversing switch can be installed at the end of the axis in the approach direction, in order to reverse the axis in the direction of the reference point switch.

On axis movements with referencing feed, a signal length of $\Delta t \geq 2 \cdot$ FM cycle must be assured for the reversing switch!

Example



The value for the reference-point shift (MD27) is traveled after the synchronization point is reached.

Without reference-point switch (RPS)

The following table describes how a reference can be recorded without a reference-point switch.

Recording of synchronization	Sequence of movements
R+, R– or Start	1. Instantaneous position is defined as reference point (reference-point coordinate). 2. Value for reference-point shift is traveled.

Note

Please see also Section 9.1.3!

Control actions

Preconditions:

- The FM 354 has been parameterized.
- The mode has been selected and confirmed
- Drive enable [AF] = 1 (control signal, FC MODE_WR)
- Stop [STP] = 0 (control signal, FC MODE_WR)
- Servo enable (RF) = 1 (FC MODE_WR, job no. 10)

Table 9-4 Control actions for “Reference point approach” mode (examples)

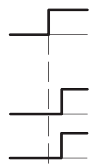
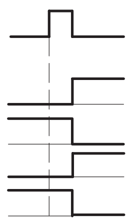
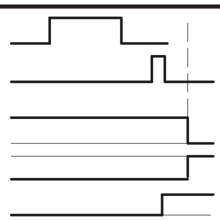
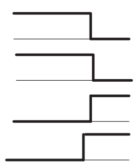
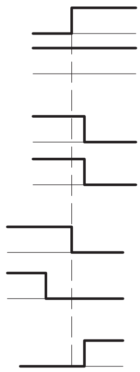
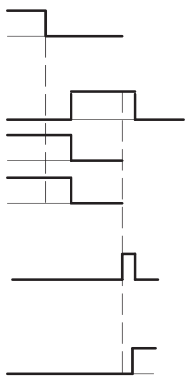
Signal name	Level	Explanation
Control action 1, enable “Reference point approach” mode		
Control signal: Mode [BA] **Checkback signals:** Active mode [BAR] Start enable [SFG]		The user initiates a [BA] command. The module returns [BAR] and [SFG].
Control action 2, move axis – positive direction		
Control signal: Direction plus [R+] **Checkback signals:** Travel plus [FR+] Start enable [SFG] Processing in progress [BL] Synchronization [SYN]		When [SFG] is active, [R+] or [Start] are actuated, for example. The axis cancels [SFG], outputs the [BL] and [FR+] messages and travels here in the positive direction (defined in MD). An existing synchronization is reset.
Control action 3, reference point switch (RPS) reached		
RPS Encoder zero marker **Checkback signals:** Travel plus [FR+] Travel minus [FR–] Synchronized [SYN]		When the RPS is reached, the velocity is reduced. The encoder is synchronized when the zero marker is detected. The axis is positioned by traversing through the reference point offset to the reference point (the direction is reversed if necessary).
Control action 4, approach reference point		
Checkback signals: Traverse minus [FR–] Position reached, stop [PEH] Processing in progress [BL] Start enable [SFG]		When reference point is reached. [FR–] is canceled. [PEH] is enabled. [BL] is also canceled. [SFG] is enabled.

Table 9-4 Control actions for “Reference point approach” mode (examples), continued

Signal name	Level	Explanation
Control action 5, ambiguous direction command (special situation)		
Control signals: Direction plus [R+] Direction minus [R–] **Checkback signals:** Travel minus [FR–] Processing in progress [BL] **Control signals:** Direction plus [R+] Direction minus [R–] **Checkback signal:** Start enable [SFG]		[R+] is defined although [R–] is active. The ambiguous direction command causes the axis to stop. [FR–] and [BL] are canceled, and an error is output. The [SFG] does not reappear until [R+] and [R–] have been canceled].
Control action 6, cancel servo enable (special situation)		
Single function “servo enable” (job no. 10) **Checkback signals:** Operator control/travel error [BF/FS] Travel minus [FR–] Processing in progress [BL] **Control signal:** Acknowledge operator control/travel error [BFQ/FSQ] **Checkback signals:** Start enable [SFG]		The “servo enable” is deactivated during the traversing movement. The axis is stopped abruptly and outputs an error. [FR–] and [BL] are canceled. When the error is acknowledged, the error message is canceled and the start enable is activated.

9.2.4 Incremental relative

Overview

In the Incremental Relative mode it is possible to execute single positionings over relative distances using user-definable increments.

The traversing movement is triggered with the direction keys (R+ and R–).

Defining the position

The options available for defining the increment with the mode parameter are:

- Via the user program,

  by defining the position for the increment **job no. 3**
- Using the increment table; see Section 5.3.2

Velocity level 1 **job no. 1** (see Section 9.2.1) is used as the velocity setpoint and can be modified during the movement.

It is **not** possible to change position on-the-fly (e.g. changing the position setpoint during a movement).

Handling by the user

The table below gives you an overview of how to handle this mode.

Triggering of movement, Direction (R)	Increment selection	Position, Distance to be traveled
R+ or R–	BP = 254	in accordance with setpoint for increment (job no. 3)
	BP = 1...100	as in SM table (DB-SM)

Position setting

Name	Lower input limit	Upper input limit	Unit
Increment	0	1 000 000 000	MSR

MSR stands for measurement system raster (see Section 5.3.1)

Note the following when interrupting a movement with “Stop”:

- To continue movement in the same direction - the residual distance is processed with the appropriate direction key.
- To continue movement with “delete residual distance” (job no. 11) the residual distance is deleted and the increment is traversed agai (provided the increment was not changed).
- To position in the opposite direction - the residual distance is deleted automatically.

Note

Please see also Section 9.1.3!

Control actions

Preconditions:

- The FM 354 has been parameterized.
- The mode has been selected and confirmed
- Drive enable [AF] = 1 (control signal, FC MODE_WR)
- Stop [STP] = 0 (control signal, FC MODE_WR)
- Servo enable (RF) = 1 (FC MODE_WR, job no. 10)
- Velocity levels have been transferred (FC MODE_WR, job no. 1)
- Axis is synchronized

Table 9-5 Control actions for “Incremental relative” mode (examples)

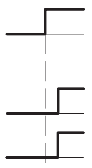
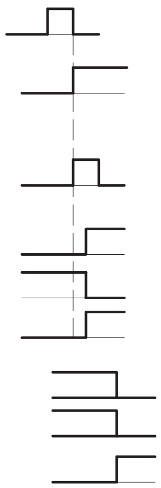
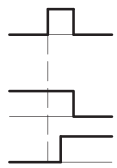
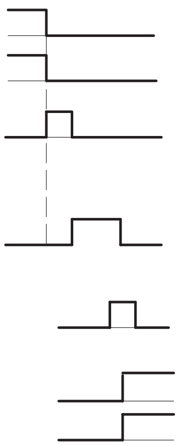
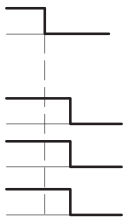
Signal name	Level	Explanation
Control action 1, enable “Incremental relative” mode		
Control signal: Mode [BA] **Checkback signals:** Active mode [BAR] Start enable [SFG]		The user initiates a [BA] command. The module returns [BAR] and [SFG].
Control action 2, define position		
Transfer increment (job no. 3) Select increment (254) **Control signal:** Direction plus [R+] **Checkback signals:** e.g. Travel plus [FR+] Start enable [SFG] Processing in progress [BL] e.g. Travel plus [FR+] Processing in progress [BL] Position reached, stop [PEH]		When the increment has been transferred and selected, [R+] can be initiated. The axis cancels the [SFG] and outputs messages [BL] and [FR+] When the defined position has been reached, the axis enables [PEH]; [SFG] and checkback signals [FR+] and [BL] are reset.
Control action 3, stop during positioning		

Table 9-5 Control actions for “Incremental relative” mode (examples), continued

Signal name	Level	Explanation
Control action 3, stop during positioning		
Control signal: Stop [STP] **Checkback signals:** Travel minus [FR–] Start enable [SFG]		If Stop is enabled during positioning, the axis stops. [FR–] is reset, and [SFG] is activated. [PEH] is not output, since positioning is not complete. Before the axis comes to a standstill, it is possible to define a new direction “through start”.
Control action 4, error during traversing movement		
Checkback signals: Travel plus [FR+] Processing in progress [BL] **Control signal:** Acknowledge operator control/travel error [BFQ/FSQ] **Checkback signals:** Start enable [SFG] **Control signal:** Direction plus [R+] **Checkback signals:** Travel plus [FR+] Processing in progress [BL]		The axis moves. An error is output during the traversing movement. [FR+] and [BL] are canceled, and [BFQ/FSQ] is enabled. When the error has been acknowledged, the start enable is activated. The movement can be restarted with [R+]. [FR+] and [BL] are activated. [SFG] is canceled.
Control action 5, change mode		
Control signal: Mode [BA] **Checkback signals:** Active mode [BAR] Travel plus [FR+] Processing in progress [BL]		[BA] is deactivated during the traversing movement. The axis is stopped by way of the deceleration ramp. [FR+] and [BL] are reset.

9.2.5 MDI (Manual Data Input)

Overview

In the MDI mode it is possible to execute single positionings by way of traversing blocks. These traversing blocks are provided by the user program.

The MDI block and MD block on-the-fly have an identical block structure.

MDI block

The structure of the MDI block is identical to the traversing program block (see Chapter 10, however it does not have a program number or block number).

The user program passes the "MDI block" (**job no. 6**) to the FM 354, and the block can then be executed. The block can be executed repeatedly, since it is stored internally. The feedrate is override-dependent.

The MDI block remains in effect until it is overwritten with a new MDI block. A new block can be transmitted while another block is being processed.

Table 9-6 MDI-Satz

Name	Lower input limit	Upper input limit	Unit
Position X / Dwell time t	–1,000,000,000 2	+ 1,000,000,000 100.000	MSR from MD7 ms
Speed F	10	500 000 000	MSR from MD7/min
G function group 1	G04 G90 G91	Dwell time Absolute measure Chain measure	–
G function group 2	G30 100% G31 10% to G39 90%	Override Acceleration/ Deceleration	–
M function group 1, 2, 3	M1...17 M19...96 M99 M97, 98 M2, M30	User functions Change signal programmed as digital output Not allowed	–

MSR stands for measurement system raster (see Section 5.3.1)

For rotary axes with absolute programming, the commands [R+], [R–] are defined as direction commands. They must be available before positioning starts.

MDI block on-the-fly

The MDI block currently being processed is canceled when the user program outputs an “MDI block on–the–fly” (**job no. 16**).

Transfer of “MDI block on–the–fly” interrupts the active “MDI block”. The new block is executed immediately without “Start”.

The MDI block on-the-fly is **not** saved in the FM 354.

Block structure

The following table shows the block structure of the MDI block.

X/t Position/dwell time programmed (fills in value 1)
G1...G2 G function group 1...2
M1...M3 M function group 1...3
F Speed programmed (fills in value 2)

Byte	Data format	Bit							
		7	6	5	4	3	2	1	0
0	Byte	0							
1	Byte	0							
2	8 bits	0	0	0	X/t	0	0	G2	G1
3	8 bits	0	0	0	0	M3	M2	M1	F
4	Byte	G function 1							
5	Byte	G function 2							
6	Byte	0							
7	Byte	0							
8	DINT	32-bit value 1							
12	DINT	32-bit value 2							
16	Byte	M function 1							
17	Byte	M function 2							
18	Byte	M function 3							
19	Byte	0							

Note:

When the assignment bit (byte 2 and byte 3) is not set, the associated values must be deleted.

Handling by the user

The table below gives you an overview of how to handle this mode.

Triggering of movement	Type of movement
Start	as defined by “MDI block” (job no. 6)
“MDI block on-the-fly” transmitted to the FM 354	as defined by “MDI block on–the–fly” (job no. 16)

Note

Please see also Section 9.1.3!

Control actions

Preconditions:

- The FM 354 has been parameterized.
- The mode has been selected and confirmed
- Drive enable [AF] = 1 (control signal, FC MODE_WR)
- Stop [STP] = 0 (control signal, FC MODE_WR)
- Servo enable (RF) = 1 (FC MODE_WR, job no. 10)
- Axis is synchronized

Table 9-7 Control actions for “MDI” mode (examples)

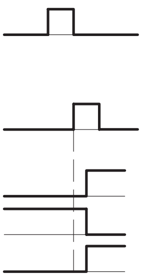
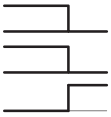
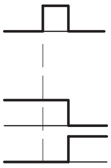
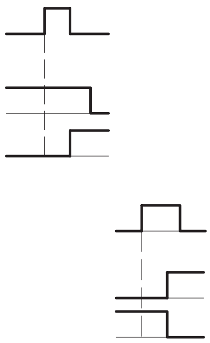
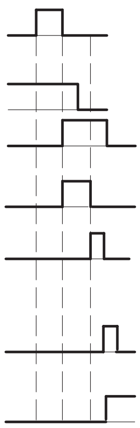
Signal name	Level	Explanation
Control action 1, define position		
Transfer MDI block (job no. 6) **Control signal:** Start [ST] **Checkback signals:** e.g. Travel plus [FR+] Start enable [SFG] Processing in progress [BL]		When the MDI block has been transferred, [ST] can be initiated. The axis cancels the [SFG] and outputs messages [BL] and [FR+]
e.g. Travel plus [FR+] Processing in progress [BL] Position reached, stop [PEH]		When the defined position has been reached, the axis enables [PEH]; [SFG] and checkback signals [FR+] and [BL] are reset.
Control action 2, change position during positioning		
Transfer MDI block on–the–fly (job no. 16) **Checkback signals:** Travel plus [FR+] Travel minus [FR–]		If a new “MDI block on–the–fly” is transferred during positioning, the current positioning operation is canceled immediately, and the new positioning operation is started on–the–fly. In this case, for example, this causes the direction to be changed from [FR+] to [FR–].

Table 9-7 Control actions for “MDI” mode (examples), continued

Signal name	Level	Explanation
Control action 3, stop during positioning with new start signal for resumed positioning		
Control signal: Stop [STP] **Checkback signals:** Travel minus [FR–] Start enable [SFG] **Control signal:** Start [ST] **Checkback signals:** Travel minus [FR–] Start enable [SFG]		If Stop is enabled during positioning, the axis stops. [FR–] is reset, and [SFG] is activated. [BL] remains active and [PEH] is not output, since positioning is not complete. If [ST] is initiated again, [FR–] and [SFG] are reset and positioning is completed. Before the axis comes to a standstill, it is possible to define a new direction “through start”.
Control action 4, stop during positioning with new start signal and new MDI block		
Control signal: Stop [STP] **Checkback signals:** Travel plus [FR+] Start enable [SFG] Transfer MDI block (job no. 6) Transfer “delete residual path” (job no. 11) **Control signal:** Start [ST] **Checkback signals:** Travel minus [FR–]		If Stop is enabled during positioning, the axis stops. [FR+] is reset, and [SFG] is activated. When a new MDI block has been transferred, [ST] is enabled again. “Delete residual path” is also enabled. The axis deletes the residual path of the old positioning operation, and starts executing the new traversing block. [FR–] is enabled, and [SFG] is reset. **Note:** If no new “MDI block” is transferred, execution of the current “MDI block” is repeated from the start. Without “delete residual path”, the interrupted positioning operation would be continued (see control action 3)

9.2.6 Automatic

Overview

In the Automatic mode (following-block mode), the FM 354 processes traversing programs autonomously. These programs are created with "Parameterize FM 354" (see Chapter 5, 5.3.4) and stored as a data block. The traversing programs contain information about movement sequences and outputs (see Chapter 10).

Program selection

Programs are selected (**job no. 17**) by way of the user program, by specifying a program number and an optional block number, as well as the direction of machining. A program can be selected only when other programs have been interrupted or terminated or at the start of a program.

A selected program remains active until it is inactivated by selecting program number = 0, or overwritten by selecting another program.

If modifications are made to a preselected program, including the subprogram, preselection of the program is canceled. You must then select the program again. A modification can be made to a program when BL = 0 (start of program/end of program) and on Stop.

Triggering of movement	Select program		Type of movement (according to programmed blocks)
	Block No.	Processing direction	
Start	0	forward	Start at beginning of program, process by ascending block number
	0	reverse	Start at end of program, process by descending block number
	e.g. 30	forward	Block search forward to block No. 30, by ascending block number
	e.g. 30	reverse	Block search in reverse to block No. 30, by descending block number
Start with automatic block search forward		forward	1. Automatic block search forward to interruption point 2. Positioning to interruption point (if a movement was performed in another mode) 3. Process the interrupted block and continue the program
Start with automatic block search in reverse		reverse	1. Automatic block search in reverse to interruption point 2. Positioning to interruption point (if a movement was performed in another mode) 3. Process the interrupted block and continue the program

User DB allocation

Data format	Significance
Byte 0	Program number
Byte 1	Block number
Byte 2	Direction of machining:0 = process forward 1 = process in reverse

Forward processing

The program processes the block numbers in ascending order.

Processing begins at Start, with the first block (specified block number = 0).

If processing is to begin at some other point of the traversing program, specify the desired block number. Processing will take place by searching forward to this block, then processing forward until the program end command is recognized.

Backward processing

The program processes the block numbers in descending order.

Processing begins at Start, with the last block (specified block number = 0).

If processing is to begin at some other point of the traversing program, specify the desired block number. Processing will take place by searching back to this block, then processing in reverse until the program beginning is recognized.

Note

If reverse processing is to execute the same sequence of movements as the forward movement, the effects of the corresponding commands must be taken into account in the programming. For example:

- M outputs should be written separately in a block; note M output (MD32) and G60/G64.
- Note change between G60/G64 and G90/G91.
- Note start and end of tool offsets.
- M18 is not executed.
- M02 and M30 at the end of the program are not processed.

Block search forward

The program is processed to the end point of the target block, including tool offset. M commands and dwell times are output and the traversing movements are suppressed.

When processing traversing programs with a forward block search, there are a number of special cases:

- The external forward block search (G50) is not executed.
- Continuous travel with functions to set (G88, 89) or delete (G87) an actual value on-the-fly is not executed.
- The blocks under G50, G87, G88, G89 (in the processing direction) should contain a path in absolute coordinates.

block search backward

Similar to block search forward

Automatic block search forward/ backward

Automatic block search forward/backward means that, after the interruption of an active automatic program (by an operating mode change), you can continue execution from this point of interruption in the appropriate direction of processing.

With forward block search, the interrupted program must previously have been going in the forward direction.

With block search in reverse, the interrupted program must previously have been going in the reverse direction.

The command for automatic forward or reverse block search is evaluated in the FM 354 at Start, and a forward or reverse search to the interruption point is executed. Positioning to the interruption point takes place (if positioning has taken place previously in some other mode), and then the interrupted block is processed, including any required output.

Control actions

Preconditions:

- The FM 354 has been parameterized.
- The mode has been selected and confirmed
- Drive enable [AF] = 1 (control signal, FC MODE_WR)
- Stop [STP] = 0 (control signal, FC MODE_WR)
- Servo enable (RF) = 1 (FC MODE_WR, job no. 10)
- Axis is synchronized

Table 9-8 Control actions for “Automatic” mode (examples)

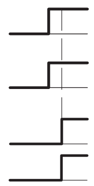
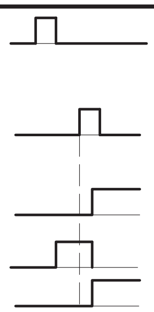
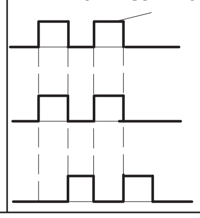
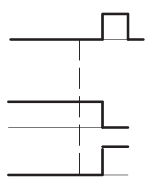
Signal name	Level	Explanation
Control action 1, Automatic/Automatic single block mode		
Control signals: Mode [BA] Read–in enable [EFG] **Checkback signals:** Active mode [BAR] Start enable [SFG]		The user initiates [BA] and [EFG]. The module returns [BAR] and [SFG].
Control action 2, positioning by program selection		
Program selection (job no. 17) **Control signal:** Start [ST] **Checkback signals:** Travel plus [FR+] or Travel minus [FR–] Start enable [SFG] Processing in progress [BL]		When [SFG] appears, the program can be activated by [ST] when [EFG] is active. Processing commences, e.g. with a positioning operation. [FR+] or [FR–] and [BL] are activated. [SFG] is reset.
Control action 3, M function output		
Checkback signals: Change M function [AMF] M function number [MNR] **Control signal:** Acknowledge M function [QMF]	2nd M command 	If M function output is acknowledgement–driven, for example, the user program can continue to process the [MNR] when [AMF] appears. M function output is complete. [QMF] acknowledges the M function, and [AMF] and [MNR] disappear.
Control action 4, M function output and positioning		
Control signal: Acknowledge M function [QMF] **Checkback signals:** Position reached, stop [PEH] Travel plus [FR+] or Travel minus [FR–]		Block with M output (same as control action 3) and position is started. The program is resumed on completion of the M function output. [FR+] and [FR–] are deactivated and [PEH] is reset.

Table 9-8 Control actions for “Automatic” mode (examples), continued

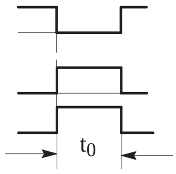
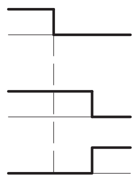
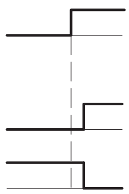
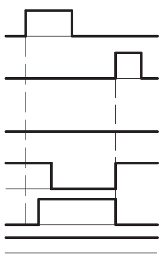
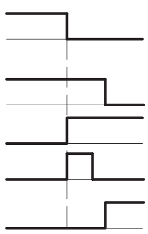
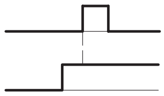
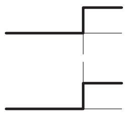
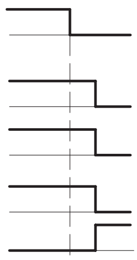
Signal name	Level	Explanation
Control action 5, traversing block with dwell		
Checkback signals: Travel plus [FR+] or Travel minus [FR–] Dwell time running [T-L] Position reached, stop [PEH]		During processing of a traversing block with dwell, the dwell time t_0 [T-L] and [PEH] are output.
Control action 6, cancelation of the read–in enable during program execution (special situation)		
Control signal: Read–in enable [EFG] **Checkback signals:** Travel plus [FR+] or Travel minus [FR–] Position reached, stop [PEH]		If [EFG] is canceled during program execution, the current block is processed up to the end, and program execution is then suspended. [FR+] and [FR–] are reset. [PEH] is actuated.
Control action 7, resume program execution after read–in enable (special situation)		
Control signal: Read–in enable [EFG] **Checkback signals:** Travel plus [FR+] or Travel minus [FR–] Position reached, stop [PEH]		The program resumes on [EFG]. [FR+] and [FR–] are reset. [PEH] is reset.
Control action 8, stop during positioning with new start signal for resumed positioning (special situation)		
Control signals: Stop [STP] Start [ST] **Checkback signals:** Position reached, stop [PEH] Travel plus [FR+] Start enable [SFG] Processing in progress [BL]		Interrupt with Stop [FR+] is cleared when the axis comes to a standstill, and [SFG] is enabled (if Stop is not active). [PEH] remains cleared, since the defined position has not yet been reached. Start clears [SFG] and enables [FR+] again. [BL] remains enabled. Before the axis comes to a standstill, it is possible to restart “through start”.
Control action 9, end of program reached		
Checkback signals: Travel plus [FR+] or Travel minus [FR–] Processing in progress [BL] Position reached, stop [PEH] M function number [MNR] Start enable [SFG]		The end of the program is indicated by the enabling of [PEH], output of M2, M30 and resetting of [BL].

Table 9-8 Control actions for “Automatic” mode (examples), continued

Signal name	Level	Explanation
Control action 10, delete start signal and residual path (special situation)		
Control signal: Start [ST] Transfer “delete residual path” (job no. 11)		If “delete residual path” is also preselected on [ST], the block interrupted by Stop is not executed up to the end, but the next block is started immediately.
Control action 11, positioning for rotary axis (special situation)		
Control signals: Direction plus [R+] or direction minus [R–] Start [ST]		If the axis is operated as a rotary axis, the FM always attempts to select the shortest path during positioning. This direction preference can be suppressed by specifying [R+] or [R–].
Control action 12, deactivate operating mode during program execution (special situation)		
Control signal: Mode [BA] **Checkback signals:** Old mode [BAR] Travel plus [FR+] or Travel minus [FR–] Processing in progress [BL] New mode [BAR]		If a new operating mode is selected during active program execution, the axis is stopped by way of the deceleration ramp. [FR+] or [FR–] and [BL] are reset.

9.2.7 Automatic single block

Overview

Functions, same as “Automatic” mode

Whereas in “Automatic” mode the FM 354 automatically starts processing the next block after completing a given block, in “Automatic single-block” mode the axis waits for a new Start signal after processing each block that contains a traversing path, dwell time or M command (except for blocks with G50, G88 or G89).

You can change between Automatic single-block and Automatic mode at any time, without stopping the movement or interrupting the output.

9.3 System data

Overview

This chapter describes settings and functions that apply in multiple modes, and that are likewise necessary in order to control and operate the FM 354, and data of the FM available for checkback messages.

These settings/functions, which you can activate by calling FC 2 or FC 3 (see Section 6) with the appropriate job no., are listed in the table below.

Before you call FC 2 with the job no., the corresponding values must be entered in the user DB.

Chapter overview

9.3.1 Change parameters/data (job no. 8)

Overview

You can use this function to change parameters and data in the data function blocks of the FM 354, or to issue an order to read parameters or data from data function blocks. The parameters/data can then be read out with function call FC RD_COM job no. 114 (see Section 9.3.17).

Structure of data record

The following table shows which parameters or data can be changed or read by setting the indicated codes.

Addr. in user DB	Data format	Symbol	Description				
126	Byte	DB type	type	1 = MD	2 = SM	3 = TO	4 = NC (traversing program)
127	Byte	data number	Info 1	MD No. (5...45)	SM No. (1...100)	TO No. (1...20)	Progr. No. (1...199)
128	Byte	number of data	Info 2	Number of MDs, consecutive (1...5)	Number of SMs, consecutive (1...5)	0 = Tool offset complete 1 = Tool length only 2 = Wear value abs. only 3 = Wear value add. only	Block No. (1...255)1)
129	Byte	job type	1 = Read job parameters 2 = Write parameters 4 = Write parameters and save				
130...149	depends on type	data array	Parameters/data (see data blocks, Section 5.3)				

1) For block format see Section 9.3.12 “Active NC block”

Example

The software limit switches (MD21, MD22) for the axis are to be set to the values 100 mm and 50,000 mm. These values are to remain in effect only until the unit is shut down.

DB type = 1
data number = 21
number of data = 2
job type = 2
data array
  Byte 5...8 = 100,000 (MD21)
  Byte 9...12 = 50,000,000 (MD22)
  Byte 13...24 = 0

For activation of the machine data, see Section 9.3.3

Notes

Please note the following when changing the parameter data:

- **Machine data**

  Machine data can always be modified. Once you have modified the machine data, the machine data have to be reactivated (for single command, see Section 9.3.3).

- **Increments**

  Modifications can be made in all operating modes (even in “Incremental relative” mode) during movement. The modifications of the increments must always be complete before a new movement is started in “Incremental relative” mode. If this is not the case, the error message “incremental dimensions do not exist” is output Cl. 2/No. 13.

- **Tool offset data**

  Modifications can be made in all operating modes and during movement. If modifications are made during starting or at block transitions when the tool compensation is active (internal access to offset values), the error message “tool offset value does not exist” is output Cl.3/No.35.

- **Traversing programs**

  – Programs which are not selected can always be modified.

  – If modifications are made to a preselected program, including the subprogram, preselection of the program is canceled. You must then select the program again. A modification can be made to a program when BL = 0 (start of program/end of program) and on Stop.

Delete block: Specify the program no. and the block no. in the “data field”. The other data/bits must not be assigned.

Insert block: The block number does not exist in the selected program. The contents should be entered in accordance with the “block format”.

Modify block: The block with the corresponding block number is overwritten with the contents in accordance with “block format”.

Retentive storage of parameter data

Please note the following when using the function “Write parameters with retentive storage” (byte 4, job type 4):

Retentive writing must only occur on demand (not cyclically)!

Retentive data are stored on FEPROM (maintenance–free, no battery required). This memory has a physical limit for the possible number of delete/reprogram cycles: minimum 10^5, typically 10^6. The possible number of delete/reprogram cycles can be multiplied, from the user’s viewpoint, by providing a larger retentive memory capacity (much larger than the parameter data memory) and organizing the memory accordingly.

$$\text{Number of delete/reprogram cycles} = \frac{64\,000 \cdot 10^6 \text{ (typical)}}{\text{Block size (in bytes), in which parameter data are modified}}$$

Block sizes:

DB Machine data	approx. 250 bytes
DB Increments	approx. 460 bytes
DB Tool offset data	approx. 310 bytes
DB Traversing programs	110 + (20 x no. of traversing blocks) bytes

Example:

Assuming a service life of 10 years and 24–hour operation, a typical limit = 10^6.

Parameterization data	DB size	Number of possible delete/reprogram cycles	Number of possible delete/reprogram cycles per minute
MD	250 bytes	$256 \cdot 10^6$	49
Traversing programs (20 blocks)	510 bytes	$125.49 \cdot 10^6$	24

Note

SDB $\geq$ 1 000 (system data block, created for module replacement), contains parameter data which were valid at the time of start–up. If data/parameters are modified during operation and stored retentively on the FM, these data are not contained in SDB $\geq$ 1 000. These modifications are lost when the module is replaced, and should be traceable in the user program.

9.3.2 Single functions (job no. 10)

Overview

You can use this function to transfer single settings to the FM 354 and activate the corresponding functions. These settings are:

- Length measurement
- Inprocess measurement
- Retrigger reference point
- Switch off enable input
- Follow-up mode
- Switch off software end position monitoring
- Switch off automatic drift compensation
- Servo enable
- Parking axis
- Simulation

Callup of single settings

The individual functions remain activated until they are reset.

Length measurement, inprocess measurement

Since both functions use the same digital input on the FM 354, only one function can be executed at a time. In double activations, both functions are switched to inactive. An error message is issued.

For function description, see Section 9.3.10

Retrigger reference point

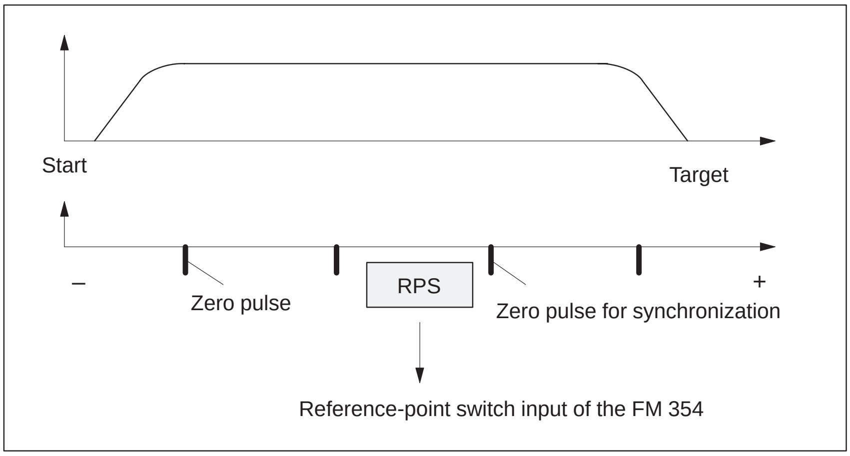
A precondition for retrigger reference point is that the axis has been synchronized by reference point approach.

With this setting, the axis is synchronized at each positive edge of the zero mark, after leaving the reference-point switch (RPS) in the direction of the zero mark (direction as in Reference-Point Approach). Regardless of the instantaneous speed, at this moment the reference-point coordinate is associated with the current actual position, taking any active shift into account.

The resulting change in the actual value causes no internal changes in the target.

When crossing the RPS, a signal length of $\Delta t \geq 2 \cdot$ FM cycle must be assured!

When a Set Actual Value On-the-Fly is pending, activation of Retrigger Reference Point is interlocked.



Hint to the user:

You can use Retrigger Reference Point, for example, to compensate for slippage of the trolley in a high-bay warehouse during operation, without having to resynchronize the axis with the Reference-Point Approach mode. When retriggering in reference point approach with a zero pulse, be careful that the total slippage between the reference point switch and the stepper motor does not become so great that the synchronizing zero pulse migrates to an “adjacent” zero pulse!

Switch off enable input

With the “switch off enable input” function, you can switch off evaluation of the enable input (see Section 9.8.1).

Follow–up mode

The “follow–up mode” function is used to cancel closed–loop control of the axis.

- For external movement of the axis, the actual value is tracked.
- This setting can be switched on or off only if “Processing in progress” = 0.

Deactivate end position monitoring

You can use this function to deactivate monitoring of the software limit switches (see Section 9.9).

It can be switched on or off only if “Processing in progress” = 0.

Deactivate automatic drift compensation

This function can be used to switch off the automatic drift compensation.

Automatic drift compensation means:

The drift is balanced to zero by an automatic matching of the analog actuating signal.

- The setting can be switched on or off if the axis is not in motion.
- Automatic drift compensation has no effect:
  - in Control mode
  - in the Follow-up Mode setting
  - if there is no servo enable
  - in the absence of a controller ready signal (if parameterized)
  - if the axis is in motion.

Servo enable

You can use this function to:

- activate **position control** (the prerequisite for closed-loop-controlled operation of the FM 354)
- to switch the signal through to the drive as indicated in MD37
- to provide control of stops between movements in the Control mode.

Parking axis

This function can be used to change over the measurement system while the complete system is running.

With this setting:

- Encoder synchronization (SYN = 0) is deleted
- Pending error messages are deleted and no new ones are triggered (including diagnostic interrupts)
- dig. outputs are inactive; analog voltage 0 V.

The setting can be switched on or off if "Processing in progress" = 0.

Simulation

You can use this function to:

- Test function sequences without the drive and measuring system.
- Evaluate all digital inputs (**Caution**, if you are going to simulate sequences that use such signals, they should be connected to the inputs of the FM 354 - e.g. for Reference-Point Approach).
- The servo simulates a controlled system; "Controller Ready" is not necessary.
- The setting can be switched on or off if "Processing in progress" = 0.
- All internal function sequences behave as in normal operation.

When the function is deactivated, the axis is reset internally (see Restart, Section 9.3.3).

9.3.3 Single commands (job no. 11)

Overview

You can use this function to transfer single commands to the FM 354. These commands are:

- Activate machine data
- Delete residual distance
- Automatic block search in reverse
- Automatic block search forward
- Restart
- Undo set actual value

Callup of single commands

The single commands are activated when the corresponding data record is transmitted to the FM 354.

The commands are deleted in the FM 354 after execution.

Activate machine data

Once you have downloaded the machine data (MD) or the MD block (from the programming device), the machine data have to be activated. At the first parameterization, the machine data is transferred automatically. In terms of effects, the FM 354 distinguishes between "K" and "E" machine data.

MD category	Effect in FM 354 after activation
"K"	"Reset" of the FM • As long as "Reset" is in progress, it is not possible to transfer other data • For internal response, see Restart
"E"	FM operating condition is maintained

For machine data, see Section 5.3.1.

This command is possible only when the axis is not in motion ("Processing in progress" = 0).

An MD block is likewise activated by switching on or off.

Delete residual path

You can use this command to delete a residual distance that remains after a job has been canceled.

- It is effective only in the “Incremental Relative”, “MDI”, and “Automatic” modes after a stop. If processing is not interrupted with a stop, the Delete Residual Distance requirement is suspended in the FM 354.
- On starting after a Delete Residual Distance in MDI mode, the active MDI block is processed from the start.
- On starting after a Delete Residual Distance in Incremental Relative and Automatic modes, processing continues with the following block.

Automatic block search forward

This command is described in Section 9.2.6.

Automatic block search backward

This command is described in Section 9.2.6.

Restart

You can use this command to reset the axis.

- The setpoint output is interrupted.
- The instantaneous processing status is canceled, and synchronization in incremental encoders is deleted.
- Active compensation values are deleted.
- An acknowledge signal is issued for all errors.

Undo set actual value

You can use this command to reset coordinates modified with the functions ”Set actual value” and “Set actual value on–the–fly” to their original value (if the axis is not in motion).

9.3.4 Zero offset (job no. 12)

Overview

You can use this function to shift the current zero point.

Function of zero offset

A selection, change or cancellation of a zero offset takes effect with the next positioning action. With a zero offset, the instantaneous shift of a coordinate system is canceled, provided that a zero offset was already active and the specified shift was executed (relatively). All coordinates and software limit switches, the reference point and the actual value are updated accordingly.

Example of a zero offset:

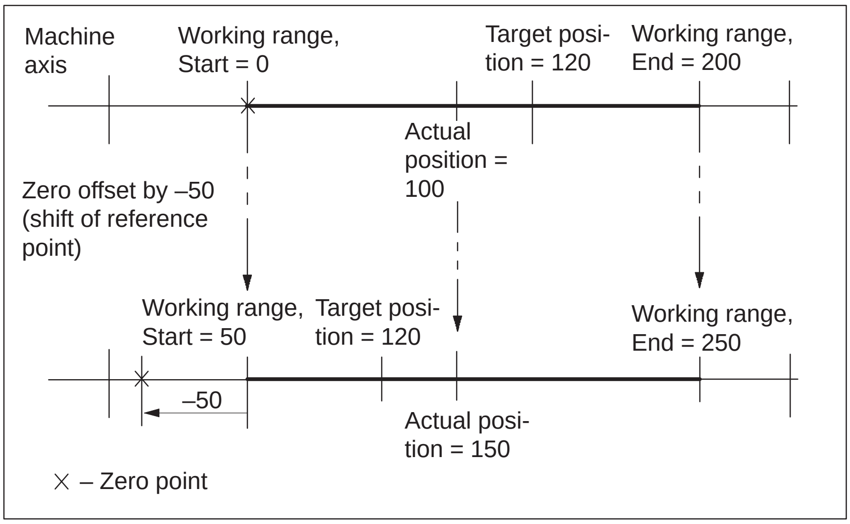


Fig. 9-1 Zero offset

The zero offset can be deleted by:

- Transmitting shift value = 0
- Starting Reference-Point Approach mode
- Set reference point
- Eliminating axis synchronization (e.g. by a restart).

Rotary axis

The following restriction applies to a rotary axis:

Zero offset < Rotary-axis range. The actual value is normalized.

Example:

The start and end of the range are shifted –50°.

Exceptions:

Int he “Incremental Relative”, “MDI” and “Automatic” modes, a zero offset is not possible until the block has been processed (position reached, programmed stop set), i.e., it is not possible when the axis is stationary after execution is interrupted with an abnormal stop.

9.3.5 Set actual value (job no. 13)

Overview

You can use this function to assign a new value to the current actual value.

Function of set actual value

By transmitting the coordinates, the actual value is set to this value when the axis is not in motion (after selecting “Processing in progress” = 0). The coordinates of the software limit switches remain unchanged.

Example of setting actual value:

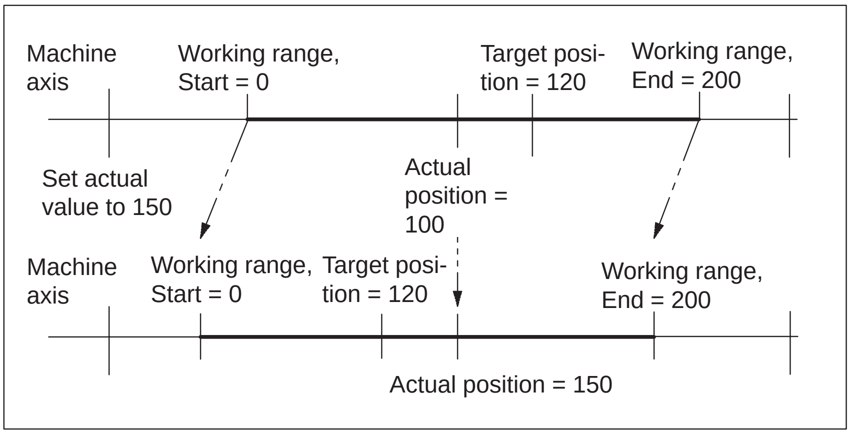


Fig. 9-2 Set actual value

The coordinates can be reset to their original value by:

- Including synchronization in Reference Point Approach mode
- Set reference point
- Reset actual value
- Restart

9.3.6 Set actual value on–the–fly (job no. 14)

Overview

You can use this function to assign a new value to the actual value by means of an external event.

Function of set actual value on-the-fly

By transmitting the coordinates (new actual value), set actual value on-the-fly is activated.

However, the Set Actual Value function is not triggered via the appropriate digital input until “Processing in progress” = 1.

Set actual value on-the-fly can be activated again by transmitting Set actual value on-the-fly again.

The coordinates can be reset to their original value by:

- Including synchronization in Reference Point Approach mode
- Set reference point
- Reset actual value
- Restart

Note:

For “set actual value on–the–fly” in “Automatic” mode, see Section 9.2.6

9.3.7 Request application data (job no. 18)

Overview

A selection of up to four display data items whose values can be read out with “read application data” (see Section 9.3.13).

Code table:

Code	Significance
0	No parameter request
1	Actual position
2	Actual velocity
3	Distance to go
4	Setpoint
5	Total current coordinate shift
6	Speed
16	DAC output value
17	Actual encoder value
18	Pulse errors
19	K_v factor
20	Following error
21	Following error limit
22	s Overshoot/Switch readjustment in Reference Point Approach mode
23	Approach time Te (ms)/drive constant Ta (ms) in Control mode

The code should be entered in CODE_AP1...AP4.

These values are always updated in the module cycle.

The selection is stored on the FM, i.e. you only need to make the selection once and read the correponding values cyclically (job no. 105).

9.3.8 Teach in (job no. 19)

Overview

In a program block selected with the program number and block number, the current actual position is entered as a position setpoint (**Caution:** This is an absolute position).

The Teach-in facility is possible only in the following modes:

- Jogging
- Incremental Relative
- MDI

and while the axis is not in motion, when "Processing in progress" = 0.

The program and the appropriate program block must be present on the FM 354 (see Parameterization, Chapter 5).

9.3.9 Set reference point (job no. 21)

Overview

You can use this function to synchronize the axis without reference point approach.

Function

With Set Reference Point, a position value at the instantaneous position of the axis, indicated as a parameter, is accepted as an actual value.

For axes with an absolute encoder, the generated position reference is entered in MD17. At a known axis position, the known actual position of the system of measurement is transmitted to the FM 354 with Set Reference Point. This value is set as the actual value of the axis. At the same time, this position reference is saved, in that the assignment of the encoder actual value to the axis reference point is calculated from the assignment of the set actual position to the encoder actual value at this point of the axis; it is then entered in MD17.

9.3.10 Measured values

Activating the measurement function

A “length measurement” or “inprocess measurement” can be activated by calling **FC 2** and **job no. 10** “single functions”.

Since both functions use the same digital input of the FM 354, only one function can be executed at a time. In double activations, both functions are switched to inactive. An error message is issued.

Reading out the measurement function

You can read out measured values from the FM 354, in the results from the execution of the “length measurement” and “inprocess measurement” functions, by calling **FC 5**.

Prerequisites

The following prerequisites must be fulfilled in order to execute the function “measurement”:

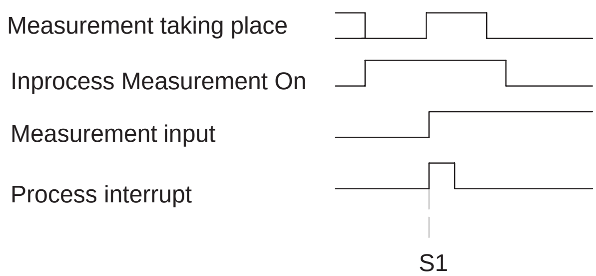
1. Connect a bounce-free switching-signal encoder (touch probe) to a digital output of the FM 354.
2. Parameterize Measurement for this input in MD34.

Function description

Measurement functions can be executed in all modes. An executed measurement is signaled by the checkback signal “ME” and optionally also by a process interrupt.

Inprocess measurement

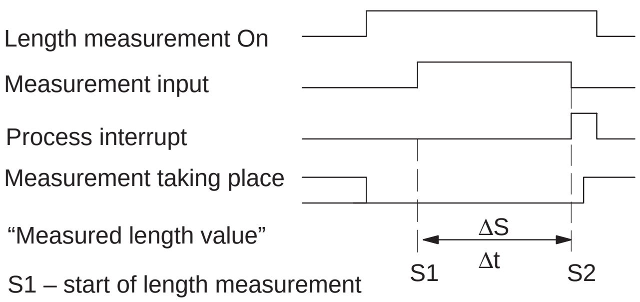
The present actual position is captured at each rising edge of the touch probe. At the same time, the axis movement is interrupted (servo-controlled braking).



S1 – execution of measurement

Length measurement

The present actual position is captured at both the rising and the subsequent falling edge of the touch probe. In addition, the actually traveled distance (amount) is calculated.



S1 – start of length measurement
S2 – end of length measurement
ΔS – Measured length value
Δt – minimum signal length at the digital input: $\geq 2 \cdot$ FM cycle

The following response takes place with functions that change the current actual value:

- Resynchronization in Reference-Point Approach mode:measured length contains measurement-error message
- Set Reference Point:measured length contains measurement-error message
- Retrigger Reference Point:measured length is difference between edge positions
- Set Actual Value:measured length is actual traversed distance

Note

The zero offset function does not change the actual position of the axis, and is thus not relevant for the above observations regarding the Length Measurement function.

Error messages

The following table lists the errors that can occur in the execution of the Measurement function.

Error	Significance
Travel errors	The “Digital input not parameterized” error is signaled when a measurement function is selected without a digital input having been parameterized (see Troubleshooting, Table 11-7, Class 3 No. 30).
Operator control errors	The “Measurement function undefined” error is signaled when both measuring functions are selected (see Troubleshooting, Table 11-6, Class 2 No. 16).
Measuring error	An erroneous length measurement is indicated by signaling back the length “–1.” Possible causes include: • Resynchronization in the Reference-Point Approach mode while a measurement is in progress • Execution of the Set Reference Point function while a measurement is in progress • Direction of travel at the falling edge is opposite to the direction of travel of the previous rising edge

Measurement checkback signals

The checkback signal ME (see Section 9.1) signals the status of function execution, as follows:

"ME"	Inprocess measurement	Length measurement
0	• the Length Measurement and Inprocess Measurement functions are inactive • with Start after a prior measurement	• the Length Measurement and Inprocess Measurement functions are inactive • with front edge of the touch probe signal after a prior measurement
1	With the front edge of the touch probe signal (= Inprocess Measurement is in progress)	With the back edge of the touch probe signal (= Length Measurement is in progress)

In association with the ME checkback signal, the read-out measured values are valid for the executed measurement process.

No.	Value 0	Value "–1"	All other positive values	All other negative values
1	the Length Measurement and Inprocess Measurement functions are inactive	Actual position for rising touch-probe edge in Inprocess Measurement and Length Measurement functions		
2	• the Length Measurement and Inprocess Measurement functions are inactive • always with the Inprocess Measurement function	Actual position for falling touch-probe edge in Length Measurement function		
3	• the Length Measurement and Inprocess Measurement functions are inactive • always with the Inprocess Measurement function • measured length 0 is actually possible, because touch probe has been connected while axis is stationary	Erroneous length measurement	Measured length	Nonexistent

9.3.11 Basic operating data (job no. 102)

Overview

The following display data are basic operating data:

- Actual position (MSR)
- Actual speed (MSR/min)
- Residual distance (MSR)
- Set position (MSR)
- Total of active coordinate shifts for tool offset, zero offset (MSR)
- Rotational speed (rotary axis only) (rpm)

9.3.12 Active NC block (job no. 103), next NC block (job no. 104)

Active NC block ... are display data in “Automatic” mode

/	Skipped block
L	Subprogram callup (fills in UP number)
P	Number of callups for subprogram (fills in UP callup number)
X/t	Position/dwell time programmed (fills in value 1)
G1-G3	G function group 1-3
D	Tool offset value number
M1-M3	M function group 1-3
F	Speed programmed (fills in value 2)

Byte	Data format	Bit							
		7	6	5	4	3	2	1	0
0	Byte	NC program number							
1	Byte	NC block number							
2	8-bit	/	L	P	X/t	0	G3	G2	G1
3	8-bit	0	0	0	D	M3	M2	M1	F
4	Byte	G function 1							
5	Byte	G function 2							
6	Byte	G function 3							
7	Byte	0							
8	DINT	32-bit value 1 (UP number, bytes)							
12	DINT	32-bit value 2 (UP callup number, bytes)							
16	Byte	M function 1							
17	Byte	M function 2							
18	Byte	M function 2							
19	Byte	D function							

Next NC block as described in “active NC block”

9.3.13 Application data (job no. 105)

Overview

The values passed with “request application data” **job no. 18** (see Section 9.3.7) are returned from the FM 354.

9.3.14 Actual value block change (job no. 107)

Overview

The “actual value block change” function is described in Section 10.1, G50, G88, G89.

9.3.15 Servicing data (job no. 108)

Overview

The following display data of the measuring circuit are servicing data:

- DAC output value [mV]
- Actual encoder value [MSR]
  - Incremental encoder:
    (0...2^{16} – 1) [encoder increments]
  - Absolute encoder: Absolut value + encoder
- Missing pulses (pulses)
- K_v factor (position control loop gain)
- Following error [MSR]
- Following-error limit (MSR)
- s overshoot value/switch alignment in “Reference point approach” mode [MSR]
- Approach time Te [ms]/drive time constant Ta [ms] in “Open–loop control” mode

9.3.16 Additional operating data (job no. 110)

Overview

The following display data are additional operating data:

- Override (%)
- NC traverse program no.
- NC block no.
- Callup SR loop no. counter
- G90/91 active, see Section 10.1
- G60/64 active, see Section 10.1
- G43/44 active, see Section 10.1
- D No. active, see Section 10.1
- Status messages 1 (data type: BOOL):
  - Speed limitation to limit value from MD23
  - Limitation to
    $\geqslant$ 10 V
  - Limitation of minimum acceleration or minimum deceleration in effect
- Status messages 2 (data type: BOOL): not in use

9.3.17 Parameters/data (job no. 114)

Overview

The parameters and data requested for reading with the “modify parameters/data” function **job no. 8** (see Section 9.3.1) can be read.

9.4 System of measurement

Overview

At the start of parameterization, you must fill in the basic machine data item **system of measurement** (MD7). This item governs the input of values.

Variants of the system of measurement

You can set the system of measurement for the following three units:

- mm
- inches
- degrees

Input of machine data

All value inputs and all value ranges refer to the setting in the system of measurement.

Internal processing of values

In “Parameterize FM 354” and in the FM 354 itself, values are processed in the following base units:

- 0.001 0.001 mm
- 0.0001 inch
- 0.0001 degree

Examples

The sample values in the table below illustrate the relation between the system of measurement and internal values:

System of measurement	Internal values	Input at interface	
mm	10^{-3} mm	10 995 · 10^{-3} mm	10.995 mm
inches	10^{-4} inch	10,995 · 10^{-4} inch	1.0995 inch
degrees	10^{-4} degree 10^{-2} degree	3,600,000 · 10^{-4} degree 36,000 · 10^{-2} degree	360.0000 degree 360.00 degree

---

Note

The measurement system (MD7) must match the measurement system specified in the other DBs.

The measurement system raster (MSR) is the smallest distance unit in the active system of measurement.

If at some point you have failed to take this precaution:

1. Delete all data blocks (which do not match the measurement system) or clear the memory of the FM 354 completely.
2. Modify the other data blocks on the PG.
3. Reload the data blocks to the FM 354.

9.5 Axis type

Overview

You can select the axis type with machine data item MD8. Choose either of the following types:

- Linear axis
- Rotary axis

Linear axis

A linear axis moves between two range limits (traversing range min -10^9, max 10^9). The range limits may be bounded by software limit switches (MD21, MD22) to form the "working range."

Linear axes have a limited traversing range. The limits are imposed by the:

- Resolution of the numerical scale
- The range covered by an absolute encoder.

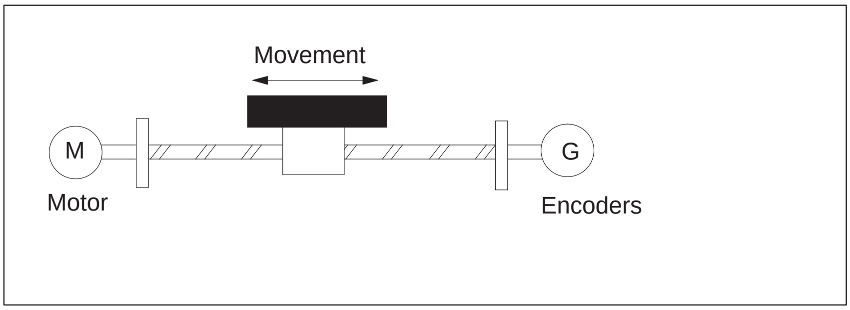


Fig. 9-3 Linear axis

Rotary axis

With rotary axes, the actual value is reset to "0" after one revolution. Rotary axes thus have an infinite traversing range.

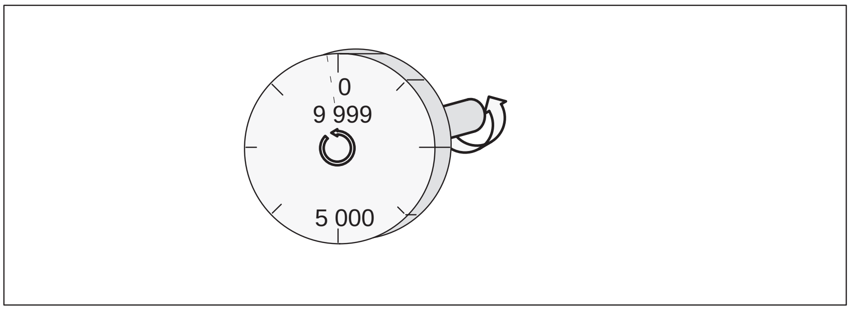


Fig. 9-4 Rotary axis

Rotary axis end

Machine data item MD9 defines the value by which the FM 354 recognizes the end of the rotary axis.

This value is the theoretical maximum that the actual value can reach. At this value, display of the actual value switches back to 0.

The theoretical maximum, however, is never displayed, because it is physically located in the same position as the start of the rotary axis (i.e. 0).

Example:

The following example in Figure 9-4 illustrates the behavior of the axis.

Assume you specify a value of 10,000 for the end of the rotary axis.

The value 10,000 will never be displayed. The display always rolls over from 9,999 to 0.

If the direction of rotation is negative, the display rolls over from 0 to 9,999.

Encoders on rotary axes

Rotary axes are subject to certain restrictions in the choice of encoder/gearbox/motor, as shown in Figure 9-5. These restrictions arise from the need to reproduce the actual position accurately over several revolutions when referencing (with incremental encoders or on POWER OFF/ON with absolute encoders). See also “Dependencies”, Section 5.3.1).

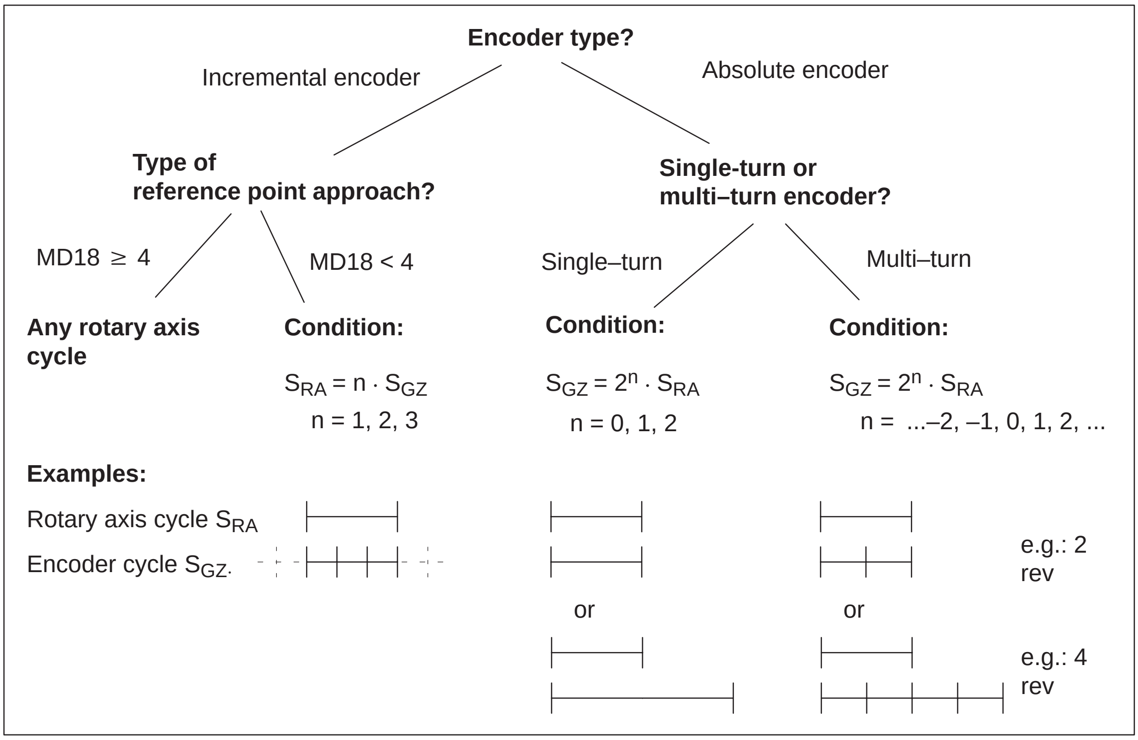


Fig. 9-5 Encoders on rotary axes

9.6 Encoders

Overview

One of the following encoders must be connected to the measuring-system interface of the FM 354 (see Fig. 1-4).

- Incremental encoder
- Absolute encoder (SSI)

Paths and positions are displayed in 10^{-3} mm, 10^{-4} inches or 10^{-4} degrees, as selected with machine data item MD7.

The path resolution of the machine axis obtained by the encoder is calculated within the FM from machine data MD11 to MD13.

Encoder selection

The prerequisite for achieving a given positioning accuracy is an n-fold improvement in path resolution by the encoder.

Recommended values for n		
Minimum	**Optimum**	**Maximum**
2	4	10

For that reason, when configuring a given specific application, select an encoder that meets the desired positioning accuracy requirements.

The known design data for the machine axis and the desired measurement resolution A:

$$A = \frac{1}{n} \cdot \text{Positioning accuracy} \qquad \text{(mm), (inches), (degrees)}$$

yield a calculation of the necessary pulse number per encoder revolution according to the following relationship (taking a metric measuring system as an example):

Incremental encoder	Absolute encoder (SSI)
$I_G = \dfrac{S\ (\text{mm})}{4 \cdot i_{GS} \cdot A\ (\text{mm})}$	$S_G = \dfrac{S\ (\text{mm})}{i_{GS} \cdot A\ (\text{mm})}$

The table below gives you an overview of the data used in this calculation and their meaning. You will find the machine data (MD) assignments under “Function parameters”.

Symbol	Significance
I_G	Increments per encoder rotation (incremental encoder)
S_G	Number of steps per encoder revolution (absolute encoder)
S	Distance per spindle or rotary table revolution [mm/rev], [inches/rev], [degrees/rev]
A	Required resolution [mm], [inches], [degrees]
4	Pulse multiplication (constant)
i_{GS}	Ratio between encoder and mechanism - Number of encoder revolutions $\left[\frac{\text{number of encoder revs}}{\text{spindle revolution}}\right]$ or $\left[\frac{\text{number of encoder revs}}{\text{or rotary table revs}}\right]$

If unusual numbers of pulses or steps result, the encoder with the next-higher number of pulses or steps should be selected.

9.6.1 Incremental encoders

Overview

Incremental encoders serve to detect position values, supplying pulses that the FM 354 adds up to form an absolute value. After the FM 354 is switched on, there is an offset, which cannot be determined in advance, between the internal position value and the mechanical position of the axis. In order to establish the position reference, the internal value must therefore be set to a predefined value at a specific axis position. This value is stored in the machine data (MD) as a reference point coordinate (see Section 9.2.3).

Incremental encoders

The following variant applications are possible:

- **Rotary incremental encoder on linear axes**

  Encoders with one zero pulse per revolution may be used. The number of encoder pulses must be a multiple of ten or a power of two.

- **Rotary incremental encoder on rotary axes**

  Encoders with one zero pulse per revolution may be used. The number of encoder pulses must be a multiple of ten or a power of two. With indirect encoder mounting and reference point approach with a zero pulse (MD18 < 4), you must ensure that the revolution of the rotary axis is divisible without remainder by the cyclical zero pulse (see “Dependencies” Sections 5.3.1. and 9.5).

- **Linear scales on linear axes**

  Scales may be used with at least one reference zero pulse, or with a cyclic zero pulse.

  In comparison to rotary incremental encoders, instead of the encoder revolution a period of division is used as a basis here, corresponding for example to the segment between two zero-mark pulses.

Function parameters

Table 9-9 shows you how to adapt the selected encoder to the FM 354.

Table 9-9 Function parameters – Incremental encoders

MD	Designation	Value/Meaning	Comments/Unit
10	Encoder type	**1 = Incremental encoder**	(Code number)
11	Displacement per encoder revolution (division period)	1...1 000 000 000 see Section 5.3.1, Dependencies	(MSR) (integer portion)
12	Residual distance per encoder revolution (division period)	$0...2^{32}–1$ see Section 5.3.1, Dependencies	$(2^{–32}$MSR) (fractional portion)
13	Increments per encoder revolution (division period)	$2^1...2^{25}$ see Section 5.3.1, Dependencies	Entry according to encoder rating plate
19.0	Direction adjustment	**1 = invert measured value direction**	–
20 20.0 20.2 20.3	Hardware monitoring	 **1 = Cable break** **1 = Pulse monitoring** **1 = Voltage monitoring**	Entry for monitoring to be switched on

MSR stands for measurement system raster (see Section 5.3.1)

Sample encoder adjustment

Encoder: Number of increments per revolution (MD13) = 2,500

(The FM 354 works by the principle of quadruple evaluation. This yields an FM-internal number of increments per revolution = 10,000.)

Machine design:

- Motor with 50:30 gear ratio on spindle with 10 mm pitch = 10,000 MSR
- Encoder on motor.

From this one can calculate the following traversing distance per encoder revolution:

$$\text{Gear ratio:} \quad i = \frac{\text{50 spindle revolutions}}{\text{30 motor revolutions}} = 1.666666...$$

$$\text{Displacement per encoder revolution} = i \cdot 10{,}000\ \text{MSR} = 16{,}666.666...\ \text{MSR}$$

The following values are entered:

MD	Value	Unit
11	16 666	$(10^{–3}$ mm)
12	$0.666... \cdot 2^{32} = 2{,}863{,}311{,}530$	$(2^{–32} \cdot 10^{–3}$ mm)
13	2 500	(puls/rev)

Monitoring/error diagnostics

If MD20 = 0 is input, all monitoring functions are active.

Individual monitoring functions can be inactivated by entering 0 in the designated bit of MD20.

You can deactivate the error messages using the single function “parking axis” (see Section 9.3.2).

Table 9-10 Error diagnostics – Incremental encoder

Diagnosis	Cause	Error message
Cable break monitoring	Signals of one track pair (A, $\overline{A}$ / B, $\overline{B}$ / N, $\overline{N}$) do not behave as negations of one another.	The FM 354 responds with a diagnostic interrupt, external channel error (see Troubleshooting, Table 11-4)
Pulse monitoring	• Signal track missing • Actual no. of pulses/rev $\neq$ MD13 • No signal change to a track pair	• If the contents of the missing-pulse memory exceed a value of 7, a diagnostic interrupt, external channel error is output (see Troubleshooting, Table 11-4). • The missing–pulse memory is cleared with the control signal “Restart” . **Note:** For encoders with non-cyclic zero pulse $\rightarrow$ switch pulse monitoring in MD20 OFF.
	In Reference-Point Approach mode, no zero pulse is recorded after leaving the reference-point switch within the path as defined in MD11, 12.	**Effect:** • Encoder cannot be synchronized. • On leaving the reference-point switch in Reference-Point Approach mode, the FM 354 will travel no more than the distance of one encoder revolution (MD11), and needs the deceleration distance from the reducing speed.
Voltage monitoring	Encoder power failure	Diagnostic interrupt, external channel error (see Troubleshooting, Table 11-4)

Warning

Hardware monitoring functions should be skipped only for test purposes, since positioning errors may destroy the machine.

Exception:

Pulse monitoring for encoders with non-cyclic zero pulse.

Encoder connection

See Section 4.5.

9.6.2 Absolute encoders (SSI)

Overview

Absolute encoders (SSI) have several significant advantages over incremental encoders:

- Longer cable lengths
- Reliable data capture by using a single-step GRAY code
- No encoder synchronization needed.

Absolute encoders (SSI)

You can use 13-bit single-turn encoders, or 25-bit multi-turn encoders with the SSI protocol.

- **Absolute encoder (SSI) on linear axes**

  Make sure the value range of the encoder is at least equal to the traversing distance of the axis.

- **Absolute encoder on rotary axes**

  Make sure that the absolute value range captured by the encoder corresponds to a ratio of 2^x or 2^{-x} to one revolution of the rotary axis, and that it encompasses at least one rotary axis revolution (see “Dependencies” Section 5.3.1 and Figure 9-5).

Function parameters

Table 9-11 shows you how to adapt the selected encoder to the FM 354.

Table 9-11 Function parameters – Absolute encoders (SSI)

MD	Designation	Value/Meaning	Comments/Unit
10	Encoder type	**3 = Absolute encoder (SSI 13 Bit)** **4 = Absolute encoder (SSI 25 Bit)** **13 = Absolute encoder (SSI 13 Bit)** **14 = Absolute encoder (SSI 25 Bit)**	GRAY Code GRAY Code Binary Code Binary Code
11	Displacement per encoder revolution (division period)	1...1 000 000 000 see Section 5.3.1, Dependencies	(MSR) (integer portion)
12	Residual distance per encoder revolution	$0...2^{32}–1$ see Section 5.3.1, Dependencies	(2^{-32} MSR) (fractional portion)
13	Increments per encoder revolution (division period)	$2^1...2^{25}$ see Section 5.3.1, Dependencies	Entry according to encoder rating plate
14	Number of revolutions of SSI encoder	0/1 = Single-turn encoder $2^1...2^{12}$ for multi-turn encoder	Only powers of 2 allowed

Table 9-11 Function parameters – Absolute encoders (SSI), continued

MD	Designation	Value/Meaning	Comments/Unit
15	SSI baud rate	1 = 78,000 Baud 2 = 156,000 Baud 3 = 312,000 Baud 4 = 625,000 Baud 5 = 1,250,000 Baud	(Code number) The baud rate depends on the cable length between FM 354 and encoder
19.0	Direction adjustment	**1 = invert measured value direction**	–
20 20.1 20.3	Hardware monitoring	 **1 = Error in absolute encoder** **1 = Voltage monitoring**	Entry for monitoring to be switched on

MSR stands for measurement system raster (see Section 5.3.1)

Sample encoder adjustment

Encoder: Number of increments per revolution (MD13) = 4096 = 2^{12}

Number of revolutions (MD14) = 256 = 2^8

Machine axis design:

- Motor with 50:30 gear ratio on spindle with 10 mm pitch = 10,000 MSR
- Encoder on motor.

From this one can calculate the following traversing distance per encoder revolution:

Gear ratio: $i = \frac{\text{50 spindle revolutions}}{\text{30 motor revolutions}} = 1.666666...$

Displacement per encoder revolution = $i \cdot 10{,}000$ MSR = 16,666.666... MSR

The following values are entered:

MD	Value	Unit
11	16 666	(10^{-3} mm)
12	$0.666... \cdot 2^{32} = 2{,}863{,}311{,}530$	($2^{-32} \cdot 10{-}3$ mm)
13	4096	(puls/rev)
14	256	(rev)

Note

The encoder covers an absolute traversing distance of 256 · 16,666.666... MSR. In the 10^{-3} mm system of measurement this corresponds to a maximum axis traversing distance of 4,266.666... mm.

Monitoring/error diagnostics

If MD20 = 0 is input, all monitoring functions are active.

Individual monitoring functions can be inactivated by entering 0 in the designated bit of MD20.

You can deactivate the error messages using the single function “parking axis” (see Section 9.3.2).

Table 9-12 Error diagnostics – Absolute encoder

Diagnosis	Cause	Error message
Voltage monitoring	Encoder power failure	Diagnostic interrupt, external channel error (see Troubleshooting, 11-4)
Error in absolute encoder	Error in protocol for data transfer between absolute encoder and FM 354	Diagnostic interrupt, external channel error (see Troubleshooting, 11-4)

Warning

Hardware monitoring functions should be skipped only for test purposes, since positioning errors may destroy the machine.

When voltage monitoring is deactivated and the power supply to the encoders or the FM is switched off, an immediate failure in absolute value signaling can cause drive movements if:

- an operating mode other than open–loop control is active
- and follow–up mode is deactivated
- and the servo enable is activated or not parameterized.

Encoder connection

See Section 4.5.

9.6.3 Synchronizing the encoders

Overview

When using incremental encoders, at switch-on there is an offset, which cannot be determined in advance, between the internal position value in the FM and the mechanical position of the axis. To establish the position reference, the value internal to the FM must be synchronized with the real position value of the axis. Synchronization is performed by taking over a position value at a known point of the axis.

When using absolute encoders (SSI), at switch-on there is already a defined relationship between the position value internal to the FM and the mechanical position of the axis. This reference can be adjusted by setting an absolute encoder alignment value (see Section 9.3.9, Setting the reference point).

Absolute encoder alignment

is the compensation value for numerical alignment of the internal FM position value.

Reference point approach

is an operating mode used to position the axis at the reference point.

Reference point

is a fixed point on the axis. It is:

- The target coordinate in the Reference-Point Approach mode
- Removed from the synchronization point by the amount of the reference-point shift, in axes with incremental encoders.

Reference point offset

Difference in distance between the synchronization point and the reference point.

The reference-point shift serves:

- for numerical measuring-system readjustment when an encoder is changed
- as a displacement reserve to brake the drive if the synchronization point is overshot.

Reference point switch (RPS)

The reference point switch selects the synchronizing (encoder) zero marker on the traversing path of the axis.

- It is also the signal encoder for a speed reduction before the synchronization point is reached.
- It is connected to a digital input of the FM 354.

Synchronization point

is a defined point on the traversing path of the axis. It is defined by the mechanical position of a reference-point switch or in association with a cyclic zero mark of an incremental encoder.

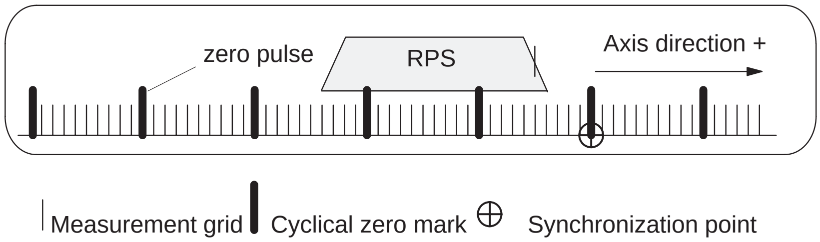
Synchronization

Creating the position reference between the internal FM position value and mechanical position of the axis.

Measured value synchronization with incremental encoders

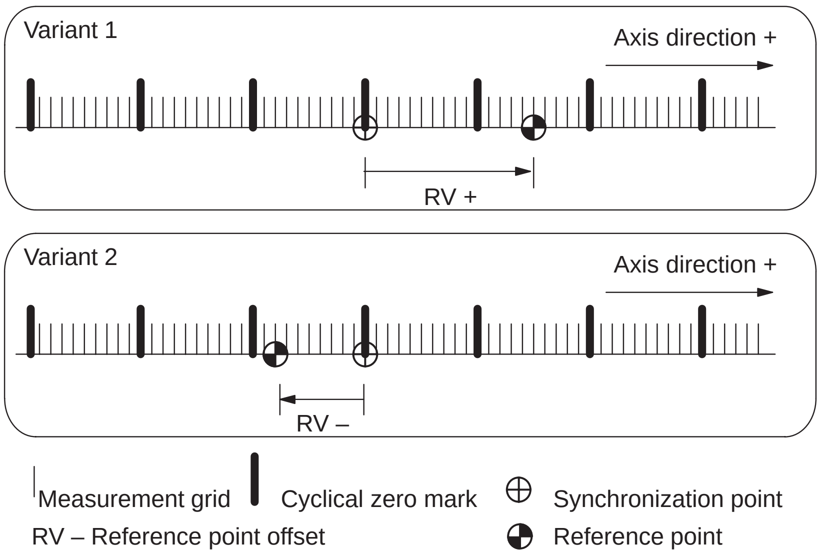
Irrespective of the approach direction, the synchronization point can be located on the side of the lower or the higher actual position values in relation to the reference point switch position. It is demarcated by the rising edge of a zero pulse or by the reference point switch. This selection is made by the MD18 (together with the approach direction).

Example



With reference to the synchronization point, the reference point can be located on the side of the lower or the higher position actual values. In the “reference point approach” operating mode the machine axis additionally traverses this distance, during its last phase of motion, once the synchronization point has been found. Consequently, the axis halts the motion, in each instance, exactly on the reference point.

Example



9.7 Position controller

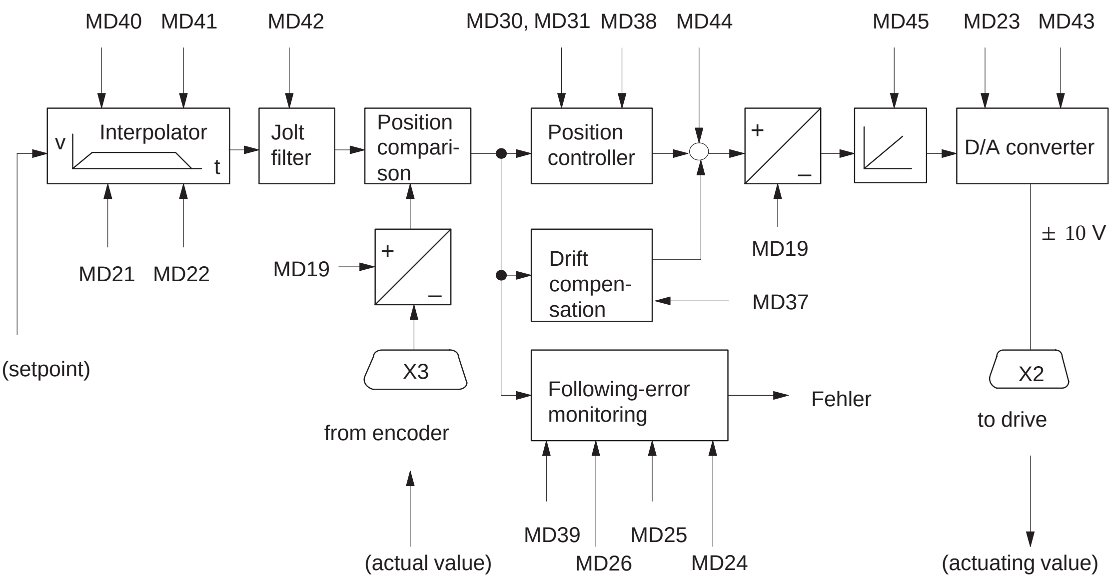
Overview

The closed-loop position controller performs the following tasks:

- guidance of the drive at the proper speed during execution of a movement
- precise approach of the axis to the programmed target position
- maintenance of the axis in position in the face of interfering factors.

The position controller is configured as a proportional-action controller. In its environment are a number of function units that provide support by performing special tasks within the complex of movement control, and that can be adapted to axis conditions by means of a variety of machine data.

The diagram 9-6 gives you an overview of the function units and the machine data acting upon them.

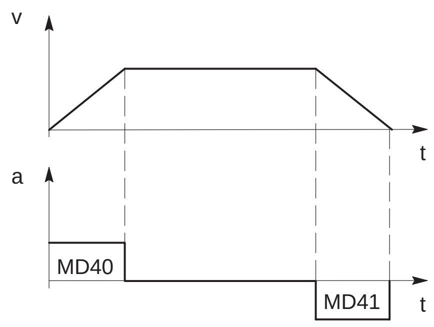


MD19 –	Direction adjustment	MD37 –	Control signal Servo Bit 16
MD21 –	Software limit switch - start	MD38 –	Positioning loop amplification
MD22 –	Software limit switch - stop	MD39 –	Min. following error, dynamic
MD23 –	Maximum speed	MD40 –	Acceleration
MD24 –	Target range; position reached, stop (PEH)	MD41 –	Deceleration
		MD42 –	Jolt time
MD25 –	Monitoring time	MD43 –	Set voltage, max.
MD26 –	Idle range	MD44 –	Offset compensation
MD30 –	Backlash compensation	MD45 –	Voltage ramp
MD31 –	Directional reference, backlash		

Fig. 9-6 Overview of position controller

Interpolator

The machine data for acceleration (MD40) and deceleration (MD41) can be used to adapt the transition response of the command variable defined by the interpolator to the transition response of the controlled system.



v – speed

a – acceleration

t – time

MD	Designation	Value/Meaning	Unit
40	Acceleration	0 = without ramp 1...100,000	[10^3 MSR/s^2]
41	Deceleration		

MSR stands for measurement system raster (see Section 5.3.1)

Software limit switches

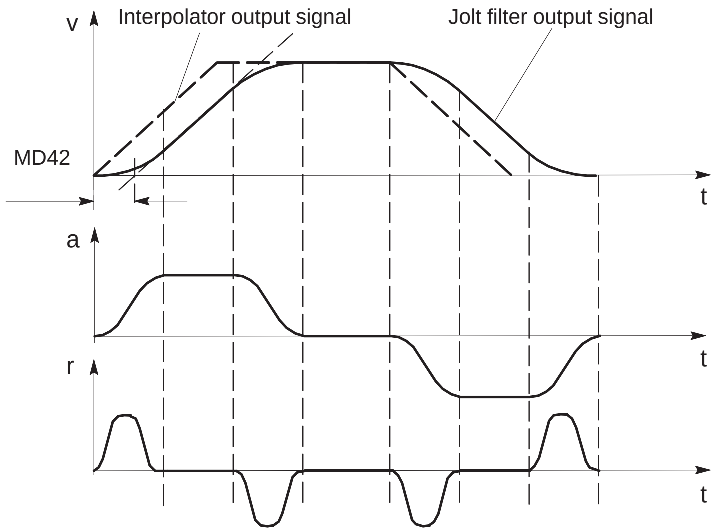
Software limit switches MD21 and MD22 (see Section 9.9) are used to limit the working area.

MD	Designation	Value/Meaning	Unit
21	Software limit switch, start	–1 000 000 000...< +1 000 000 000 see Section 5.3.1, Dependencies	[MSR]
22	Software limit switch, stop		

Jolt filter

Without jolt limitation the acceleration and deceleration act as abrupt variables. Jolt limitation allows the break points of a ramp-like speed curve to be smoothed out for both acceleration and deceleration. This yields particularly “soft” (jolt-free) acceleration and braking for certain positioning tasks, such as conveying of fluids.

Jolt time can be set in MD42 as the parameter for jolt limitation.



v – speed
a – acceleration
r – jolt
t – time

MD	Designation	Value/Meaning	Unit
42	Jolt time	0...10,000	[ms]

Basic diagnostics in the position control loop

In servo–controlled mode, the manipulated variable is compared periodically with the possible maximum values ($\geqslant$ 10 V or maximum frequency). A violation of the maximum limit is interpreted as follows:

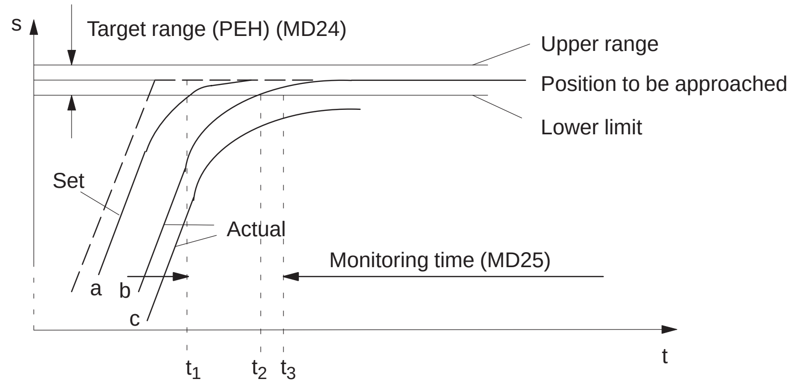
No axis movement:	“No drive movement” error message (see Table 11-7, Class 3/No. 65)
Traverse in the opposite direction:	“Direction of drive rotation” error message (see Table 11-5, Class 1/No. 11)
Correct travel direction:	Oversteer message in status message 1 (see Section 9.3.16, Additional operating data)

In all operating modes except Control mode, the “servo enable” signal is required for the duration of every traversing movement, irrespective of the parameter definitions. If the servo enable is not detected or is deactivated during the movement, the “servo enable missing” message is triggered (see Table 11-5, Class 3/No. 61).

In all operating modes except Open–loop control mode, the “servo ready” signal is required for the duration of every traversing movement when the parameter is active (MD37.2). If the servo ready is not detected or is deactivated during the movement, the “servo not ready” message is triggered (see Table 11-5, Class 3/No. 62).

Position approach

The following diagram shows position approach.



PEH – Position reached, stop
s – path
t – time

On approach to a position, the monitoring time is activated:

Time	Position monitoring
t_1 (a)	After the interpolator reaches the target position, the monitoring time (MD25) for reaching the target range is started in the CL controller, after the overtravel in the jolt filter dies down to the target range value (PEH on setpoint side).
t_2 (b)	Before the monitoring time expires, the actual position reaches the target range. Positioning is completed. A PEH is signaled, and exact matching is performed by the CL controller.
t_3 (c)	After the monitoring time expires, the actual position has not reached the target range (PEH). **Error message:** “PEH Target Range Monitoring” (see Troubleshooting, Table 11-7, Class 3/No. 64)

MD	Designation	Value/Meaning	Unit
24	Target range	0...1 000 000	[MSR]
25	Monitoring time	0 = no monitoring 1...100,000	[ms], rounded to 2–ms steps

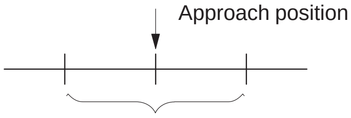
Following error monitoring

Axis standstill

A message is output on an axis standstill setpoint or deactivated servo enable if disturbances cause the axis to move out of position.

MD	Designation	Value/Meaning	Unit
26	Stationary range	0...1 000 000	[MSR]

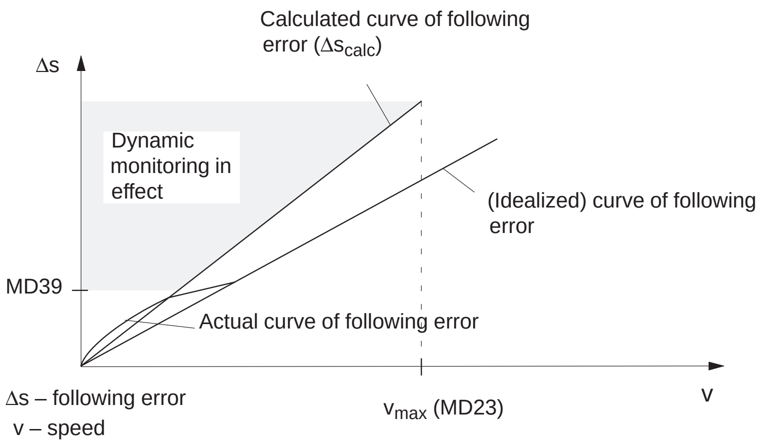
The standstill zone is located symmetrically around the target approach position.



When the tolerance window for idle is exceeded, the FM 354 signals a "Stationary Range" error (see Troubleshooting, Table 11-5, Class 1/No. 12).

Axis moving

To monitor following error during movement, the FM 354 calculates the allowable following error for the instantaneous traveling speed from the parameterized positioning loop amplification (MD38). Above the parameterized "Minimum following error (dynamic)," a comparison is performed with the actual value for the following error.



MD	Designation	Value/Meaning	Unit
39	Minimum following error (dynamic)	0 = no monitoring 0...100,000	[MSR]

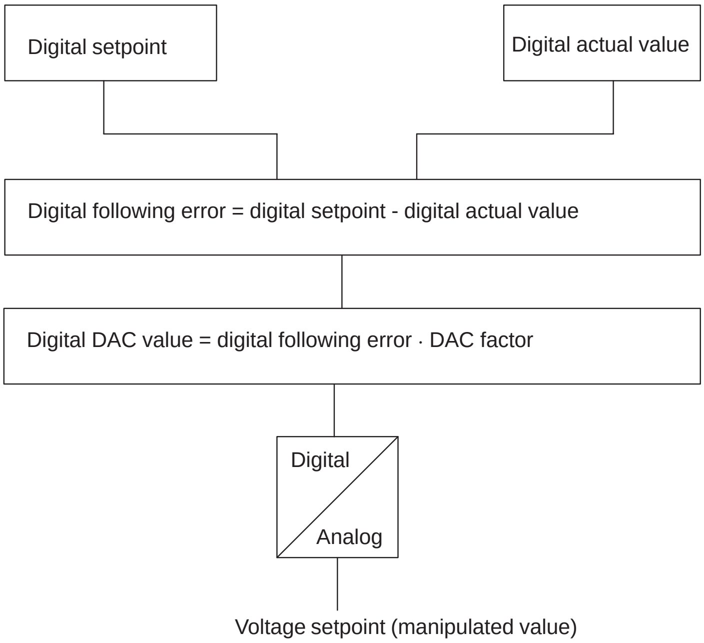
When the calculated following-error limit is exceeded, the FM 354 signals a "Following error too great" error (see Troubleshooting, Table 11-7, Class 3/No. 66).

Exception:

If an axis standstill occurs above the "minimum dynamic following error",the error message described under Basic diagnostics "no drive movement" is output (see Table 11-7, Class 3/No. 65).

Position controller

The following error is calculated by periodical comparison of the set position defined by the interpolator with the actual position of the axis detected by the encoder. From this error, the position controller calculates the actuating signal needed for the drive in order to balance to a following error value of zero. The actuating signal is provided via a D/A converter (DAC) as an analog signal at the FM 354 output.



Position control loop gain

The positioning loop amplification (K_v factor) specifies at what speed of axis travel a given following error sets in. The mathematical (proportional) relationship is:

$$K_v = \frac{\text{Velocity}}{\text{Following error}} = \frac{v\ [10^3\ \text{MSR/min}]}{\Delta s\ [\text{MSR}]}$$

Although the magnitude of the following error plays no dominant role for a single axis, the K_v factor still affects the following important characteristics of the axis:

- Positioning accuracy and stopping control
- Uniformity of movement
- Positioning time

The following relationship applies for these characteristics:

The better the axis design, the greater the achievable K_v factor, and the better the axis parameters from the technological viewpoint. The size of the K_v factor is especially affected by the time constants, backlash and spring components in the controlled system. In real applications the K_v factor moves within the following bandwidth:

- K_v = 0.2...0.5 poor-quality axis
- K_v = 0.5...1.5 good axis (normal case)
- K_v = 1.5...2.5 high-quality axis

The MD38 value is input with a resolution of 10^3, so that the following input value results:

$$\text{MD38} = 10^3 \cdot K_v = 10^3 \cdot \frac{\text{Velocity}}{\text{Following error}} = 10^3 \cdot \frac{v\,[10^3\ \text{MSR/min}]}{\Delta s\,[\text{MSR}]}$$

MD	Designation	Value/Meaning	Unit
38	Positioning loop amplification	0...10,000	[(MSR/min)/MSR]

Drift compensation

Thermal conditions will shift the zero error in the control loop during operation. This effect is called drift. In a closed control loop with a proportional-action controller, this results in a temperature-dependent positioning error. You can activate automatic drift compensation with MD37, under which continuous balancing takes place in the positioning control loop.

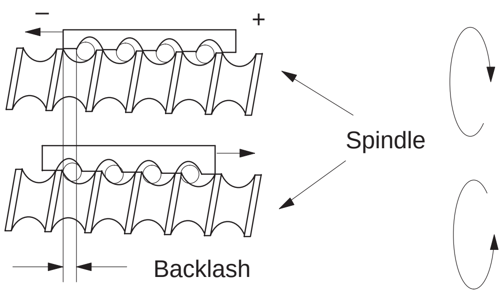
A basic compensation of the zero point error by means of the offet is required for the optimum effect of the drift compensation
(see MD44, offset compensation).

MD	Designation	Value/Meaning	Unit
37	Servo control signal	16 = automatic drift compensation active	–

Backlash compensation

Mechanical drive components as a rule have a certain amount of backlash (free play).

Mechanical reversing backlash can be compensated with MD30. In an indirect measuring system (with the encoder on the motor), the mechanical backlash is traveled at each change of direction before any axis movement occurs. The result is positioning errors.



When the position encoder is situated on the machine part to be positioned (e.g. on a saddle - direct arrangement), backlash adversely affects the achievable K_v factor. On the other hand, if the position encoder is attached to the drive motor (indirect arrangement), a high K_v factor can be achieved, but at the cost of position deviations that cannot be detected by the position controller. A backlash amount entered in MD30 is applied as a correction by the position controller as a function of the traveling direction at a given moment, thus achieving an approximate compensation for backlash in positioning.

MD31 is used to label the “backlash-free” or “accurate-measurand” traveling direction of the axis. If MD31 = 0, the “backlash-free” direction is the one that matches the direction of axis movement when synchronization is recorded. Depending on MD18, this will correspond to the following association:

MD18 = 0, 2, 4, 8: Plus direction is backlash-free
MD18 = 1, 3, 5, 9: Minus direction is backlash-free

MD	Designation	Value/Meaning	Unit
30	Backlash compensation	–1 000 000...+1 000 000	(MSR)
31	Directional reference of backlash	0 = as in reference point approach (not for absolute encoders) 1 = positive 2 = negative	–

Offset compensation

The analog modules in the positioning control loop (D/A converter of the FM354 and closed-loop controller module of the drive) cause a zero error because of operating-voltage and component tolerances. The result is that at an internal digital rotational-speed specification of zero in the FM 354, the drive motor will already be running undesirably. As a rule, drive controllers have adjustment capabilities for balancing. But by setting a voltage offset via MD44 the analog system can be balanced at startup from the FM side.

MD	Designation	Value/Meaning	Unit
44	Offset compensation	–1,000...+1,000	(mV)

For calculation of the offset value, see Section 7.3.2, Drive interface.

Direction alignment

MD19 allows you to align the direction by defining an assignment between the voltage sign of the manipulated signal and the axis movement.

MD	Designation	Value/Meaning	Unit
19.1	Direction adjustment	1 = invert analog value	–

Voltage ramp

A ramp–shaped voltage rise/drop can be defined in MD45 for the voltage output to the drive when the position controller is inactive. This serves to limit acceleration and thus power for the drive controller, and is preferable to setting options that may be available on the drive, since it has no adverse effects on active position control.

The voltage ramp is active in the following situations:

- Continuously in Control mode
- Deceleration on cancelation of the drive enable [AF] (see Section 9.1.1)
- Deceleration on transition of CPU from RUN to STOP
- Deceleration on error response “Everything Off” (see Sections 11.1, Tab. 11-4 and 11-5)

MD	Designation	Value/Meaning	Unit
45	Voltage ramp	0...10,000,000	(mV/s)

D/A converter

Output of analog voltage setpoint

Velocity assignment and max. voltage setpoint

The manipulated signal calculated by the position controller is available internally on the FM as a velocity setpoint (see position loop gain). To convert this value to the analog actuating signal, a conversion factor (DAC factor) within the FM is necessary. This factor is formed as the quotient of MD43 and MD23. MD23 contains the configured maximum speed of the machine axis, and MD43 contains the voltage setpoint of the actuating signal to be output by the FM 354 for this purpose; as a compromise between the highest possible resolution and adequate close-loop control reserve, this voltage should lie between 8 V and 9.5 V.

Warning

This assignment MUST be identical with the setting on the drive!

MD	Designation	Value/Meaning	Unit
23	Maximum speed	10...500,000,000	(MSR/min)
43	Set voltage, max.	1,000...10,000	(mV)

9.8 Digital inputs/outputs

Overview

Four digital inputs and four digital outputs of the FM 354 can be used specifically to a given application.

The conventions and parameterization for this purpose are defined in the machine data MD34 to MD36.

The signals are processed in the FM cycle.

The signal status of the digital inputs and outputs can be recognized by read-back (**job no. 101**).

Function parameters

Table 9-13 shows you the functions assigned to each digital I/O.

Table 9-13 Function parameters for digital I/Os

MD	Designation	Data type, bit array/meaning				
34	Digital inputs[1)]	**I0**	**I1**	**I2**	**I3**	
		0	8	16	24 =	External start[2)]
		1	9	17	25 =	Enable input
		2	10	18	26 =	External block change
		3	11	19	27 =	Set actual value on-the-fly
		4	12	20	28 =	Measurement (inprocess measurement, length measurement[2)])
		5	13	21	29 =	Reference point switch for reference point approach[2)]
		6	14	22	30 =	Reversing switch for reference point approach[2)]
35	Digital outputs[1)]	**Q0**	**Q1**	**Q2**	**Q3**	
		0	8	16	24 =	Position reached, stop
		1	9	17	25 =	Aaxis movement forwards
		2	10	18	26 =	Axis movement backwards
		3	11	19	27 =	Change M97
		4	12	20	28 =	Change M98
		5	13	21	29 =	Start enable
		7	15	23	31 =	Direct output

1) see Section 5.3.1, Dependencies
2) Signal length $\geq 2 \cdot$ FM cycle

Level adjustment

MD	Designation	Value/Meaning	Comments
36	Input adjustment	8 = I0 inverted 9 = I1 inverted 10 = I2 inverted 11 = I3 inverted	Front edge always activates the function

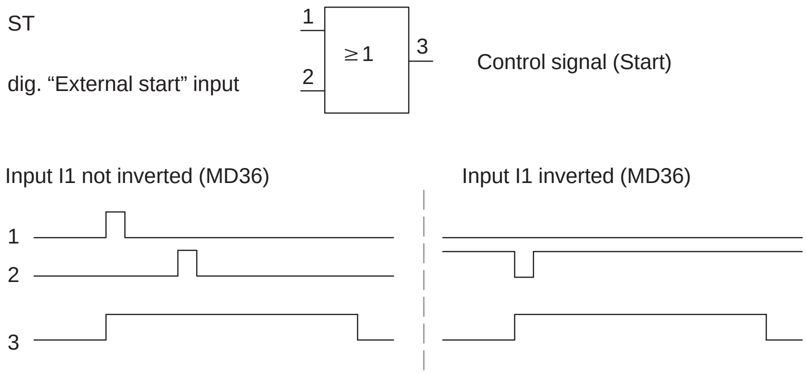
9.8.1 Function description of digital inputs

External start

The control signals of the axis include the start signal which triggers a positioning operation in “Reference point approach”, “MDI” and “Automatic” modes. A logical OR is established with the “External Start” digital input and the control signal (ST).

External start is connected to digital input I1.

Example



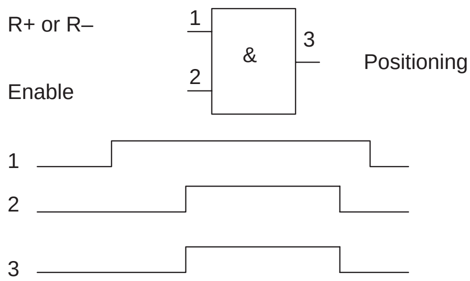
Minimum signal length at the digital input: ≥ 2 · FM cycle

Enable input

The enable input signal must be set, if defined in MD34, for a positioning operation/movement/output of the axis to take place. A reset stops the movement (external movement enable).

- In the Jogging and Control modes, the movement of the axis proceeds as long as the AND link continues between the control signal (R+/R–) and the enable input.

Example



- In the other modes, note the following:

  If the enable input is still not enabled after a start edge, the start edge is stored internally and “waiting for enable” is indicated by the checkback signals. When the input is set, movement begins and the stored Start edge is deleted (a Stop likewise deletes the stored Start edge).

External block change

see Chapter 10

Set actual value on–the–fly

see Chapter 10 and Section 9.3.6

Measurement

see Section 9.3.10

Reference point switch for reference point approach

see Section 9.2.3

Reversal switch for reference point approach

see Section 9.2.3

9.8.2 Function description of digital outputs

Output of PEH, FR+, FR–, SFG

The following checkback signals: position reached, stop (PEH), axis movement forward (FR+), axis movement in reverse (FR–), and enable Start (SFG), are additionally output via digital outputs. The output assignment is parameterized by way of MD35.

Output of change M97 or M98

The change M-function (AMF) checkback signal for the M functions M97 and M98 is output as a digital output. It allows these M functions (switching signals) to be applied without being delayed by the user cycle time.

Direct output

Outputs Q0...Q3 (D_OUT1...D_OUT4), which are defined in MD35 as “direct output”, can be used directly by the user program (**job no. 15**) and can also be controlled by the FM 354.

Since the same memory is used in the user DB for job 15 and job 101, the jobs cannot be used simultaneously in the cycle.

Note

The outputs are subject to deactivation on module errors of error classes with the response “Everything Off”.

9.9 Software limit switches

Overview

To limit the working range, entries in the machine data (MD21 and MD22) specify the start and stop limit switches. These limit switches are active at synchronization of the axis.

If the limit switches are not needed, values lying outside the possible working range should be entered in the machine data (MD21 and M22), or monitoring should be switched off via the user program.

Warning

The software limit switches do not replace the hardware limit switches for EMERGENCY STOP responses.

Effect of software limit switches in modes

"Jogging mode"

At the limit switch the traveling movement is stopped in the limit-switch position, and an error is signaled.

"Control mode"

If the actual value is beyond the end position, the traveling movement is stopped and an error is signaled. The limit-switch position is overshot by the amount of the necessary deceleration distance.

"Reference-point approach" mode

No effect.

"Incremental relative", "MDI", "Automatic mode"

Movement is stopped, or not even started, as soon as read-in of the set position reveals that the position lies outside the working range. An error is signaled.

The following special cases exist:

- Continuous travel (–) for set actual value on–the–fly (G88 see Chapter 10)
- Continuous travel (+) for set actual value on–the–fly (G89 see Chapter 10)

Effect of software limit switches in tracking mode

If the actual value is beyond the end position, an error is signaled.

Response after error

Leaving end position or traveling into working range after error

1. Acknowledge the error message!
2. Travel to the working range with the "Jogging", "Control", "Incremental Relative" or "MDI" mode.

Rotary axis

The end position of MD_{start} may be greater than MD_{stop}.

When traveling into the working range (e.g. end position was previously switched off), the shortest path is always chosen.

If both default values are parameterized the software limit switches are inactive.

9.10 Process interrupts

Overview

Process interrupts are interrupts that quickly signal states in the current process to the user program.

The appropriate setting in the machine data (MD5) specifies which signals are to be quickly communicated to the user program.

Process-interrupt generation

The process interrupt is generated by way of machine data item MD5:

MD	Designation	Significance
5	Process-interrupt generation (data type - bit field)	0 = Position reached 1 = Length measurement completed 3 = Change block on-the-fly 4 = Inprocess Measurement

Hint to the user

You must program the interrupt processing routine in OB40.

The prerequisite is that process-interrupt signaling must have been activated as part of the environment definition (see Chapter 5).

■

Writing Traversing Programs

10

Overview

To execute the desired operations of the machine axis (sequence, position, etc.) in “Automatic” mode, the FM 354 needs certain information. This information is programmed with “Parameterize FM 354” (traversing program creation) in the form of a traversing program, based in principle on DIN 66025.

Traversing programs

Each traversing program is stored under a program number.

A traversing program consists of not more than 100 traversing blocks.

The program number and traversing blocks are converted to an internal format (see Section 9.3.12), are stored in the appropriate data block, and are transferred to the module. where they are administered.

The possible number of programs depends on the amount of memory available (max. 16 Kbytes) and on the length of the individual programs.

Program length in bytes: 110 + (20 x no. of traversing blocks)

Program name

Any program can be assigned a name (optional).

The program name may have up to 18 characters, and is saved with the program.

Program number

Programs may be numbered from % 1 to % 199.

Traversing block

A traversing block contains all the data required to perform a machining step.

Program structure

A program consists of several blocks. Each block number occurs only once, and numbers are arranged in ascending order.

A sample program structure follows:

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D	L	P	
	5	90			500 000	100 000	10						Start of program = lowest block number
	6	91			...	...							
	7	...											
	:												
	45												
	46						2						End of program = M2 or M30

Chapter overview

10.1 Traversing blocks

Block structure

The following Figure gives you an overview of the structure of traversing blocks.

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D	L	P

/ – Identifier for skipped block
N – Block number
G1 – G function of first function group
G2 – G function of second function group } see Table 10-1
G3 – G function of third function group
X/t – Position/dwell time
F – Speed
M1 – M function of first function group
M2 – M function of second function group } see Table 10-2
M3 – M function of third function group
D – Tool offset number
L – Call a program as a subprogram
P – Number of subprogram calls

Skip block /

Program blocks which are not to be executed every time the program runs can be identified as skippable blocks by an oblique "/ ". When the program is being processed, the "Skip block" control signal can be used to decide whether skippable blocks are to be skipped. The last block cannot be skippable.

Block number N

The program is executed in ascending order of block numbers, from 1 to 255, or in descending order if executed in reverse.

G function group 1...3

In each traversing block only one G function may be entered from each G function group.

The following figure shows an example.

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	10	90	34	43	100 000	400 00				10

G functions

Table 10-1 lists the possible G functions and the individual G function groups.

Table 10-1 G functions

G No.	G Function	G Function group
04[1)]	Dwell time	1
87	Turn off measuring system shift for Set Actual Value On-the-Fly	
88[1)]	Continuous travel for (–) for Set Actual Value On-the-Fly	
89[1)]	Continuous travel for (+) for Set Actual Value On-the-Fly	
90	Absolute measure	
91	Chain measure	
30 31 32 . . . 39	100% override on acceleration/deceleration 10% override on acceleration/deceleration 20% override on acceleration/deceleration . . . 90% override on acceleration/deceleration	2
43	Tool offset (+)	3
44	Tool offset (–)	
50[1)]	External block change	
60	Block change – exact positioning	
64	Set actual value on-the-fly, continuous–path mode	

1) These G functions take effect only on a block-by-block basis. The other G functions remain active until canceled explicitly.

G30, G90 and G64 are the **initial settings** after the start of the program.

Dwell G04

A traversing block with dwell can only contain M functions and the time parameter apart from this G function.

The following applies for dwell time:

Name	Lower input limit	Upper input limit	Unit
Dwell time	2	100,000	ms

Odd input values are rounded upward. Dwell times take effect only on a block-by-block basis.

If no value is input for G04 in the block, the lower input limit applies.

Block change G60, G64 (approach conditions)

With G60, the exact programmed position is approached and the feed movement is stopped (exact stop block change).

G64 causes the next block to be processed immediately as soon as the point of deceleration is reached (change block on-the-fly).

G60 and G64 are mutually exclusive and self-maintaining.

M commands have no effect on G64 operation.
(For a detailed description, see Section 10.3).

External block change (G50) with delete residual path

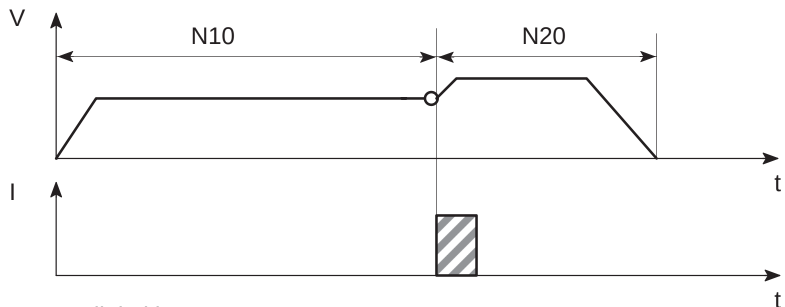
The “external block change” function causes a block change on–the–fly triggered by a digital input. The fast input must be parameterized with the “External block change” function by way of machine data item MD34.

The function takes effect only on a block-by-block basis (no effect on G60 and G64).

Example of external block change

The following figures show the program structure and program flow of an example of “External block change.”

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	10			50	10 000					
	20									



I – digital input

Notes to the example of external block change

The axis travels until a signal change from 0 to 1 takes place at the digital input. This triggers two reactions:

- A block change on-the-fly, and thus immediate processing of block N20.
- Storage of the actual position at the time of this signal change to “Actual value block change.” This position is also the starting position for any subsequent chain-measure programming.

Depending on the situation, N20 is processed as follows:

- If the block position in N20 is less than the actual position at the time when the digital input is received (reversal of direction), the equipment is stopped so that the position can then be approached in the opposite direction.
- If no position is programmed in block N20, movement is braked, the functions programmed in N20 are executed, and processing then moves on to the next block (except if the block contains M0, M2 or M30).
- If the programmed path in block N20 is less than the deceleration distance, the programmed position is overshot and then positioned by a reversal of direction.

If no signal change occurs at the digital input, the target position of N10 is approached, with the following additional response:

When the target position is reached, the error message “Digital input not actuated” is output (see Table 11-5, Class 2 No. 15).

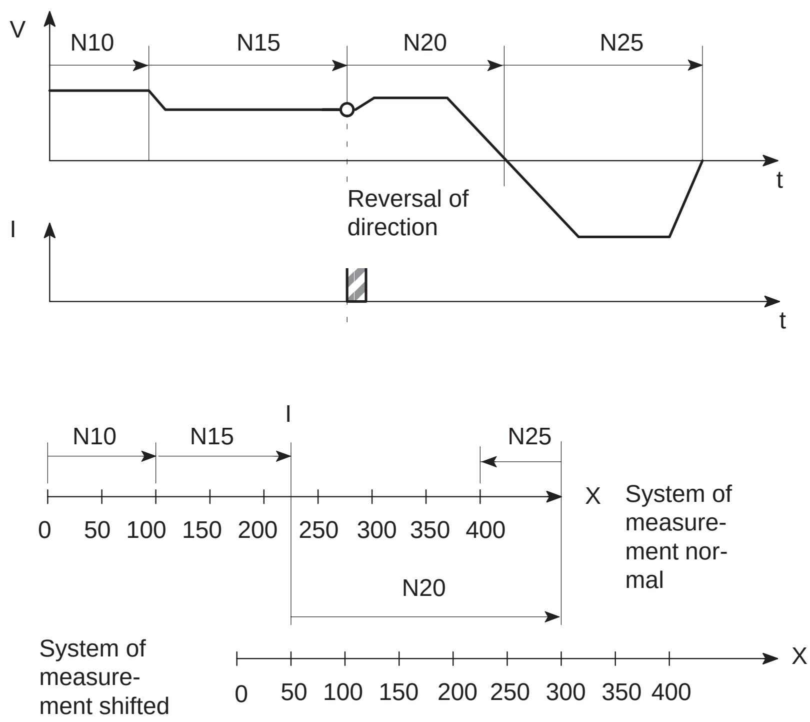
Set actual value on-the-fly G87, G88, G89

The “Set actual value on-the-fly” function is programmed and triggered by a digital input; the block change occurs on the fly and the actual value is set to a new dimension (programmed coordinate) at the same time. The digital input must be parameterized with the “Set actual value on-the-fly” function by way of machine data item MD34.

Example of set actual value on-the-fly

The following figures show the program structure, program flow and actual-value curve for an example of “Set actual value on-the-fly.”

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	10	90			100	400 000				
	15	89 (88)			50	200 000				
	20	90			300	400 000				
	25	87			400	400 000				



I – digital input

Notes to the example of set actual value on-the-fly

This changes blocks on-the-fly from N10 to N15, with G89 causing movement in a positive direction and G88 causing movement in a negative direction at the speed programmed in N15.

The axis now travels in the specified direction until a positive edge change occurs at the digital input. This triggers the following responses:

- Block change on-the-fly and immediate processing of block N20
- Set actual value on-the-fly to the block position from N15 (50 in the example), and resulting shift of the coordinate system
- Save current actual value.

The programmed position in block N20 refers to the shifted coordinate system.

At the block change from N20 to N25, G87 cancels the shift of the coordinate system and causes reference-measure programming to the block position of N25.

The saved actual value can be read out with “Actual value block change.”

The shift of the coordinate system is maintained until it is canceled by G87 or by a mode change. It is possible to use the existing shift of the coordinate system in different programs. The coordinate system can be shifted again without previously canceling an existing coordinate system shift.

G88, G89 can be programmed multiple times. The shift in each case refers to the original state. The software limit switches are always shifted concurrently.

If the signal change of the digital input does not occur, the axis runs until it reaches the limit switch.

Note

The G functions G87, G88 and G89 take effect only on a block-by-block basis and must be reselected if necessary.

Dimensions G90, G91

The traversing movement at a specific point can be described by

- Reference-measure input (absolute measure input) G90 or
- Incremental input (relative measure input) G91

You can switch back and forth at will between reference-measure and incremental input.

The status at startup is reference-measure programming, G90.

G90 and G91 are modal.

Absolute dimensioning G90

Absolute dimensioning is the method used to specify dimensions that generally refer to the coordinate system.

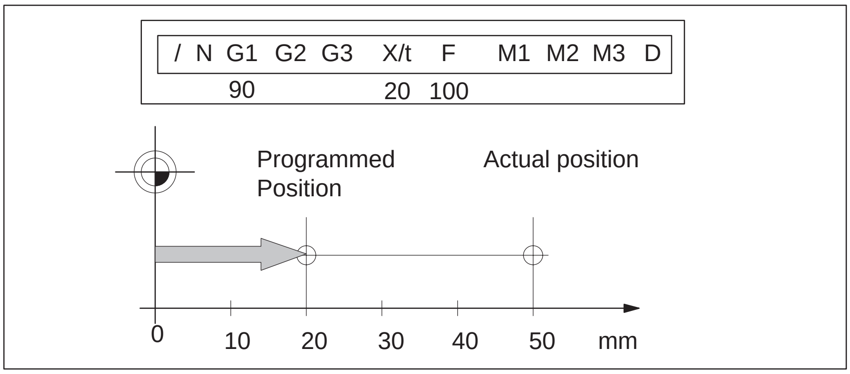


Fig. 10-1 Reference-measure input G90

Note

To ensure precise reproduction of the program, the first block should contain reference-measure programming.

Incremental dimensioning G91

Incremental dimensioning is the method used to specify incremental dimensions that refer to the last actual position.

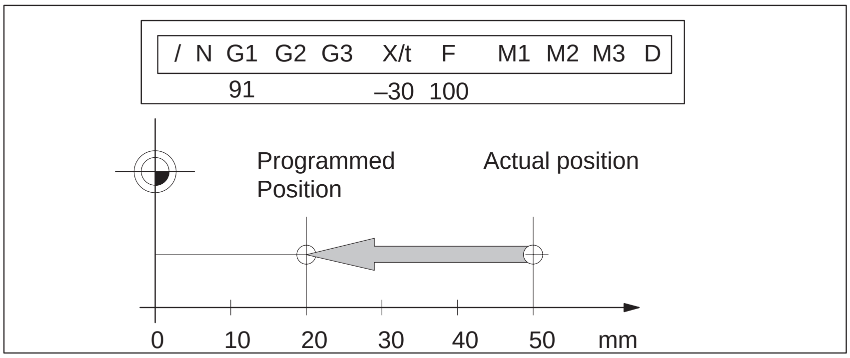


Fig. 10-2 Incremental input G91

Axis as rotary axis

If the axis is operated as a rotary axis, the measuring system must be adjusted in such a way that the measurement scale refers to the full circle (e.g. 0° and 360°).

- Reference-measure input G90

  In a full circle with 360°, reference-measure programming (G90) has the peculiarity that there are always two options for reaching the set position.

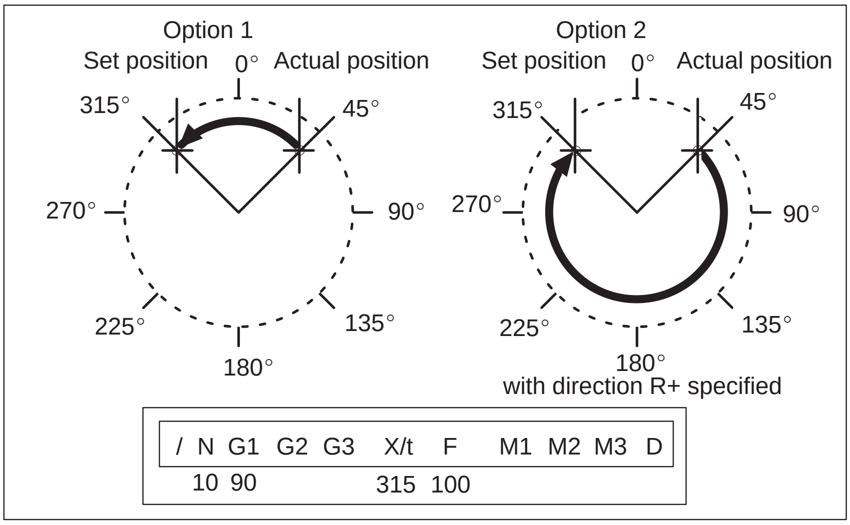


Fig. 10-3 Rotary axis

Option 1:

With G90, the axis autonomously always takes the shortest path to reach the set position of 45°, going via 0° to 315°.

Option 2:

The control signals (R+) or (R–) force the respective direction of the axis - in this example 45° via 180° to 315°. (R+) or (R–) must already be pending when positioning is activated (START).

Note

The direction (R+) or (R–) must be specified sufficiently in advance. A traversing direction **cannot** be forced on a traversing block that is already active, or on the traversing blocks (up to 4) that have already been calculated in advance in G64 operation.

Operation with option 1 or option 2 is at the user's discretion.

- Incremental input G91

  With incremental programming G91, the direction of rotation of the rotary axis is defined by the sign of the position setpoint. Multiple rotations can be programmed by setting a value > 360° as the position setpoint.

Acceleration override G30...G39

The acceleration override is used to control acceleration and deceleration during positioning movements. The acceleration and deceleration values are set by machine data. G30 through G39 in the traversing block can be used to achieve a percentage reduction in both values. These functions are modal.

G Function

30	100% override for acceleration/deceleration
31	10% override for acceleration/deceleration
to	
39	90% override for acceleration/deceleration

Changing the acceleration override in the program prevents block change on-the-fly. Consequently G60 response is forced in the preceding block.

The acceleration override is turned off by:

- Mode changes
- Resetting the axis with a Restart (single command)
- Changing or ending the program.

Tool compensation G43, G44

Tool compensation allows you to continue using an existing machining program, even when the tool dimensions have changed.

Tool offset is selected with G43 or G44, as applicable, and the tool offset number D1...D20. Tool offset is turned off with G43 or G44, as applicable, and the tool offset number D0.

A total of 20 tool offset storage areas and tool wear storage areas are available. The values are loaded to the module with the “Tool offset data” data block and are saved permanently. When selected, changed or turned off, the tool offset is not taken into account until the next positioning action.

A selected tool offset is maintained in effect until it is either turned off or replaced with a new one. Likewise a mode or program change, or the end of a program, will turn tool offset off.

Variants in tool offset

Tool offset is made up of two correction-value components:

- Tool length offset

  The tool length offset is the actual tool length from tool zero to the tool tip.

- Tool length wear value

  The tool length wear value allows the change in tool length due to wear to be compensated in two ways:

  Absolutely: by specifying a fixed wear value

  Additively: by adding an “offset value” to the current tool length wear value contents.

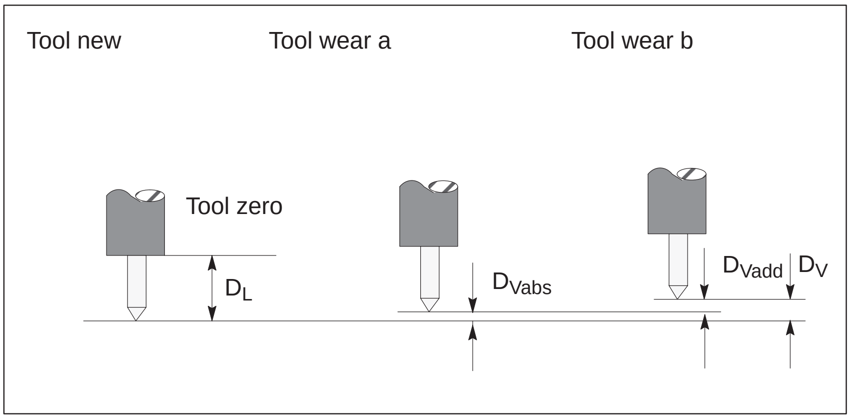


Fig. 10-4 Tool offset

Notes to the figure:

The tool offset thus consists of the tool length offset and the tool length wear value:

$D = D_L – D_V$

$D_V = D_{Vabs} + D_{Vadd}$

D	–	Tool offset
D_L	–	Tool length offset (positive or negative)
DV	–	Tool length wear value (positive or negative)
D_{Vabs}	–	Wear, absolute (positive or negative)
D_{Vadd}	–	Wear, additive (positive or negative)D_{Vadd}

Direction of tool offset

The functions G44 (–) and G43 (+) correct the position value in such a way that the tool tip reaches the programmed set position.

- **Negative tool offset G44**

  As a rule, the tool points to the workpiece in a negative direction. With the infeed adjustment, the positioning value (traversing path) becomes smaller.

  Referred to the measuring system, the following position is thus approached:

  $X_{ms} = X_{set} + (D)$

  X_{ms} – Position of measuring system
  X_{set} – Programmed set position
  D – Tool offset

- **Positive tool offset G43**

  The positioning value (traversing path) becomes greater with the infeed adjustment. The position value is corrected by:

  $X_{ms} = X_{set} - (D)$

To program a tool offset in the traversing block, at least the tool length offset must be input. If no correction is to be applied even when the function has been selected, the tool length offset and tool length wear value must be preset to 0.

A tool length wear value can be deleted by an absolute input of 0.

Position X

Positions may be input with a negative or positive sign. The plus sign on positive values may be omitted.

Name	Lower input limit	Upper input limit	Unit
Position	– 1,000,000,000	+ 1,000,000,000	MSR from MD7

Speed F

The input speed is calculated against the override. If the speed value is numerically greater than the maximum allowed speed, it is limited to the magnitude of the machine data item. Speeds are self-maintaining and need to be re-input only when changed.

Name	Lower input limit	Upper input limit	Unit
Speed	10	500 000 000	MSR from MD7/min

M functions

Up to three M functions can be programmed in one traversing block, with any assignment of M1, M2 and M3. The output sequence of the M functions is always M1→M2→M3 (for information about output see Section 9.1).

The following figure shows an example.

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	10	90	34	43	100 000	400 00	10	11	12	1

Table 10-2 M Functions

M No.	M Function	M Function group
0	Stop at end of block	1, 2, 3
2, 30	End of program	
1, 3...17	User functions	
18	Endless loop (skip back to start of program)	
19...29, 31...96	User functions	
97, 98	Change signal programmable as digital output	
99	User functions	

M0, M2, M18 and M30 are always output at the end of the traversing movement.

M0, M2, M18 and M30 are mutually exclusive within a single block.

Stop at end of block M0

If M no. 0 is programmed in a traversing block, the program stops at the end of the traversing block and M0 is output. Only a new START edge causes the traversing program to be continued.

End of program M2, M30

If M2 or M30 is programmed in a block, then after positioning is complete the M function is output with a subsequent programmed stop and a jump back to the start of the program. The Start edge can restart the program. M2 or M30 is always the last output in the block.

If the program is called up as a subprogram, the action skips to the main program. In this case M2 or M30 is not output.

Infinite loop M18

M18 is always output as the last M function in the block.

Two cases are distinguished:

- M function M18 is output like any other M function. Only after the block has been processed all the way to the end (including M18) does the axis skip back to the start of the program.
- If M function M18 is programmed alone in the last block of a traversing program, the M function is not output, and the axis immediately skips back to the start of the program.

Change signal programmable as digital output M97, M98

If M97 or M98 is programmed in a block, the M function output proceeds via the digital outputs as defined in machine data item MD35, in the same way as the checkback signals.

Tool offset number D

Twenty tool offset numbers (D1 – D20) are available. D0 in conjunction with G43 or G44 causes the tool offset to be switched off. The offset values must previously have been loaded to the module. Nonstandard offset values have a value 0.

Subprogram call P, L

A block with a subprogram call (P is the “number of calls”, L is the “program number”) cannot contain any further information.

Name	Lower input limit	Upper input limit
P = Number of subprogram callup	1	250

10.2 Program execution and direction of machining

Forward processing

As a rule, programs are processed by ascending block number.

Reverse processing

If programs are processed in reverse, the effects of commands must be taken into account in the programming:

- Commands are self-maintaining (G90, G91, G60, G64, G30...G39)
- Active tool offset (G43, G44, D0...D20)
- Change of coordinate systems via G87, G88, G89.

For these reasons, a distinction can be made between forward processing and reverse processing, in terms of both geometry and block transition response.

10.3 Block transitions

Overview

This chapter describes the influence of certain commands on block transitions.

Exact positioning - G60

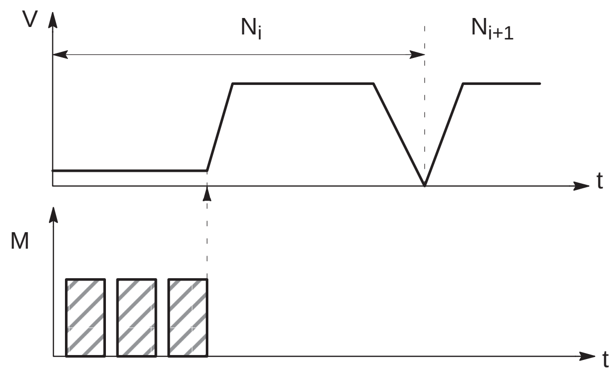
G60 mode is overlaid with G50 and G88 to G89 (force block change on-the-fly).

The program advances to the next block when the target range is reached.

The influence of M functions is as indicated in machine data item MD32.

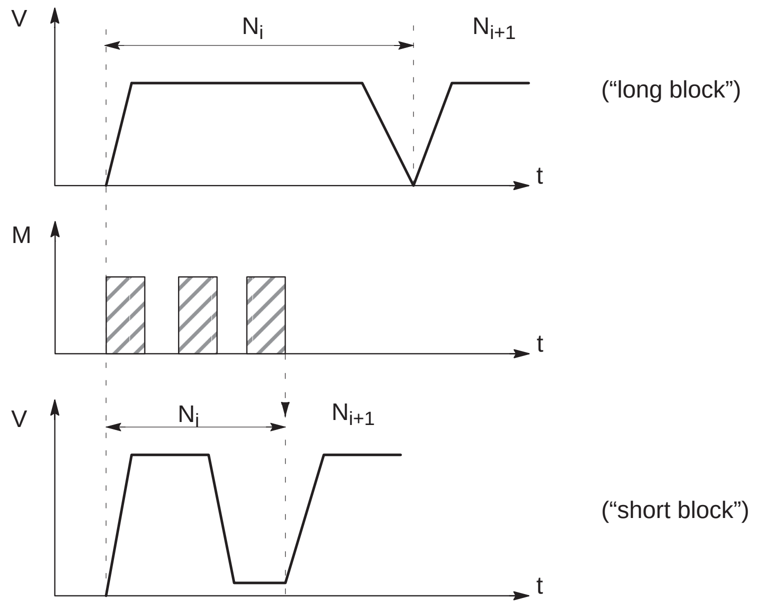
Output of M function before positioning

Case 1



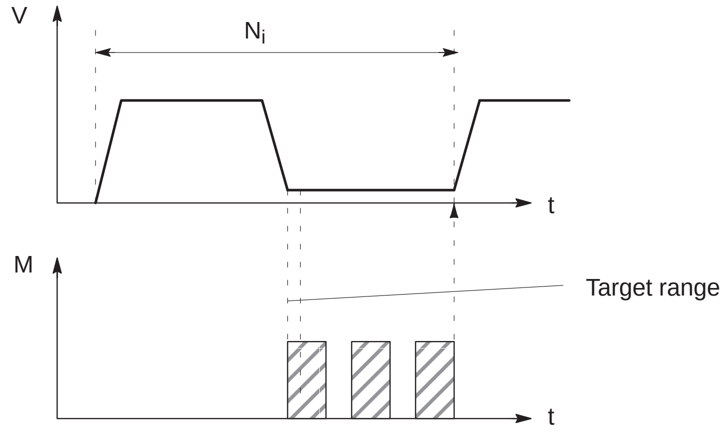
Output of M function during positioning

Case 2



Output of M function after positioning

Case 3



Change block on-the-fly - G64 (standard case)

Changing from one traversing block to the next proceeds without stopping the axis.

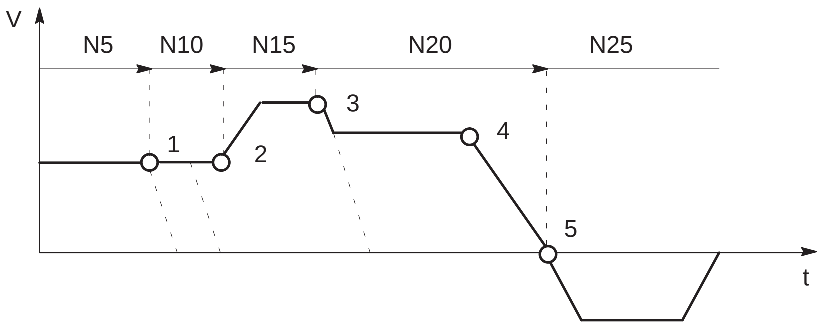
The acceleration and braking function is calculated for multiple blocks when the G64 function is programmed. The number of blocks processed in advance is three.

When the block changes, the feed rate is changed in such a way that a higher speed from a preceding block is never carried over into the next block, and a higher speed from a following block never goes into effect while a given block is still traversing its own path. This means that acceleration does not begin until the starting point of the block, and deceleration to a lower speed for a following block is initiated as with G60. When the speed of the following block is reached, the residual distance in the current block is processed at the feed rate of the following block.

Sample programming (standard case)

The following figure shows a sample program with the programming flow.

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	5	90		64	10 000	100 00				
	10				20 000					
	15				30 000	200 00				
	20				40 000	150 00				
	25			64	30 000	100 00				



1 – Block N10 is started at the point of deceleration of N5.

2 – N15 is started at the point of deceleration of N10. Acceleration to the higher traversing speed begins when the set position of N10is reached.

3 – N20 is started at a lower traversing speed at the braking point of N15.

4 – In a change of traversing direction, the axis brakes until it comes to a standstill and waits until the actual value of the encoder has reached the target range.

5 – When the target range is reached, the axis accelerates in the opposite direction up to the traversing speed of the new block.

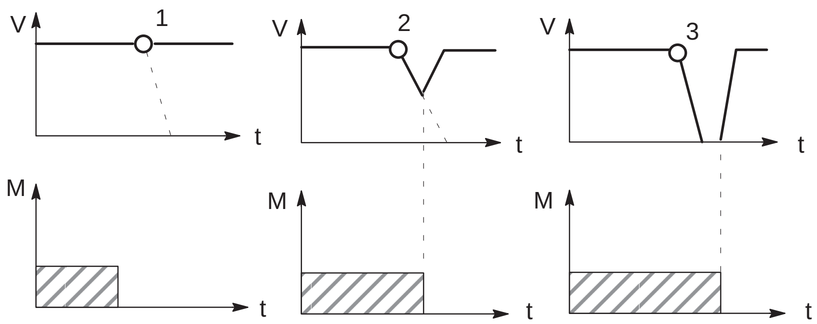
To be able to approach a position correctly, the axis must calculate the point of deceleration. The relevant parameters for this calculation are the residual traversing distance, the deceleration value and the current traversing speed.

The point of deceleration is also the earliest possible time for a block change.

Change block on-the-fly - G64 (deceleration)

There are a number of conditions that may delay or prevent a block change on-the-fly. Here a distinction is necessary between the case in which this type of block change is suppressed intentionally, and the case in which the selected function does not permit a block change on-the-fly.

- Block change on-the-fly is suppressed:
  - By removing the Enable read-in control signal - this stops program processing at the end of the current block. To continue the program, the enable must be re-input.
  - By output of the M function before or after positioning.
  - By M function M0 (stop at end of block). To continue the program, the START control signal must be reset.
  - By a block with a dwell time.
  - By processing a program in the Automatic/Single Block mode. Each block must be activated individually.
  - By a change in the acceleration override.
- Functions that themselves prevent block change on-the-fly:
  - M functions (during positioning).



1 – Since the M output is completed at the point of deceleration, a block change takes place on-the-fly.

2 – The M output is not yet complete at the point of deceleration. The axis begins to brake. At the end of the M output, the axis returns to speed (transition on-the-fly from deceleration ramp to acceleration ramp).

3 – Axis comes to a complete standstill and waits for the end of M output.

Influence of M function on block change on-the-fly

Machine data can specify the output time for M functions:

- M function is output before or after positioning with a block change

  M-function output and positioning proceed in alternation.

  – M function output before positioning causes exact-positioning response in the preceding block.

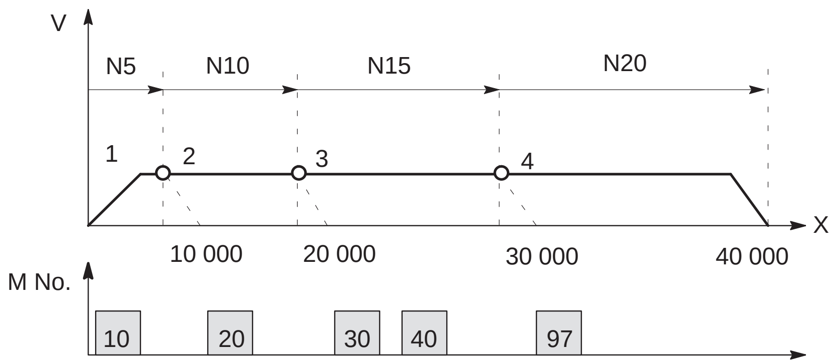
  – M function output after positioning causes exact-positioning response within the block.

- M function is output during positioning

  M-function output and positioning proceed simultaneously.

  The following figure shows a sample program with M function output “during positioning”.

/	N	G1	G2	G3	X/t	F	M1	M2	M3	D
	5	90			10 000	100 00	10			
	10				20 000		20			
	15				30 000	200 00	30	40		
	20			60	40 000	150 00			97	



1 – Output of M10 is **not** position-dependent, since no relevant position for a position-dependent M function is present.

2 – At the block change from N5 to N10, output is prepared. The axis the M function does not proceed until the actual position has reached the programmed position of N5.

3 – If two M functions are programmed in a traversing block, the first M function is output depending on position, followed by the second M function.

4 – The change signal for M97 or M98 is output with the G64 block transition (digital output) if the actual position has reached the programmed position of the block. The actual position runs behind the set position (difference = overtravel).

■

Troubleshooting

11

Overview

The FM 354 provides diagnostics for the following:

- I/Os
- Module processes

This chapter “Troubleshooting” describes the different types of errors, their cause, effect and elimination.

Error localization

The FM 354 distinguishes between

- Errors which trigger a diagnostic interrupt in the CPU, and
- Errors which the module reports by way of checkback messages.

In the event of diagnostic interrupt, STATUS LEDs light up in addition.

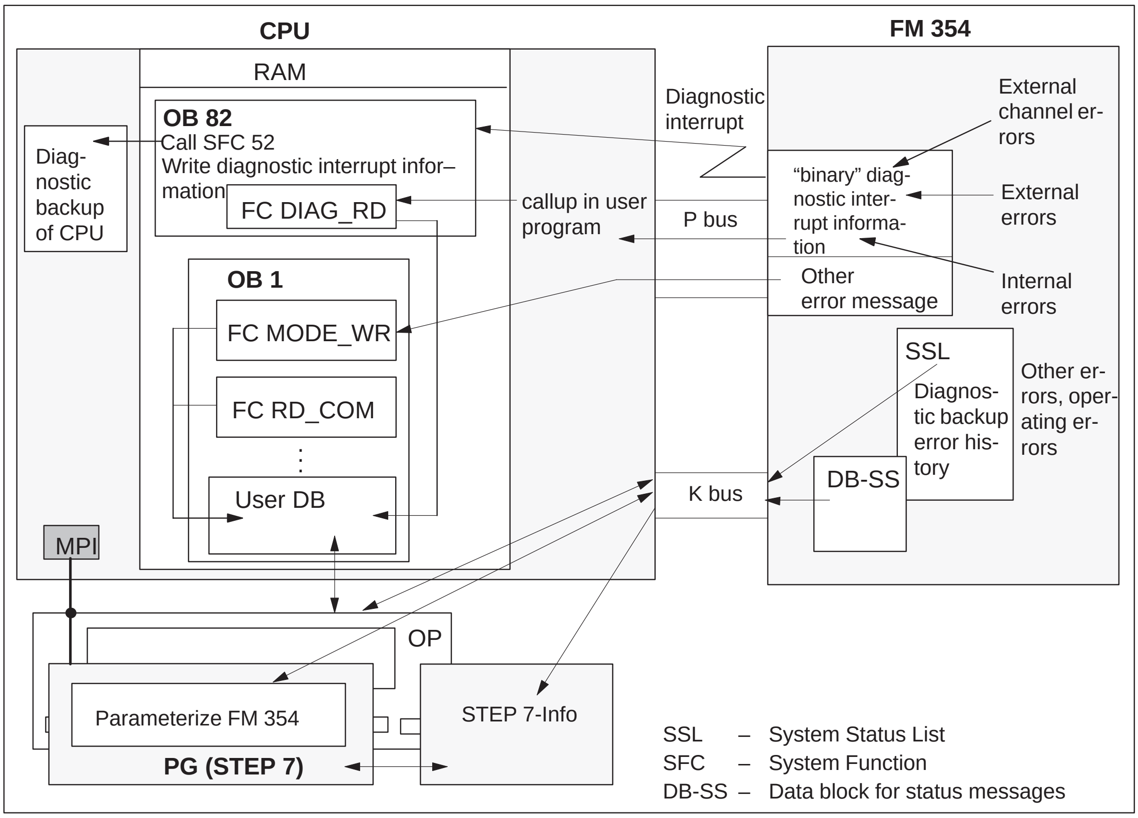


Fig. 11-1 Overview of diagnostics/errors

Error evaluation by software

The following manuals describe how to include diagnostics–capable modules in your user program, and how to evaluate the diagnostic messages:

- Programming manual *System Software for S7-300/400; Program Design* (OB Types, Diagnostic Interrupt OB 82)
- Reference manual *System Software for S7-300/400; System and Standard Functions*

A basic description of the diagnostic system of the S7-300 can be found in the user manual *Standard Software for S7 and M7, STEP 7*.

Chapter overview

11.1 Error classes and module responses

Overview

The FM 354 contains monitoring circuits which are active during startup or during continuous operation. Errors occurring during those times are reported to the system and to the user program.

The table below lists the error classes and their meaning.

Table 11-1 Error classes, overview

Message	Error Class	Response	Significance
Diagnostic interrupt	Internal errors	Everything OFF	...are hardware faults in the module which are discovered by diagnostics routines (e.g. memory errors). (see Section 6.4 for diagnostic interrupt data and error list, Table 11-4)
	External errors		...are errors which can occur due to faulty module connection (e.g. parameters for initialization of module MP station numbers are missing or are incorrect). (see Section 6.4 for diagnostic interrupt data and error list, Table 11-4)
	External channel errors		...are measurement system errors or errors which can occur by connecting the digital outputs or during operation (operating errors) of the FM 354 (e.g. cable break, incremental value encoder). (see Section 6.4 for diagnostic interrupt data and error list, Table 11-4 and 11-5)
Checkback signals	Operator control and travel errors	Feed STOP	... are errors (operator/travel errors) which can occur during “operation” of the FM 354 (e.g. direction signals R+ and R– set at the same time, see Error List, Tables 11-6 and 11-7).
	Data errors	Warning!	... are errors (data, machine data and traversing program errors) which are detected on interpretation of invalid data (see Error List, Table 11-8).

Error response

Each error message triggers an appropriate response.

Table 11-2 Overview of internal error responses

Error response	Significance
Everything OFF	• Stop movement by way of voltage ramp (MD45) • Disable digital outputs • Deactivate servo enable • SYN is cleared • No new travel jobs possible
Feed STOP	• Stop movement by controlled deceleration • Travel job is canceled and terminated. • Measured data acquisition and position control are continued. • No new travel jobs possible
Warning	• Message only • Movement and control of axes not affected

11.2 Error messages

Introduction

The following approaches to error localization are available for the FM 354:

- Error display by LEDs
- Error messages to the system and to the user program

11.2.1 Fault indication by LED

Status and error displays

The FM 354 features the following status and error displays:

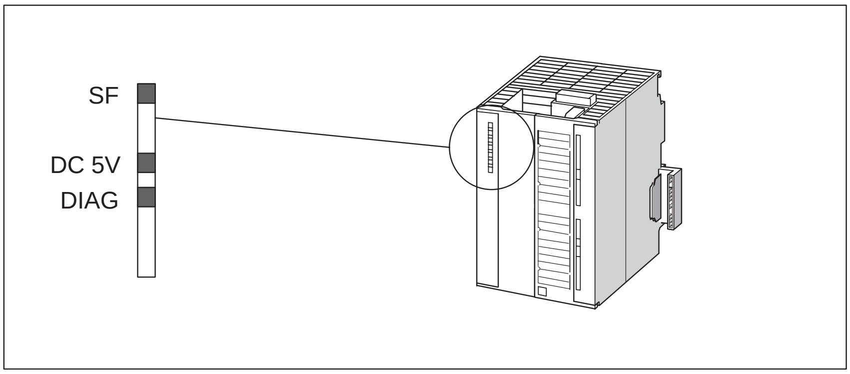


Fig. 11-2 Status and error displays of the FM 354

Significance of the status and error displays

The status and error displays are explained in the order in which they are arranged on the FM 354.

Table 11-3 Status and error displays

Display	Significance	Explanations
SF (red) LED-ON	Group errors	This LED indicates an error condition in the FM 354. Diagnostic interrupt (internal error, external error or external channel error) To eliminate the error see Error List, Table 11-4.
5 V DC (green)	Power supply switched on	This LED indicates that the hardware is ready for operation. If not illuminated, this may indicate one of the following conditions: • Your line voltage is not OK. • Load current supply is faulty. • Module is incorrectly connected. • Your control circuit is configured incorrectly (sum of rated and starting currents is too great), or • The module is faulty.
DIAG (yellow) LED-ON LED-blinking	Diagnostics	This LED indicates the various diagnostic conditions. Diagnostic interrupt (external error or external channel error). To eliminate the error see Error List, Table 11-4. If this LED blinks when the LED “SF” is simultaneously activated, this indicates a system error. If this occurs, then please consult the appropriate sales department. The exact circumstances which resulted in the error are of major importance in this case.

11.2.2 Diagnostic interrupts

Overview

Internal errors, external errors and external channel errors are indicated to an interrupt–capable system by means of diagnostic interrupts (see diagnostic interrupt data in Tables 11-4, 11-5). This presupposes that the diagnostic interrupt message was activated at the time of configuration (see Chapter 5). If the system is not interrupt–capable, the diagnostic interrupt information must be read out cyclically with FC 6.

The diagnostic interrupts are identified by setting the corresponding byte.bit no. (see diagnostic interrupt data, Section 6.4).

Error Class	Coding	Message
Internal errors	Byte-.Bit-No. 0.1 Group error byte 2, 3	LED “SF”
External errors	Byte-.Bit-No. 0.2	LED “SF” and “DIAG”
External channel errors	Byte-.Bit-No. 0.2, 0.3 Group error byte 8	LED “SF” and “DIAG”

The operator control errors in the “external channel error” class are specified again, and are stored in data block 164 or in the diagnostic buffer.

The FM 354 signals a diagnostic interrupt “incoming” or “outgoing.”

<table>
<tr><th colspan="6">Diagnostic interrupt</th></tr>
<tr><td colspan="4">Message to the CPU (precondition: interrupt message activated (see Section 5.2)</td><td rowspan="4">Message in the “troubleshooting” display of “Parameterize FM 354”
Menu: Test ▶ Error evaluation</td><td rowspan="4">Entry in diagnostic buffer</td></tr>
<tr><td rowspan="3">No OB 82 exists → CPU switches to STOP</td><td colspan="2">OB 82</td><td>OB 1</td></tr>
<tr><td rowspan="2">Enters the diagnostic information in the diagnostic buffer of the CPU (4 bytes) and calls SFC 52</td><td>Enters the diagnostic information in the user DB starting at address 72 and calls FC 4</td><td>Calls FC 6</td></tr>
<tr><td colspan="2">On operator control error:
(addr. in user DB 80.7)
for further error specification read DS 164 in OB 1</td></tr>
</table>

Alarm acknowledgement

If processing is to continue after a diagnostic interrupt, then Restart after the error has been remedied (see write job no. 11 Sections 6.2.1 and 9.3.3).

Internal errors cannot be acknowledged. External errors are self–acknowledging.

11.2.3 Error messages in checkback signals

Overview

Operator/travel errors [BF/FS] and data errors/machine data errors/traversing program errors [DF], are communicated to the user by way of checkback signals (FC 2 call). The error-specification is stored in the form of an error number (see error list in Table 11-6...11-8) in the corresponding data block (DS162 and DS163).

Checkback signals [BF/FS] and [DF] (group error messages)		
Error specification ...		
... in user program (if necessary)	... programming device/PC	
Read out DS 162 (on BF/FS) or read out DS 163 (on DF)	Message in the “troubleshooting” display of “Parameterize FM 354” Menu: **Test** ▶ **Alarms**	in diagnostic buffer

Error acknowledgement

Set/clear control signal [BFQ/FSQ]
or
on message [DF] → write a new write job

Note

Invalid data are not accepted. The original data are retained.

Meldung der Fehlernummer

If a specific error analysis is called for in the user program, then the error numbers can be read out by calling up the corresponding system function (SFC 59, see Reference Manual *Standard and System Functions)*), see also Section 6.7, user example 2).

Data records:

- DS162 (operator and travel errors)
- DS 163 (general data errors, machine data errors and traversing program errors)
- DS164 (operator control errors)

DS	Data block, structure	Significance	
162 163 164	2-byte error number	DEKL	DENR
	2 x 1 byte free	Free	Free

The errors are identified in DS 162 through 164 by the detail event class (DEKL) and by the detail event number (DENR).

In the case of operating errors within the error class “external channel errors” the message is displayed using the bit combination **0.0, 0.2, 0.3, 8.7** as a diagnostic interrupt (see Section 6.4) and additionally as DEKL and DENR.

Error technology class	DEKL	DENR	Message
Operating error	1	1...n	Diagnostic interrupt
Operator control errors	2	1...n	Checkback signals
Travel errors	3	1...n	Checkback signals
Data error	4	1...n	Checkback signals or Data block
Machine data errors	5	1...n	
Traversing program errors	8	1...n	

11.2.4 Message in data block

Overview

Please note the following for direct access to DBs (e.g. using an OP).

If data errors/machine data errors/traversing program errors are detected when the parameters are written to the data block (e.g. in the parameterization tool), an error message is stored in the data block. The error-specification is stored in the form of an error number in the corresponding data block (see error list in Table 11-8). The error message occurs each time the data block is written to until the cause has been eliminated.

It is recommended to scan the error message after every write operation.

11.2.5 Viewing the diagnostic buffer (PG/PC)

Overview

The last five error messages are stored in the diagnostic buffer.

There are two ways of proceeding:

1. In the **S7 SIMATIC Manager** select the menu **File** ▸ **Open** ▸ **Accessible Nodes**.
2. In the **Accessible Nodes** window, select the MPI address of your module.
3. You can view the diagnostic buffer in the menu **Target system** ▸ **Module status**.

or

1. Open your project in the **S7 SIMATIC Manager**.
2. Select the menu **View** ▸ **Online.**
3. In the dialog box, select the FM 354 and the associated program.
4. You can view the diagnostic buffer in the menu **Target system** ▸ **Module status**

11.3 Error lists

Note

In the following tables, please note:

The module response described under “Effect” refers to the error–specific module response. The error response described in Table 11-2 occurs in addition.

11.3.1 Diagnostic interrupts

Overview

The diagnostic interrupts are listed according to error class in Tables 11-4, 11-5.

Table 11-4 Diagnostic interrupt

Byte. Bit	Error message, error analysis and elimination		Message/ Display
0.1	**Internal errors**	**Error response: “Everything Off”, as in Table 11-2**	
2.1 (8031)	**Communication disturbance**		SF DC5V DIAG
	Cause	MPI/K-bus communication fault caused by unknown event	
	Effect		
	Elimination	• Check connection • Check programming device/CPU • Switch module on/off • Replace module	
2.3 (8033)	**Internal time monitoring circuit (Watchdog)**		
	Cause	• Pronounced noise conditions on the FM 354 • Errors in the FM 354	
	Effect	• Deactivate entire FM 354 • LED indicators: SF: On DIAG: Flashing cycle	
	Elimination	• If this manual is observed, the errors should not occur However, should this still be the case, please consult the responsible sales department. When doing so, it is vitally important to also report the exact circumstances leading to the error. • Replace the FM 354	

Note: (xxxx) value = Hexadecimal notation in diagnostic buffer

Table 11-4 Diagnostic interrupt, continued

Byte. Bit	Error message, error analysis and elimination		Message/ Display
0.1	**Internal errors**	**Error response: “Everything Off”, as in Table 11-2**	
2.4 (8034)	**Internal module power supply failure**		SF DC5V DIAG
	Cause	• Drastic voltage dip • FM 354 power supply faulty	
	Effect	Deactivate entire FM 354	
	Elimina-tion	• Check FM 354 power connection • If FM 354 power supply defective, replace FM 354	
3.2 (8042)	**FEPROM errors**		
	Cause	Memory for firmware code faulty	
	Effect		
	Elimina-tion	Replace the FM 354	
3.3 (8043)	**RAM errors**		
	Cause	• Faulty RAM data memory • Faulty flash-EPROM data memory	
	Effect		
	Elimina-tion	Replace the FM 354	
3.6 (8046)	**Process interrupt lost**		
	Cause	• A process interrupt event was detected by the FM 354 and cannot be reported, because the same event has not yet been acknowledged by the user program/CPU. • Faults on backplane bus	
	Effect		
	Elimina-tion	• Incorporate OB40 into user program • Check bus connection of the module • Deactivate using MD5 process interrupt • Switch module on/off	
0.2	**External errors**	**Error response: “Everything Off”, as in Table 11-2**	
0.6 (8006)	**Module not parameterized**		SF DC5V DIAG
	Cause	No MPI parameterization data received from the CPU	
	Effect	MPI interface is initialized with default values. MPI address = 12	
	Elimina-tion	Check parameterization, see S7-300 description	

Note: (xxxx) value = Hexadecimal notation in diagnostic buffer

Table 11-4 Diagnostic interrupt, continued

Byte. Bit	Error message, error analysis and elimination		Message/ Display
0.2, 0.3	**External channel errors**	**Error response: “Everything Off”, as in Table 11-2**	
8.0 (8090)	**Cable break, incremental encoder**		SF DC5V DIAG
	Cause	• Measurement system cable not plugged in or sheared off • Encoder without internode signals • Incorrect pin connection • Cable too long	
	Effect		
	Elimina- tion	• Check encoder and measurement system cable • Observe limit values • Using the MD20, monitoring can be temporarily skipped, at the responsibility of the owner/operator.	
8.1 (8091)	**Error, absolute encoder**		
	Cause	Telegram traffic between FM 354 and the absolute encoder (SSI) is faulty or is disrupted: • Measurement system cable not plugged in or sheared off • Unauthorized type of encoder (only allowable per MD10) • Encoder incorrectly set (programmable encoder) • Telegram length (MD13, MD14) incorrectly specified • Encoder delivers erroneous values • Noise interference on measurement system cable • Baud rate set too high (MD15)	
	Effect		
	Elimina- tion	• Check encoder and measurement system cable • Check telegram traffic between encoder and FM 354 • Using the MD20, monitoring can be temporarily skipped, at the responsibility of the owner/operator.	

Note: (xxxx) value = Hexadecimal notation in diagnostic buffer

Table 11-4 Diagnostic interrupt, continued

Byte. Bit	Error message, error analysis and elimination		Message/ Display
0.2, 0.3	**External channel errors**	**Error response: “Everything Off”, as in Table 11-2**	
8.2 (8092)	**Erroneous pulses, incremental encoder or zero reference mark missing**		SF DC5V DIAG
	Cause	• Encoder monitoring circuit has discovered erroneous pulses • In “reference point operation” operating mode, no zero reference mark came within one encoder revolution after the reference point switch was passed. • Number of pulses per encoder revolution (MD13) incorrectly input • Encoder faulty: does not deliver the specified number of pulses • Zero reference mark faulty or missing altogether • Pulse length of zero reference mark shorter than 1.25 µs • Interference on the measurement system cable	
	Effect		
	Elimina-tion	• Enter MD13 correctly • Check encoder and measurement system cable • Observe limit values • Observe rules on shielding and grounding • Using the MD20, monitoring can be temporarily skipped, at the responsibility of the owner/operator.	
8.3 (8093)	**Voltage monitoring, encoder**		
	Cause	• Short-circuit in encoder supply cable (5 V incrementally, 24 V SSI) • Failure of module internal encoder supply unit	
	Effect		
	Elimina-tion	• Check connections • Replace FM 354 if encoder cable ok • Using the MD20, monitoring can be temporarily skipped, at the responsibility of the owner/operator.	
8.7 (8097)	**For operator control errors, see Table 11-5**		

Note: (xxxx) value = Hexadecimal notation in diagnostic buffer

Table 11-5 Operator control errors

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Operator control errors		**Error response: “Everything Off”, as in Table 11-2**		
1 (01)	**1 (01)**	**Software limit switch, beginning is passed**		Diagnostic interrupt
		Cause	Limit switch passed: in “Control” or “Correction” operating mode	
		Effect	• The limit switch position is passed by the necessary stopping distance. • Set actual value is not executed.	
		Elimina-tion	• Following acknowledgment of the error, it is possible to tra-verse to the working range. • Alter value of software limit switch (MD21) • Disable limit switch monitoring! (With the limit switches (MD21/22) disabled, the travel range limits are established by the maximum allowable values for the limit switches).	
1 (01)	**2 (02)**	**Software limit switches, end is passed**		Diagnostic interrupt
		Cause	Limit switch passed: in “Control” or “Correction” operating mode	
		Effect	• The limit switch position is passed by the necessary stopping distance. • Set actual value is not executed.	
		Elimina-tion	• Following acknowledgment of the error, it is possible to tra-verse to the working range. • Alter value of software limit switch (MD22) • Disable limit switch monitoring! (With the limit switches (MD21/22) disabled, the travel range limits are established by the maximum allowable values for the limit switches).	
1 (01)	**3 (03)**	**Beginning of traversing range passed**		Diagnostic interrupt
		Cause	When operating in “Control” operating mode with soft limits dis-abled, the traversing range beginning was passed.	
		Effect	The limit switch position is passed by the necessary stopping dis-tance.	
		Elimina-tion	Following acknowledgment of the error, it is possible to traverse to the working range.	
1 (01)	**4 (04)**	**Traversing range end passed**		Diagnostic interrupt
		Cause	When operating in “Control” operating mode with soft limits dis-abled, the traversing range beginning was passed.	
		Effect	The limit switch position is passed by the necessary stopping dis-tance.	
		Elimina-tion	Following acknowledgment of the error, it is possible to traverse to the working range.	

Cl. = Detail event class, No. = Detail event number

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-5 Operator control errors, continued

<table>
<tr><th>Cl.</th><th>No.</th><th colspan="2">Error message,
error analysis and elimination</th><th>Message/
Display</th></tr>
<tr><td colspan="5">Operator control errors Error response: “Everything Off”, as in Table 11-2</td></tr>
<tr><td rowspan="4">1 (01)</td><td rowspan="4">11 (0B)</td><td colspan="2">Drive, direction of rotation</td><td rowspan="4">Diagnostic interrupt</td></tr>
<tr><td>Cause</td><td>Drive turns in wrong direction</td></tr>
<tr><td>Effect</td><td></td></tr>
<tr><td>Elimina-
tion</td><td>• Check drive
• Check or correct MD19
• Following “Restart” continue working using the user program</td></tr>
<tr><td rowspan="4">1 (01)</td><td rowspan="4">12 (0C)</td><td colspan="2">Stoppage area</td><td rowspan="4">Diagnostic interrupt</td></tr>
<tr><td>Cause</td><td>The zero speed control range was left when the servo enable was deactivated or when an axis standstill was reached in the PEH target range</td></tr>
<tr><td>Effect</td><td></td></tr>
<tr><td>Elimina-
tion</td><td>• Check electrical and mechanical drive disable (terminals, connecting cables, control element functions)
• Match MD26</td></tr>
<tr><td rowspan="4">1 (01)</td><td rowspan="4">90...99
(5A...63)</td><td colspan="2">System errors</td><td rowspan="4">Diagnostic interrupt
LED
“DIAG”
blinking</td></tr>
<tr><td>Cause</td><td>Internal errors in the module</td></tr>
<tr><td>Effect</td><td>Undefined effects possible</td></tr>
<tr><td>Elimina-
tion</td><td>If this manual is observed, the errors should not occur
However, should this still be the case, please consult the responsible sales department. When doing so, it is vitally important to also report the exact circumstances leading to the error.</td></tr>
</table>

Cl. = Detail event class, No. = Detail event number

Note: Value (xx) = Hexadecimal notation of the error number

11.3.2 Error messages in checkback signals

Overview

The errors are listed in Tables 11-6...11-8 according to error class.

Table 11-6 Operator control errors

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Operator control errors		**Error response : "Feed STOP" see Table 11-2**		
2 (02)	**1 (01)**	**Operating mode not allowed**		CBS
		Cause	The operating mode selected is not allowed.	
		Effect		
		Elimina-tion	Select an allowed operating mode	
2 (02)	**4 (04)**	**Incorrect operating mode parameters**		CBS
		Cause	In the "Jogging" and "Control" operating modes, the selected velocity or control level is not 1 or 2. In incremental operation the set value number is not allowed (1 - 100, and 254 permitted).	
		Effect		
		Elimina-tion	Set operating mode parameters to an allowable value.	
2 (02)	**5 (05)**	**Start enable missing**		CBS
		Cause	A travel command was given in the absence of a start enable (start, external start, R+/R–)	
		Effect		
		Elimina-tion	Restore travel command and wait for start enable	
2 (02)	**9 (09)**	**Axis is not synchronized**		CBS
		Cause	Synchronization of the axes is necessary in the "Incremental, relative," "MDI" and "Automatic" operating modes.	
		Effect		
		Elimina-tion	Execute reference point approach	
2 (02)	**11 (0B)**	**Direction specification not allowed**		CBS
		Cause	In operating modes "Jog," "Control" or "Incremental, relative" the direction settings R+/R- are active at the same time. With "Reference point operation" the direction setting no longer agrees with the startup direction specified in the MD.	
		Effect		
		Elimina-tion	Correct the direction parameters	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-6 Operator control errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Operator control errors		**Error response : “Feed STOP” see Table 11-2**		
2 (02)	**12 (0C)**	**Axis movement not possible**		CBS
		Cause	With an unacknowledged error, no drive enable or stop, a traverse command was triggered.	
		Effect		
		Elimina-tion	Restore traverse command and switch acknowledge error or Stop to inactive, or give drive enable.	
2 (02)	**13 (0D)**	**Incremental value not in place**		CBS
		Cause	The setpoints defined by the operating mode parameters are missing or a change in incremental dimensions occurred when the operating mode started.	
		Effect		
		Elimina-tion	Parameterize and read in setpoint parameters	
2 (02)	**14 (0E)**	**No program preselected**		CBS
		Cause	No program preselected at “Start.”	
		Effect		
		Elimina-tion	First preselect program then start.	
2 (02)	**15 (0F)**	**Digital input not activated**		CBS
		Cause	The programmed target was reached in a block with external block change (G50).	
		Effect		
		Elimina-tion	Check programming (MD34) and connection of digital input.	
2 (02)	**16 (10)**	**Measurement function undefined**		CBS
		Cause	Length measurement and inprocess measurement selected simultaneously	
		Effect	No measurement function effective.	
		Elimina-tion	Reselect one of the two measurement functions.	
2 (02)	**21 (15)**	**Activate machine data not allowed**		CBS
		Cause	“Processing in progress” is still active	
		Effect	Activate machine data not executed	
		Elimina-tion	Terminate processing, repeat activation	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-7 Travel errors

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Travel errors		**Error response : “Feed STOP” see Table 11-2**		
3 (03)	**1 (01)**	**Software limit switch, beginning**		CBS
		Cause	Limit switch approached: in “Jog” operating mode, in “Automatic” operating mode if G88/89, without switching signal from the corresponding digital input. The axis is located to the left of the software limit switch because of actual value set.	
		Effect	• Axis movement is stopped at the limit switch position. • Set actual value is not executed.	
		Elimina-tion	• Following acknowledgment of the error, it is possible to traverse to the working range. • Alter value of software limit switch (MD21) • Disable limit switch monitoring! (With the limit switches (MD21/22) disabled, the travel range limits are established by the maximum allowable values for the limit switches).	
3 (03)	**2 (02)**	**Software limit switch, end**		CBS
		Cause	Limit switch approached: in “Jog” operating mode, in “Automatic” operating mode if G88/89, without switching signal from the corresponding digital input. The axis is located to the right of the software limit switch because of actual value set.	
		Effect	• Axis movement is stopped at the limit switch position. • Set actual value is not executed.	
		Elimina-tion	• Following acknowledgment of the error, it is possible to traverse to the working range. • Alter value of software limit switch (MD22) • Disable limit switch monitoring! (With the limit switches (MD21/22) disabled, the travel range limits are established by the maximum allowable values for the limit switches).	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-7 Travel errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Travel errors		**Error response : "Feed STOP" see Table 11-2**		
3 (03)	**3 (03)**	**Traversing range beginning approached**		CBS
		Cause	• During traversing with soft limit switches disabled, the traversing range beginning was approached. • The axis is located to the left of the traversing range beginning because of actual value set. (Traversing range: $\pm 10^9$ or from range covered by absolute encoder)	
		Effect	• Axis movement is stopped at the traversing range limit. • Set actual value is not executed.	
		Elimination	Travel in the opposite direction	
3 (03)	**4 (04)**	**Traversing range end approached**		CBS
		Cause	• During traversing with soft limit switches disabled, the traversing range end was approached. • The axis is located to the left/right of the traversing range end because of actual value set. (Traversing range: $\pm 10^9$ or from range covered by absolute encoder)	
		Effect	• Axis movement is stopped at the traversing range limit. • Set actual value is not executed.	
		Elimination	Travel in the opposite direction	
3 (03)	**5 (05)**	**Target position not within traversing range**		CBS
		Cause	• The position to be approached is outside the working range limited by the software limit switches. • The rotary axis programming is specified as a reference value which does not fall within the positive complete circle.	
		Effect		
		Elimination	• Correct position to be approached. • Alter value of software limit switch (MD) • Disable limit switch monitoring! (With the limit switches (MD21/22) disabled, the travel range limits are established by the maximum allowable values for the limit switches).	
3 (03)	**23 (17)**	**Target velocity zero**		CBS
		Cause	• Zero was entered as programmed velocity. • No feed was programmed for positioning.	
		Effect		
		Elimination	Input an allowable velocity value	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-7 Travel errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Travel errors		**Error response : “Feed STOP” see Table 11-2**		
3 (03)	**28 (1C)**	**M2/M30 missing**		CBS
		Cause	• In the last program, block, no M2, M30 or M18 is programmed. • The last program, block is a skip block.	
		Effect		
		Elimina-tion	Per causes	
3 (03)	**30 (1E)**	**Digital input not parameterized**		CBS
		Cause	For traversing with set actual value on the fly (G88, G89), external block change (G50) or measurement, no digital input necessary for that purpose is parameterized.	
		Effect	The functions are not started.	
		Elimina-tion	Parameterize the digital inputs by way of MD34.	
3 (03)	**35 (23)**	**Tool offset value not in place**		CBS
		Cause	No tool offset values are available on the FM 354 or tool offsets are accessed and modified when an override is active.	
		Effect		
		Elimina-tion	Parameterize and read in tool offset values	
3 (03)	**36 (24)**	**Set actual value on the fly, incorrect value**		CBS
		Cause	Value is no longer within the range $\pm 10^9$	
		Effect		
		Elimina-tion	Input a correct value	
3 (03)	**37 (25)**	**MDI-block on the fly, incorrect syntax**		CBS
		Cause	Incorrect M or G commands or incorrect block structure	
		Effect		
		Elimina-tion	Input a correct MDI block	
3 (03)	**38 (26)**	**MDI block on the fly, incorrect velocity**		CBS
		Cause	Velocity not within the range between > 0 and max. allowable traverse velocity (500,000,000 MSR/min)	
		Effect		
		Elimina-tion	Input a correct MDI block	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-7 Travel errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Travel errors		**Error response : “Feed STOP” see Table 11-2**		
3 (03)	**39 (27)**	**MDI block on the fly, incorrect position or dwell time**		CBS
		Cause	Position or dwell time is outside allowable values. Position: $\pm 10^9$ MSR Dwell time: > 100,000 ms	
		Effect		
		Elimina-tion	Input a correct MDI block	
3 (03)	**40 (28)**	**MDI block on the fly erroneous**		CBS
		Cause	Incorrect block syntax	
		Effect		
		Elimina-tion	Input a correct MDI block	
3 (03)	**61 (3D)**	**Controller enable missing**		CBS
		Cause	Traverse command of the axis without controller enable (except for “Control” operating mode) or Removal of controller enable during “Processing in progress”	
		Effect	No axis movement or Axis stopped (at same time, controller enable is held until axis comes to rest)	
		Elimina-tion	Set controller enable by way of user program	
3 (03)	**62 (3E)**	**Controller not ready for operation**		CBS
		Cause	Axis started without “Controller ready message” or “Controller ready message” canceled whilst “processing in progress”.	
		Effect	No axis movement or Axis is stopped with actual value transfer after axis comes to rest (internally like “follow-up)	
		Elimina-tion	• Check drive/connecting cables • Analysis of the “Controller ready” message can be disabled by MD37!	
3 (03)	**64 (40)**	**PEH target area monitoring**		CBS
		Cause	Following conclusion of the setpoint value specification to the position controller, the target area is not reached within the specified time.	
		Effect		
		Elimina-tion	• Check drive • Match MD24, MD25	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-7 Travel errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Travel errors		**Error response : “Feed STOP” see Table 11-2**		
3 (03)	**65 (41)**	**No drive movement**		CBS
		Cause	• Axis standstill at maximum drive control signal (±10 V) • on violation of the defined following error limit	
		Effect	Acceptance of actual value (internally like “follow–up mode”)	
		Elimina-tion	• Check drive/connecting cables • Check controller enable signal between FM 354 and drive	
3 (03)	**66 (42)**	**Following error too great**		CBS
		Cause	Excessive following error during axis movement	
		Effect		
		Elimina-tion	• Check drive • Check MD23, MD43	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block
Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: “Warning” see Table 11-2**		
4 (04)	**1 (01)**	**Data at time of transmission unacceptable**		CBS or DB
		Cause	Data not transmitted in appropriate operating mode	
		Effect	Data not accepted	
		Elimina-tion	Transmit data in appropriate operating mode	
4 (04)	**2 (02)**	**Velocity level 1 incorrect**		CBS or DB
		Cause	Velocity not within the range between > 0 and max. allowable traverse velocity (500,000,000 MSR/min)	
		Effect	Velocity does not become effective	
		Elimina-tion	Input an allowed velocity value	
4 (04)	**3 (03)**	**Velocity level 2 incorrect**		CBS or DB
		Cause	Velocity not within the range between > 0 and max. allowable traverse velocity (500,000,000 MSR/min)	
		Effect	Velocity does not become effective	
		Elimina-tion	Input an allowed velocity value	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block
Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: "Warning" see Table 11-2**		
4 (04)	**4 (04)**	**Voltage level 1 incorrect**		CBS or DB
		Cause	Voltage specified does not fall within the range of $\pm$10 V	
		Effect	Voltage does not become effective	
		Elimina-tion	Input an allowed voltage value	
4 (04)	**5 (05)**	**Voltage level 2 incorrect**		CBS or DB
		Cause	Voltage specified does not fall within the range of $\pm$10 V	
		Effect	Voltage does not become effective	
		Elimina-tion	Input an allowed voltage value	
4 (04)	**6 (06)**	**Preset incremental value too high**		CBS or DB
		Cause	Incremental value is greater than 10^9 MSR	
		Effect	Original incremental value is retained	
		Elimina-tion	Input an allowable incremental value	
4 (04)	**7 (07)**	**MDI block, incorrect syntax**		CBS or DB
		Cause	Incorrect M or G commands or incorrect block structure	
		Effect	Original MDI block is retained	
		Elimina-tion	Input a correct MDI block	
4 (04)	**8 (08)**	**MDI block, incorrect velocity**		CBS or DB
		Cause	Velocity not within the range between > 0 and max. allowable traverse velocity (500,000,000 MSR/min)	
		Effect	Original MDI block is retained	
		Elimina-tion	Input a correct MDI block	
4 (04)	**9 (09)**	**MDI block, position or dwell time incorrect**		CBS or DB
		Cause	Position or dwell time falls outside the allowable values Position: $\pm 10^9$ MSR Dwell time: > 100,000 ms	
		Effect	Original MDI block is retained	
		Elimina-tion	Input a correct MDI block	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: "Warning" see Table 11-2**		
4 (04)	**10 (0A)**	**Zero offset value, offset value incorrect**		CBS or DB
		Cause	Value falls outside the range $\pm 10^9$ MSR	
		Effect	Does not become effective	
		Elimina-tion	Input a correct value	
4 (04)	**11 (0B)**	**Set actual value, actual value incorrect**		CBS or DB
		Cause	Actual value falls outside the software limit switches or outside the range $\pm 10^9$ MSR	
		Effect	Set actual value does not become effective	
		Elimina-tion	Input a correct value	
4 (04)	**12 (0C)**	**Set reference point value, reference point incorrect**		CBS or DB
		Cause	Value falls outside the range $\pm 10^9$ MSR	
		Effect	Set reference point does not become effective	
		Elimina-tion	Input a correct value	
4 (04)	**13 (0D)**	**Digital output not possible**		CBS or DB
		Cause	Output not available for direct output of the user program	
		Effect	Output is not executed	
		Elimina-tion	• Correct user program • Correct parameterization of the output assignment within the MD35 to the desired assignment	
4 (04)	**14 (0E)**	**Request application data incorrect**		CBS or DB
		Cause	Incorrect request code	
		Effect	Old application data are retained	
		Elimina-tion	Request code 0-6, 16-23 and 25 possible	
4 (04)	**15 (0F)**	**Teach In, program number incorrect**		CBS or DB
		Cause	The program was not parameterized or read in.	
		Effect	Teach In is not executed	
		Elimina-tion	Parameterize and read in program or correct program number	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: "Warning" see Table 11-2**		
4 (04)	**16 (10)**	**Teach In, block number incorrect**		CBS or DB
		Cause	The block number in the program selected is not in place.	
		Effect	Teach In is not executed	
		Elimina-tion	Specify correct block number	
4 (04)	**17 (11)**	**Teach In, dwell time or subprogram-request in block**		CBS or DB
		Cause	The block number in the program selected is not in place or incorrect block number was selected.	
		Effect	Teach In is not executed	
		Elimina-tion	Specify correct block number	
4 (04)	**18 (12)**	**Teach In, no axis stoppage**		CBS or DB
		Cause	Axis is still in motion	
		Effect	Teach In is not executed	
		Elimina-tion	Stop axis and repeat task	
4 (04)	**40 (28)**	**Transmit non-relevant data**		CBS or DB
		Cause	The data (data blocks) transmitted are unknown to the FM 354	
		Effect	Data not accepted	
		Elimina-tion	Correct user program	
4 (04)	**81 (51)** **82 (52)** **83 (53)** **84 (54)** **85 (55)**	**Programmable modules communication: unauthorized DB type** **Programmable modules communication: Info 1 incorrect** **Programmable modules communication: Info 2 incorrect** **Programmable modules communication: unauthorized task** **Programmable modules communication: data errors**		CBS or DB
		Cause	Incorrect data	
		Effect	Task is not executed	
		Elimina-tion	Correct and retransmit	
4 (04)	**120 (78)**	**Measurement system grid deviates**		CBS or DB
		Cause	The measurement system in the DBs "NC, SM, TO" does not agree with MD7.	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: "Warning" see Table 11-2**		
4 (04)	**121 (79)**	**Incorrect DB type in the module**		CBS or DB
		Cause	An incorrect type of DB has been transmitted into the FM 354	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Delete DB, correct and retransmit	
4 (04)	**122 (7A)**	**DB type or DB no. already exists**		CBS or DB
		Cause	DB type already exists	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Delete corresponding DB prior to transmission	
4 (04)	**123 (7B)**	**NC program number already exists**		CBS or DB
		Cause	NC program number already exists	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Prior to transmission, delete corresponding DB with the program number	
4 (04)	**124 (7C)**	**Parameter "Save" incorrect**		CBS or DB
		Cause	Coding not 0 or 1	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Coding not 0 or 1	
4 (04)	**125 (7D)**	**DB memory filed**		CBS or DB
		Cause	The available memory is assigned	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Delete unnecessary programs (DBs) or compress memory by way of parametering interface	
4 (04)	**126 (7E)**	**Allowable program length exceeded**		CBS or DB
		Cause	Number of blocks too high	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct program and retransmit	
4 (04)	**127 (7F)**	**Writing parameters/data is not possible**		CBS or DB
		Cause	Axis does not come to a stop	
		Effect	Parameters/data do not become effective	
		Elimina-tion	Stop axis	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
General data errors		**Error response: "Warning" see Table 11-2**		
4 (04)	**128 (80)**	**Incorrect module identification**		CBS or DB
		Cause	DBs which do not belong to the module were transmitted (no identification 354)	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Transmit the DBs belonging to the FM 354	
4 (04)	**129 (81)**	**Incremental value, incorrect value**		CBS or DB
		Cause	Value range outside $\pm 10^9$	
		Effect	Incremental value not effective	
		Elimina-tion	Transmit correct value	
4 (04)	**130 (82)**	**Tool offset, incorrect value**		CBS or DB
		Cause	Value range outside $\pm 10^9$	
		Effect	Tool offset not effective	
		Elimina-tion	Transmit correct value	
4 (04)	**131 (83)**	**Not possible to insert block**		CBS or DB
		Cause	Memory full	
		Effect	Function is not executed	
		Elimina-tion	Delete unnecessary DBs and repeat function	
4 (04)	**132 (84)**	**Not possible to delete block**		CBS or DB
		Cause	Block does not exist, no "assignment bits (bytes 2 and 3) enabled in block (when data available).	
		Effect	Function is not executed	
		Elimina-tion	Check program and repeat function with correct block number	
Machine data errors		**Error response: "Warning" see Table 11-2**		
5 (05)	**7 (07)**	**Measurement system**		CBS or DB
		Cause	The measurement system grid (MSR) entered does not agree with the MSR in the other DBs of the module.	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	• Check MSR and correct as necessary • When making correct input, delete the other DBs on the module before retransmitting.	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Machine data errors		**Error response: "Warning" see Table 11-2**		
5 (05)	**8 (08)**	**Type of axis**		CBS or DB
		Cause	No linear or rotary axis parameterized	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**9 (09)**	**Rotary axis**		CBS or DB
		Cause	Impermissible value range or dependency violation (see Section 5.3.1)	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**10 (0A)**	**Encoder type**		CBS or DB
		Cause	Unacceptable type of encoder	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**11 (0B)** **12 (0C)** **13 (0D)** **14 (0E)**	**Travel per encoder revolution** **Distance to go per encoder revolution** **Increments per encoder revolution** **Number of revolutions, absolute encoder**		CBS or DB
		Cause	Impermissible value range or dependency violation on no. 11, 12, 13 (see Section 5.3.1)	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**15 (0F)**	**Baud rate, absolute encoder**		CBS or DB
		Cause	Unacceptable baud rate	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**16 (10)** **17 (11)**	**Reference point coordinates, absolute encoder adjustment**		CBS or DB
		Cause	Unacceptable value range	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Machine data errors		**Error response: “Warning” see Table 11-2**		
5 (05)	**18 (12)**	**Type of reference point travel**		CBS or DB
		Cause	Unacceptable type of reference point travel	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**19 (13)**	**Direction matching undefined**		CBS or DB
		Cause	Direction matching undefined	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**20 (14)**	**Disable hardware monitoring undefined**		CBS or DB
		Cause	Disable hardware monitoring undefined	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**21 (15)** **22 (16)** **23 (17)** **24 (18)** **25 (19)** **26 (1A)** **27 (1B)** **28 (1C)** **29 (1D)** **30 (1E)**	**Software limit switch, begin** **Software limit switch, end** **Maximum velocity** **Target range (PEH)** **Monitoring time** **Stoppage area** **Reference point offset** **Referencing velocity** **Reducing velocity** **Backlash compensation**		CBS or DB
		Cause	Impermissible value range or dependency violation on no. 21, 22, 28, 29 (see Section 5.3.1)	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**31 (1F)**	**Backlash vector reference**		CBS or DB
		Cause	Backlash vector reference undefined	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Machine data errors		**Error response: "Warning" see Table 11-2**		
5 (05)	**32 (20)**	**Type of output, M-function**		CBS or DB
		Cause	Type of output, M-function not defined	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**33 (21)**	**Output time, M-function**		CBS or DB
		Cause	Unacceptable value range	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**34 (22)**	**Digital inputs**		CBS or DB
		Cause	Inputs undefined or defined more than once	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**35 (23)**	**Digital outputs**		CBS or DB
		Cause	Outputs undefined or defined more than once	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**36 (24)**	**Input adapter**		CBS or DB
		Cause	Input adapter undefined	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**38 (26)** **39 (27)** **40 (28)** **41 (29)** **42 (2A)** **43 (2B)** **44 (2C)** **45 (2D)**	**Positioning circuit amplification** **Minimum following error, dynamic** **Speed-up** **Slow-down** **Jerk time** **Set voltage, max.** **Offset compensation** **Voltage ramp**		CBS or DB
		Cause	Unacceptable value range	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Machine data errors		**Error response: “Warning” see Table 11-2**		
5 (05)	**96 (60)**	**Software limit unacceptable**		CBS or DB
		Cause	With linear axes: Software begin limit switch greater than software limit switch end with rotary axes: Software begin/end limit switches not within rotary axis cycle and not at maximum input value.	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
5 (05)	**97 (61)**	**Limitation, software limit with absolute encoder**		CBS or DB
		Cause	Travel distance between software limit switch begin and end is greater than the absolute value range of the encoder.	
		Effect	DB does not become effective and is stored non-retentively	
		Elimina-tion	Correct and retransmit	
Traversing program errors		**Error response: “Warning” see Table 11-2**		
8 (08)	**1 (01)**	**Program selection, subroutine error**		CBS or DB
		Cause	The subroutine requested in the program is not in place on the FM 354.	
		Effect	Program selection is not executed	
		Elimina-tion	• Parameterize and read in program, correct as necessary • Select another program	
8 (08)	**8 (08)**	**Program selection, program number not in place**		CBS or DB
		Cause	The program was not parameterized, not in place on the FM 354.	
		Effect	Program selection is not executed	
		Elimina-tion	• Parameterize and read in program, correct as necessary • Select another program	
8 (08)	**9 (09)**	**Program selection, block number missing**		CBS or DB
		Cause	The block number is missing in the program selected	
		Effect	Program selection is not executed	
		Elimina-tion	• Correct program • Select different block number	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Traversing program errors Error response: "Warning" see Table 11-2				
8 (08)	**10 (0A)**	**Program, block number unacceptable**		CBS or DB
		Cause	Block number missing or outside of the number range	
		Effect	Program is not stored	
		Elimination	Correct program	
8 (08)	**11 (0B)**	**Program selection, direction specification incorrect**		CBS or DB
		Cause	Direction specification incorrect	
		Effect	Program selection is not executed	
		Elimination	Correct program selection and repeat.	
8 (08)	**12 (0C)**	**Program selection unacceptable**		CBS or DB
		Cause	Another program was preselected during a movement	
		Effect	Program selection is not executed	
		Elimination	Use STOP to stop program in progress, or repeat program selection at end of program.	
8 (08)	**20 (14)**	**Error, program number**		CBS or DB
		Cause	Program numbers in the blocks incorrect	
		Effect	Program is not stored	
		Elimination	Correct program, per cause	
8 (08)	**21 (15)**	**No block in program**		CBS or DB
		Cause	No block in program	
		Effect	Program is not stored	
		Elimination	Correct program, per cause	
8 (08)	**22 (16)**	**Error, block number**		CBS or DB
		Cause	Block number value range incorrect	
		Effect	Program is not stored	
		Elimination	Correct program	
8 (08)	**23 (17)**	**Block number sequence incorrect**		CBS or DB
		Cause	Block number not in ascending order	
		Effect	Program is not stored	
		Elimination	Correct program	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

<table>
<tr><th>Cl.</th><th>No.</th><th colspan="2">Error message,
error analysis and elimination</th><th>Message/
Display</th></tr>
<tr><td colspan="5">Traversing program errors Error response: “Warning” see Table 11-2</td></tr>
<tr><td rowspan="4">8 (08)</td><td rowspan="4">24 (18)</td><td colspan="2">G function 1 unacceptable</td><td rowspan="4">CBS
or
DB</td></tr>
<tr><td>Cause</td><td>• The number programmed as G function 1 is not allowed.
• In block, other data besides M-functions were programmed with dwell time (G04).</td></tr>
<tr><td>Effect</td><td>Program/block not stored</td></tr>
<tr><td>Elimina-tion</td><td>Correct program, per cause</td></tr>
<tr><td rowspan="4">8 (08)</td><td rowspan="4">25 (19)</td><td colspan="2">G function 2 unacceptable</td><td rowspan="4">CBS
or
DB</td></tr>
<tr><td>Cause</td><td>The number programmed as G function 2 is not allowed.</td></tr>
<tr><td>Effect</td><td>Program/block not stored</td></tr>
<tr><td>Elimina-tion</td><td>Correct program, per cause</td></tr>
<tr><td rowspan="4">8 (08)</td><td rowspan="4">26 (1A)</td><td colspan="2">G function 3 unacceptable</td><td rowspan="4">CBS
or
DB</td></tr>
<tr><td>Cause</td><td>• The number programmed as G function 3 is not allowed.
• External block change (G50) was programmed in a block together with continuous operation for setting actual value on the fly (G88/89).
• A tool offset (G43, G44) was called up without D number.
• In selecting a D number, the direction specification is missing for the tool offset (G43, G44).</td></tr>
<tr><td>Effect</td><td>Program/block not stored</td></tr>
<tr><td>Elimina-tion</td><td>Correct program, per cause</td></tr>
<tr><td rowspan="4">8 (08)</td><td rowspan="4">27 (1B)</td><td colspan="2">M function unacceptable</td><td rowspan="4">CBS
or
DB</td></tr>
<tr><td>Cause</td><td>• The number programmed as M function is not allowed.
• At least two of the M functions M0, M2, M18, M30, which cancel each other out, are found in one block.</td></tr>
<tr><td>Effect</td><td>Program/block not stored</td></tr>
<tr><td>Elimina-tion</td><td>Correct program, per cause</td></tr>
<tr><td rowspan="4">8 (08)</td><td rowspan="4">28 (1C)</td><td colspan="2">Position/dwell time missing</td><td rowspan="4">CBS
or
DB</td></tr>
<tr><td>Cause</td><td>• No dwell time specified in block with G04.
• Target position missing with external block change (G50).
• No new actual value programmed for the function continuous operation with setting actual value on the fly (G88/89).</td></tr>
<tr><td>Effect</td><td>Program/block not stored</td></tr>
<tr><td>Elimina-tion</td><td>Correct program, per cause</td></tr>
</table>

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

Table 11-8 General data errors, machine data errors, traversing program errors, continued

Cl.	No.	Error message, error analysis and elimination		Message/ Display
Traversing program errors Error response: "Warning" see Table 11-2				
8 (08)	**29 (1D)**	**Incorrect D-NO (>20)**		CBS or DB
		Cause	The number for tool offset is greater than 20	
		Effect	Program/block not stored	
		Elimina-tion	Correct program, per cause	
8 (08)	**30 (1E)**	**Error, subroutine**		CBS or DB
		Cause	Subroutine without callup number	
		Effect	Program is not stored	
		Elimina-tion	Correct program, per cause	
8 (08)	**31 (1F)**	**Velocity missing**		CBS or DB
		Cause	No velocity was programmed	
		Effect	Program/block not stored	
		Elimina-tion	Correct program, per cause	
8 (08)	**32 (20)**	**Error, callup subroutine**		CBS or DB
		Cause	Block syntax for callup subroutine is incorrect	
		Effect	Program is not stored	
		Elimina-tion	Correct program, per cause	
8 (08)	**33 (21)**	**D function unacceptable**		CBS or DB
		Cause	Block syntax for invoking a D function is incorrect	
		Effect	Program is not stored	
		Elimina-tion	Correct program, per cause	
8 (08)	**34 (22)**	**Incorrect program length**		CBS or DB
		Cause	Maximum block number exceeded	
		Effect	Program is not stored	
		Elimina-tion	Correct program, per cause	

Cl. = Detail event class, No. = Detail event number, CBS = checkback signals, DB = data block

Note: Value (xx) = Hexadecimal notation of the error number

■

Technical Specifications

A

Overview

This chapter describes the technical data for the FM 354 positioning module.

- General technical data
- Dimensions and weight
- Load memory
- Encoder inputs
- Setpoint output
- Digital inputs
- Digital outputs

General technical data

General technical data include:

- Electromagnetic compatibility
- Shipping and storage conditions
- Ambient mechanical and climate conditions
- Data on insulation testing, protection class and degree of protection

This information contains standards and test values with which the S7-300 complies, or according to whose criteria the S7-300 was tested.

The general technical data are described in the manual “Installing an S7-300.”

UL/CSA certifications

The following certifications are on record for the S7-300:

UL Recognition Mark
Underwriters Laboratories (UL) in compliance with
UL Standard 508, File E 116536

CSA Certification Mark
Canadian Standard Association (CSA) in compliance with
Standard C 22.2 No. 142, File LR 48323

FM approval

The FM approval is on record for the S7-300:
FM certification in accordance with Factory Mutual Approval Standard Class Number 3611, Class I, Division 2, Group A, B, C, D.

Warning

Potential for personal injury and property damage.

In areas where there is a risk of explosion, personal injury and property damage may occur if you disconnect plugs while the S7-300 is in operation.

In areas where there is a risk of explosion, always cut off power to the S7-300 before disconnecting plugs.

Warning

WARNING - NEVER DISCONNECT WHILE CIRCUIT IS LIVE UNLESS LOCATION IS KNOWN TO BE NONHAZARDOUS

CE marking

Our products are in compliance with the EU Guideline 89/336/EEC “Electromagnetic Compatibility” and the harmonized European standards (EN) which it embodies.

The EC Declaration of Conformity in accordance with Article 10 of the EU Guideline referenced above is contained in this manual (see Chapter B).

Application

SIMATIC products are designed for application in an industrial environment.

Application	Requirement concerning	
	Noise emission	**Noise immunity**
Industry	EN 50081-2 : 1993	EN 50082-2 : 1995
Residential	Individual license	EN 50082-1 : 1992

Observe installation guidelines

SIMATIC products meet the requirements, provided you observe the installation guidelines set forth in the manuals during installation and operation.

Power ratings

Technical data: Power ratings

Table A-1 Power ratings

Supply voltage	20.4...28.8 V
Power consumption from 24 V	0.35 A
Power loss	8 W
Startup current	2.2 A
Power consumption from 5 V backplane bus	100 mA

Dimensions and weights

Technical data for dimensions and weights:

Table A-2 Technical data: Dimensions and weight

Dimensions W × H × D (mm/in)	80 × 125 × 118
Weight (g/lb)	approx. 550 / 1 lb, 3.3 oz

Memory for parameter data

RAM memory 16 Kbytes
FEPROM for retentive storage of parameter data

FM cycle

2 ms

Encoder inputs

Technical data for encoder inputs:

Table A-3 Technical data: encoder inputs

Position detection	• Incremental • Absolute (SSI)
Signal voltages	Inputs: 5 V per RS422
Encoder supply voltage	• 5.2 V/300 mA • 24 V/300 mA
Input frequency and line length for incremental encoder	• Max. 1 MHz wih 10 m conductor length shielded • Max. 500 kHz with 35 m conductor length shielded
Data transmission rates and line length for absolute encoder (SSI)	• Max. 1.25 Mbit/s with 10 m conductor length shielded • Max. 125 kbit/s with 100 m conductor length shielded

Drive port

Technical data concerning setpoint output:

Table A-4 Technical data: setpoint output

Rated voltage range	–10...10 V
Output current	–3...3 mA

Digital inputs

Technical data for digital inputs:

Table A-5 Technical data: digital inputs

Number of inputs	5 (including controller ready)
Supply voltage	24 V DC (allowable range: 20.4...28.8 V)
Electrical isolation	No
Input voltage	• 0 Signal: –3...5 V • 1 Signal: 11...30 V
Input current	• 0 Signal: ≤ 2 mA • 1 Signal: 6...15 mA
Input delay (DI1...4)	• 0 → 1 Signal: typ. 15 µs • 1 → 0 Signal: typ. 150 µs
Connecting a 2-conductor sensor	Possible

Digital outputs

Technical data for digital outputs:

Table A-6 Technical data: Digital outputs

Number of outputs	4
Supply voltage	24 V DC (allowable range: 20.4...28.8 V)
Electrical isolation	No
Output voltage	• 0 Signal: Residual current max. 2 mA • 1 Signal: (Power supply –3 V)
Output current on signal “1” • at ambient temperature of 40°C – Rated value – Permissible value range – Lamp load • at ambient temperature of 60°C – Rated value – Permissible value range	 0.5 A (total current 2 A) 5 mA...0.6 A (over power supply range) max. 5 W 0.1 A (total current 0.4 A) 5 mA...0.12 A (over power supply)
Short circuit protection	Yes
Switching rate	• Resistive load: max. 100 Hz • Inductive load: max. 0.25 Hz

■

EC Declaration of Conformity

B

SIEMENS

EG-Konformitätserklärung

Nr. E002 V 21/03/97

Hersteller: SIEMENS AG

Anschrift: SIEMENS AG AUT 2
Frauenauracherstraße 80
91056 Erlangen

Produktbezeichnung:

SINUMERIK	805, 805SM-P, 805SM-TW, 810, 810D, 820, 840C, 840CE, 840D, 840DE, FM NC
SIMATIC	FM 353, FM 354, FM 357
SIROTEC	RCM1D, RCM1P
SIMODRIVE	610, 611A, 611D, MCU, FM STEPDRIVE

Die bezeichneten Produkte stimmen mit den Vorschriften folgender Europäischer Richtlinie überein:

89/336/EWG-Richtlinie des Rates zur Angleichung der Rechtsvorschriften der Mitgliedsstaaten über die elektromagnetische Verträglichkeit (geändert durch 91/263/EWG, 92/31/EWG und 93/68/EWG)

Die Einhaltung dieser Richtlinie setzt einen EMV-gerechten Einbau der Produkte in die Gesamtanlage voraus.
Anlagenkonfigurationen, bei der die Einhaltung dieser Richtlinie nachgewiesen wurde, sowie angewandte Normen, siehe:

- **Anhang A1 - A14** (Anlagenkonfigurationen)
- **Anhang B1 - B7** (Komponenten)
- **Anhang C** (Normen)

SIEMENS

Erlangen, den 21.03.1997

R. Müller
Entwicklungsleitung
Name, Funktion — Unterschrift

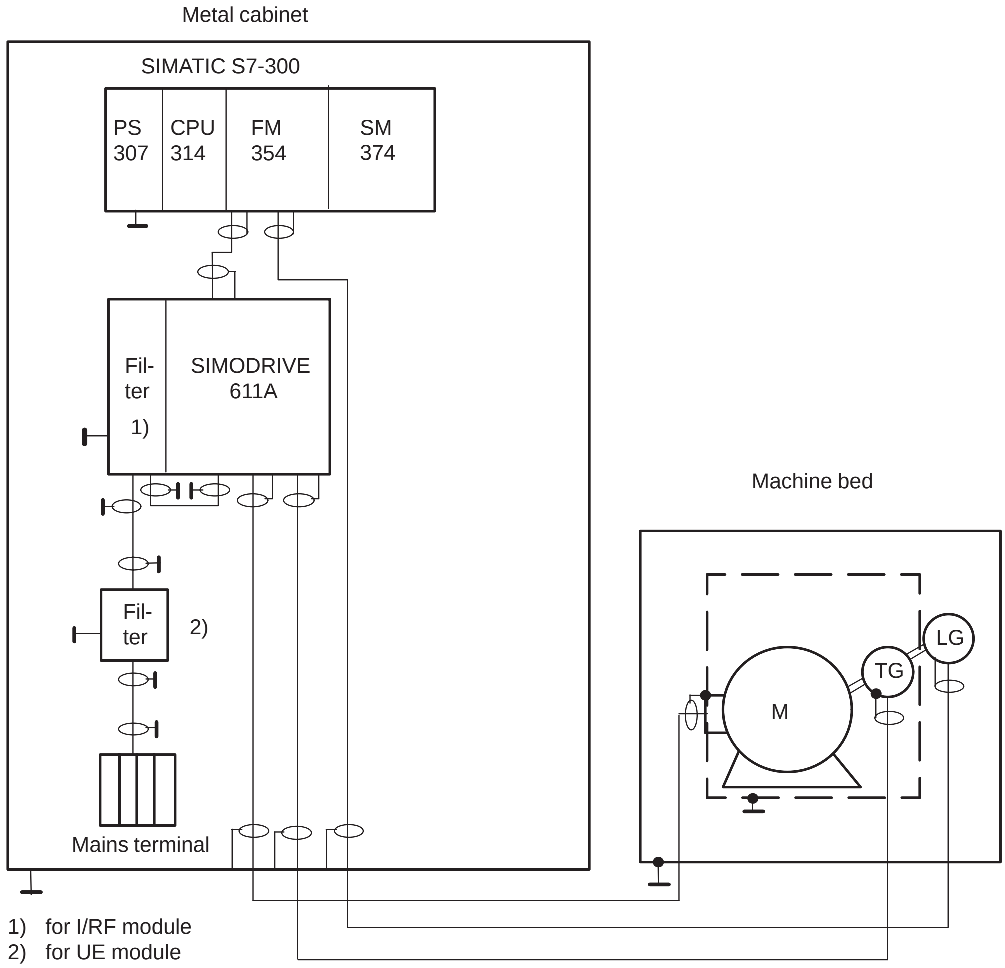
K. Krause
Qualitätsmanagement
Name, Funktion — Unterschrift

Der Anhang ist Bestandteil dieser Erklärung.
Diese Erklärung bescheinigt die Übereinstimmung mit der genannten Richtlinie, ist jedoch keine Zusicherung von Eigenschaften im Sinne der Produkthaftung.
Die Sicherheitshinweise der mitgelieferten Produktdokumentation sind zu beachten.

Appendix A to EC Declaration of Conformity No. 002 V 01/02/96

A12: Typical plant configuration

FM positioning module (FM354 for servo drive) / SIMODRIVE 611A



1) for I/RF module
2) for UE module

- All components which are approved in accordance with the ordering document for a combined FM positioning module / SIMODRIVE 611A plant meet the requirements of the 89/336/EEC directive when operated together.
- For conformity to standards, please see Appendix C

Note

The plant configuration sketch shows only the basic measures required for conformity of a typical plant configuration with the 89/336/EEC directive. Installation instructions for EMC–oriented plant configurations are provided in the product documentation and the Siemens EMC guideline (Order No.: 6ZB5410-0HX01-0AA0). These instructions must be followed, especially where the plant configuration differs from the description above.

Appendix C to EC Declaration of Conformity No. E002 V 21/03/97

C: Compliance of the products with the 89/336/EEC directive has been verified by tests performed in accordance with the following basic technical specifications and the basic specifications listed therein:

Basic technical specification: EN 50081-2 Status 8/93

Basic specifications:	
EN 55011	1)

Basic technical specification: EN 50082-2 Status 3/95

Basic specifications:	Test subject:	
ENV 50140	2)	High–frequency irradiation
ENV 50141	3)	HF flow on conductors (amplitude–modulated)
ENV 50204		HF flow on conductors (pulse–modulated)
EN 61000-4-8	4)	Magnetic fields
EN 61000-4-2	5)	Static discharge
EN 61000-4-4	6)	High–speed transients (bursts)

Specifications also fulfilled:

cf 1):	VDE 0875 Part 11
cf. 2):	VDE 0847 Part 3
cf 3):	IEC 801-6
cf. 4):	VDE 0847 Part 4-8
	IEC 1000-4-8
cf 5):	VDE 0847 Parts 4-2
	EN 60801 Part 2
	IEC 801-2
	VDE 0843 Part 2
cf. 6):	VDE 0843 Part 4
	VDE 0847 Part 4-4
	IEC 801-4

List of Abbreviations

C

AS	Automation system
BA	Mode
BA “A/AE”	“Automatic/Automatic single block” mode
BA “REF”	“Reference point approach” mode
BA “STE”	“Open-loop control” mode
BA “SM”	“Incremental approach” mode
BA “T”	“Jogging” mode
BIE	Binary result
BP	Mode parameter
CPU	Central Processing Unit of the SIMATIC S7
DB	Data block
DBB	Data block byte
DBX	Data block bit
DB-MD	Data block for machine data
DB-SM	Data block for increments
DB-WK	Data block for tool offset data
DB-NC	Data block for traversing programs
DB-SS	Data block for status messages
DEKL	Detail event class
DENR	Detail event number
DP	Distributed I/O
ESD	ESD
EMC	Electromagnetic compatibility
EN	Enable (input parameter in LAD representation)
ENO	Enable Output (output parameter in LAD representation)
EPROM	Erasable programmable read-only memory
EXE	External pulse shaper
FB	Function block

FC	Function
FEPROM	Flash EPROM: Read/Write memory
FM	Function module
HEX	Hexadecimal
HMI	Device for operating and monitoring of a process
I	Input parameter
IM	Interface module (SIMATIC S7)
I/Q	In/out parameter (initialization parameter)
LAD	Ladder program
LED	Light Emitting Diode
MLFB	Machine-readable order designation
MPI	Multi Point Interface
MSR	Measurement system raster
MDI	Manual Data Input
OB	Organization block
OP	Operator panel
PG	Programming device
PLC	Porgrammable controller
PS	Power Supply (SIMATIC S7)
PWM	Pulse width modulation
Q	Output parameter
RFG	Controller enable
RPS	Reference point switch
SDB	System data block
SFC	System Function Call (integrated fucntions)
STEP 7	Programming device software for SIMATIC S7
S7-300	PLC of medium performance range
SM	Signal module (SIMATIC S7, e.g. input/output module)
SSI	Synchronous Serial Interface
STL	Statement list
SZL	System status list
TF	Technology function
UP	User program

■

Index

S

T