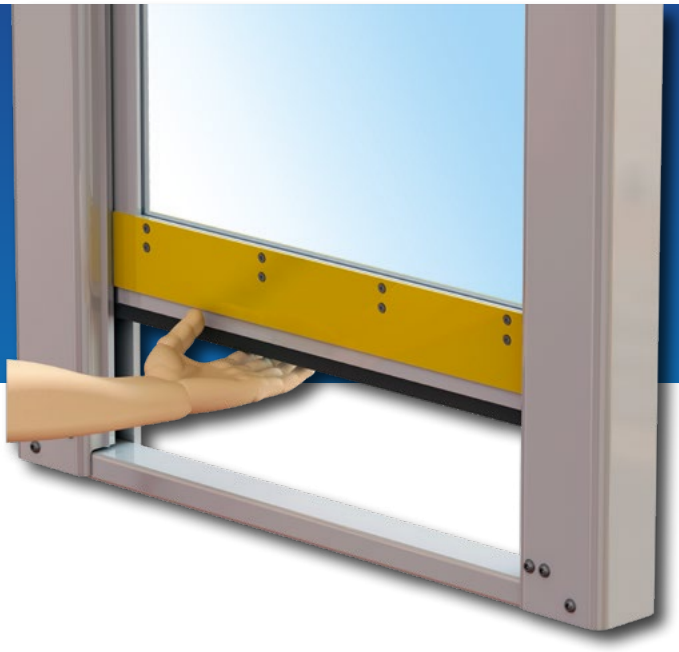
STRASSER
inspired by ideas



MecLock© Safeguards
with integrated operator protection

CONTROL OF ELECTRICAL DRIVEN SAFEGUARDS WITH PLC OR FIELDBUS INTERFACE

*INFORMATION BROCHURE**

*) This information brochure is a supplement to the assembly instructions. Use only in combination with the complete assembly instructions!

Content

1 Drive system and operating cycle

Drive concept servo motor

All electrically driven MecLock safeguards have a positioning motor with integrated positioning control and controller. All records and actuator parameters are already stored at the factory in the motor control to ensure quick commissioning.

The positioning motor is operated directly in the 230 Volt or 400 Volt AC mains. For the positioning motor, the interface variants PLC, Profibus or Profinet can be selected for the control. In addition, the motor is also available with the option STO (Safe Torque Off).

Commissioning and further parameter settings in the motor control can be carried out via the serial RS232 interface using the software ServoLink. Status messages and diagnostic functions are supported by this interface.

For versions with fieldbus, all parameters can be set via the fieldbus. As standard, electrically operated safeguards are parameterized with three movement data records (opened position, closed position, intermediate position half stroke).

Quick stop function servo motor

When the MecLock protection system is activated, the closing door leaf or protective field is stopped without overrun. Since an immediate stop of the electric drive is not possible, slip or overload clutches are used to decouple the drive train from the locked door leaf or protective field and to aviod damage to the gear unit and servo motor. The quick stop function, which is triggered by a ground circuit or an actuated band switch, then automatically stops the still running servo motor.

Sensor reference

The safeguard references itself automatically via the built-in reference sensor if no reference position has been saved, a previous error has led to the loss of the reference position or the power supply of the logic part to the servo motor has been interrupted. An interruption of the power supply at the drive part of the servo motor while maintaining the power supply at the logic part does not lead to the loss of the reference position.

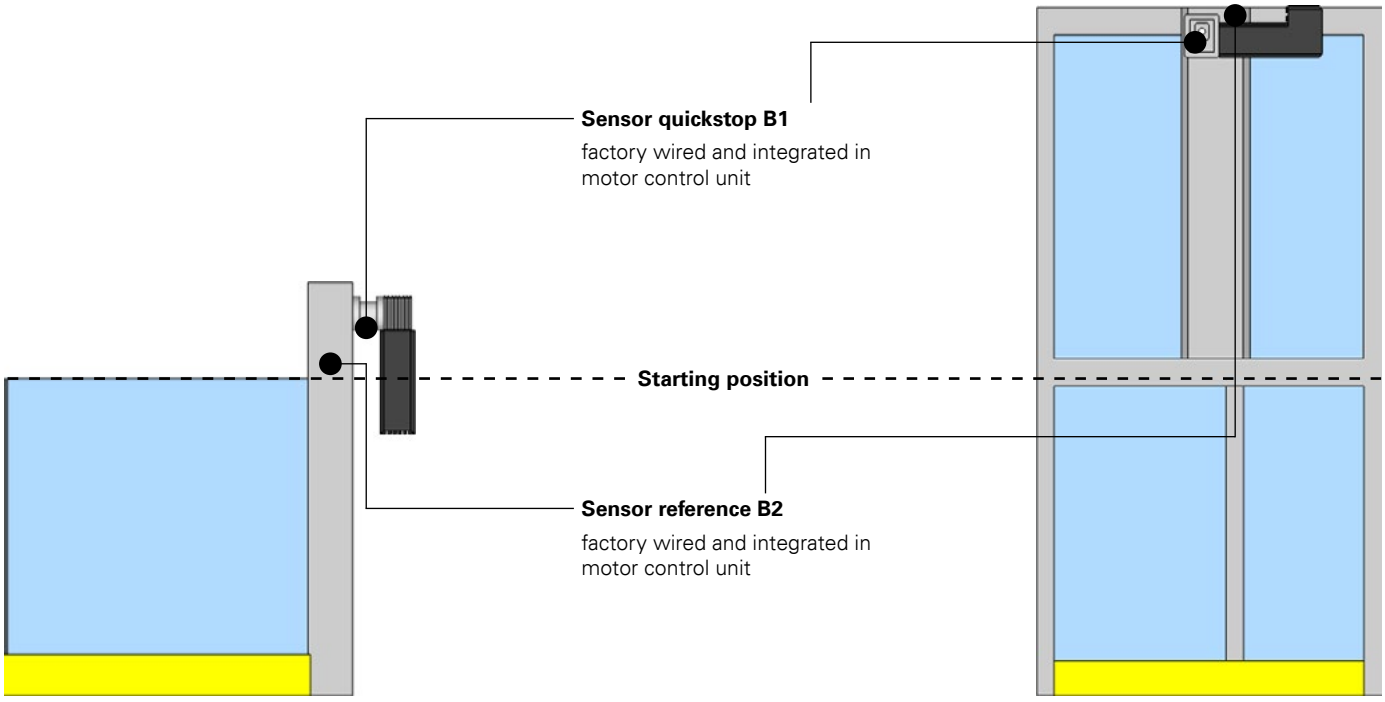
Intermediate position

In the case of cycle-time critical processes, the full stroke of the safeguard for loading and unloading workpieces is often not required in normal operation. By using an intermediate position, the opening and closing cycle can be optimized here, since it is no longer necessary to travel the full stroke of the safeguard in the cycle.

By default, electrically operated safeguards have three controllable positions ex works (opened position, closed position, intermediate half-stroke position). The position of the factory-programmed intermediate position can be changed or other intermediate positions can be added (only PROFIBUS DP or PROFINET interface). For drives with a pure PLC interface, only the setting, adjustment and control of one intermediate position is possible via programming cable.

Tubular fixed resistor (optional)

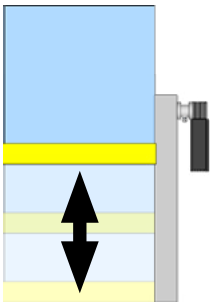
For very heavy safeguards and very short downtimes, a tubular fixed resistor may need to be installed.



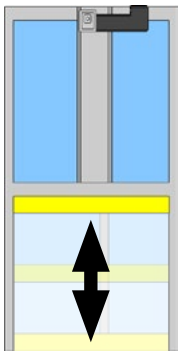
Safeguard switch on, restart and operation

	Operation sequence	Implementation
1.	Power supply on	Customer control unit
2.	Reference movement	Motor control unit
3.	Starting position movement	Motor control unit
4.	Safeguard is ready and fully operational	
	Intermediate position (1/2 stroke) command	Customer control unit
	Closed end position command	Customer control unit
	Opened end position command	Customer control unit

Safety lifting column

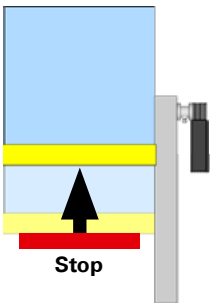


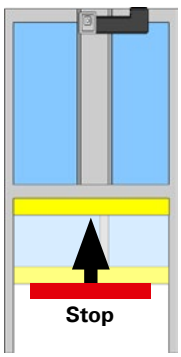
Safety window



Reaching in closing safeguard during operation (interruption)

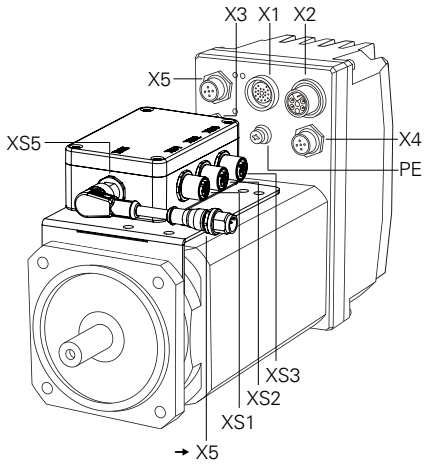
	Operation sequence	Implementation
1.	Safeguard close	Customer control unit
2.	Contact bar actuate	Operator / obstacle
3.	Quickstop pane frame / protection field	Motor control unit
4.	Reference movement	Motor control unit
5.	Starting position movement	Motor control unit
6.	Safeguard is ready and fully operational	





2 Technical specifications

2.1 Servo motor 32/3



The drive unit is installed in the MecLock safety guards ex works ready for operation.

The associated power and signal cables must be ordered separately in the required length.

A programming set and suitable tubular fixed resistors are optionally available.

Safety information about the servomotor with STO according to EN ISO 13849-1:2007

Category: 3	MTTFd [a]: 100 (high)
Performance level: e	DC: 92.85 (medium)
PFH [1/h]: 3.71×10^{-8}	Service life [a]: 20

Logic & I/O – Supply

Nominal value	24 VDC
Admissible range	22.3 – 26.1 VDC
Ripple	≤ 10%
Reverse polarity protection	with diode
External fuse	max. 2 A / medium slow
Current consumption	approx. 0.15 A at 24 VDC
with released brake	approx. 0.9 A
Resistance to drop-outs	
Duration of drop-outs	10 ms
Repetition	1 s

Motor – Supply

Galvanic isolation	Safe isolation according to EN 50178
External fuse	8 A
Current consumption	ca. 5 A
Connection	1-phase
Nominal value	230 VAC
Admissible range	100 V – 253 VAC
Frequency	48 – 63 Hz

Motor

Motor power (S3 25%)	1.82 kW
Rated power	1.05 kW
Rated torque (S3 25%)	5.1 Nm
Peak torque	8.0 Nm
Rated speed	3000 rpm
Inertia	0.16×10^{-3} kgm²
Inertia with brake	0.216×10^{-3} kgm²
Motor supply	230 VAC
Peak current	10 A
Rated motor current	4.5 A AC
Logic supply	24 VDC
Axle resolution with resolver	4096 Ipr
Protection class (mounting specific)	IP 65
Weight with brake	6.6 kg

Holding brake

The optional standstill brake is operated by the controller.
The supply of the brake is the logic supply.

Holding torque brake	9 Nm
Response time opening	40 ms
Response time closing	7 ms
Nominal current	0.75 A

Digital inputs

Input voltage	
Nominal value	+24 VDC
Maximal voltage	+30 VDC
Threshold high-Level	+12 VDC
Threshold low-level	+3 VDC
Minimum input voltage	0 VDC
Input current at nominal voltage	typical 1.5 mA
Delay times	
Low -> High	typical 10 µs
High -> Low	typical 10 µs
Galvanic isolation	no

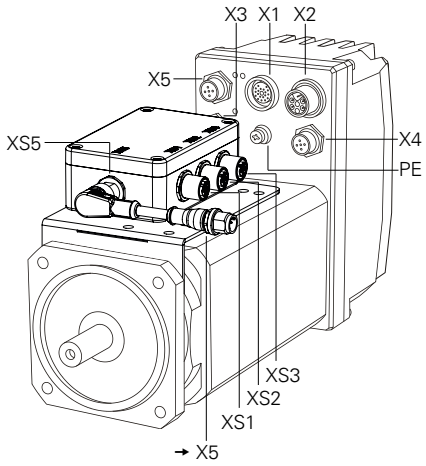
Digital outputs 24 VDC

Type of outputs	Transistor (MOS-FET)
Output current at high signal	
Nominal value	0.5 A
Residual current at low signal	250 µA
Power supply	+24 V logic supply
Short circuit protection	yes
Current limitation	1 A
Trip current	1 A

External protection with inductive loads: varistor (VG-A/24) or quenching diode (e.g. Murrelektronik LG-A01) recommended

Parallel connection admissible	no
Delay time outputs	
Low -> High	typical 100 µs
High -> Low	typical 100 µs
Galvanic isolation	no

2.2 Servo motor 433/4



The drive unit is installed in the MecLock safety guards ex works ready for operation.

The associated power and signal cables must be ordered separately in the required length.

A programming set and suitable tubular fixed resistors are optionally available.

Safety information about the servomotor with STO according to EN ISO 13849-1:2007

Category: 3	MTTFd [a]: 100 (high)
Performance level: e	DC: 92.85 (medium)
PFH [1/h]: 3.71×10^{-8}	Service life [a]: 20

Logic & I/O – Supply

Nominal value	24 VDC
Admissible range	22.3 – 26.1 VDC
Ripple	≤ 10%
Reverse polarity protection	with diode
External fuse	max. 2 A / medium slow
Current consumption	approx. 0.15 A at 24 VDC
with released brake	approx. 0.9 A
Resistance to drop-outs	
Duration of drop-outs	10 ms
Repetition	1 s

Motor – Supply

Galvanic isolation	Safe isolation according to EN 50178
External fuse	8 A
Current consumption	ca. 5 A
Connection	3-phase
Nominal value	3× 400 VAC
Admissible range	150 V – 440 VAC
Frequency	48 – 63 Hz

Motor

Motor power (S3 25%)	2.49 kW
Rated power	1.78 kW
Rated torque (S3 25%)	7.0 Nm
Peak torque	14.0 Nm
Rated speed	3400 rpm
Inertia	0.24×10^{-3} kgm²
Inertia with brake	0.296×10^{-3} kgm²
Motor supply	3× 400 VAC
Peak current	10 A
Rated motor current	5.1 AAC
Logic supply	24 VDC
Axle resolution with resolver	4096 Ipr
Protection class (mounting specific)	IP 65
Weight with brake	7.9 kg

Holding brake

The optional standstill brake is operated by the controller.
The supply of the brake is the logic supply.

Holding torque brake	9 Nm
Response time opening	40 ms
Response time closing	7 ms
Nominal current	0.75 A

Digital inputs

Input voltage	
Nominal value	+24 VDC
Maximal voltage	+30 VDC
Threshold high-Level	+12 VDC
Threshold low-level	+3 VDC
Minimum input voltage	0 VDC
Input current at nominal voltage	typical 1.5 mA
Delay times	
Low -> High	typical 10 µs
High -> Low	typical 10 µs
Galvanic isolation	no

Digital outputs 24 VDC

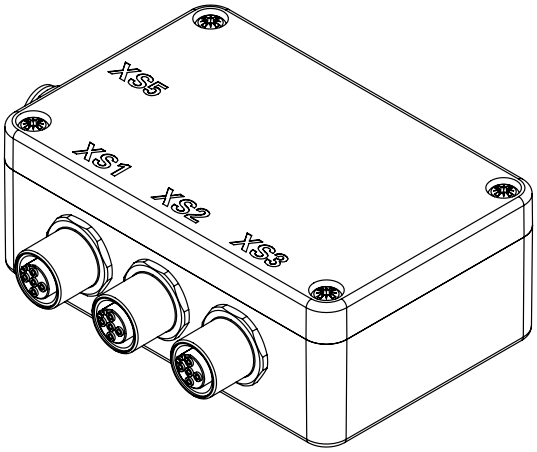
Type of outputs	Transistor (MOS-FET)
Output current at high signal	
Nominal value	0.5 A
Residual current at low signal	250 µA
Power supply	+24 V logic supply
Short circuit protection	yes
Current limitation	1 A
Trip current	1 A

External protection with inductive loads: varistor (VG-A/24)
or quenching diode (e.g. Murrelektronik LG-A01) recommended

Parallel connection admissible	no
Delay time outputs	
Low -> High	typical 100 µs
High -> Low	typical 100 µs
Galvanic isolation	no

2.3 Connection box for servo motor type 32/3 and type 433/4

The connection box is by default attached to the servo motor.



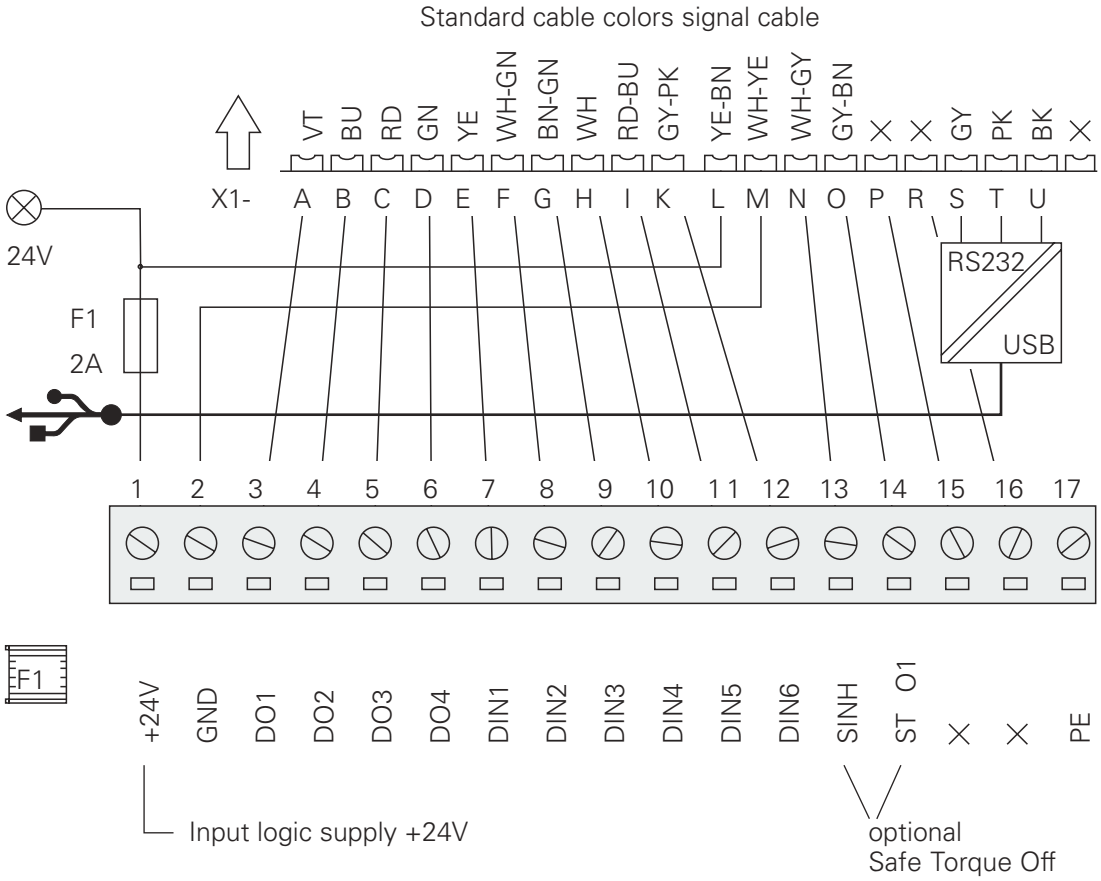
Terminal designation:

- XS1 In: quick stop
- XS2 In: quick stop
- XS3 In: sensor reference
- XS5 Out: servo motor

2.4 Terminal box EPS11 (optional)

The terminal box is installed in the control cabinet and the 24-volt signal line is clamped on. This enables online diagnostics directly at the control cabinet without plugging directly into the servo motor. The use of a terminal box is recommended if the servo motor is in a location that is difficult to access, e.g. due to its height. Due to the fixed installation, a separate terminal box is required for each servomotor.

- Plug-in screw terminals for signal connections
- USB interface converter with galvanic isolation
- Integrated fuse for 24 V supply
- Indicator light for 24 V
- DIN rail module, 22.5 mm wide



DI... : Digital input
DO...: Digital output
SINH: Start inhibitor
STO1: Safe Torque Off
GND: Ground
PE: Protective earth

2 Technical specifications

2.5 Tubular fixed resistor EPB135 (optional)

Cemented wirewound tubular fixed resistor in one-tube design, degree of protection IP20, in perforated steel sheet enclosure, mounting vertical to mounting surface, connections optionally at terminals or at screw or fast-on clips at the resistor. For integration into switch cabinets. When using the tubular fixed resistor, the matching power cables for external ballast resistor must also be ordered.

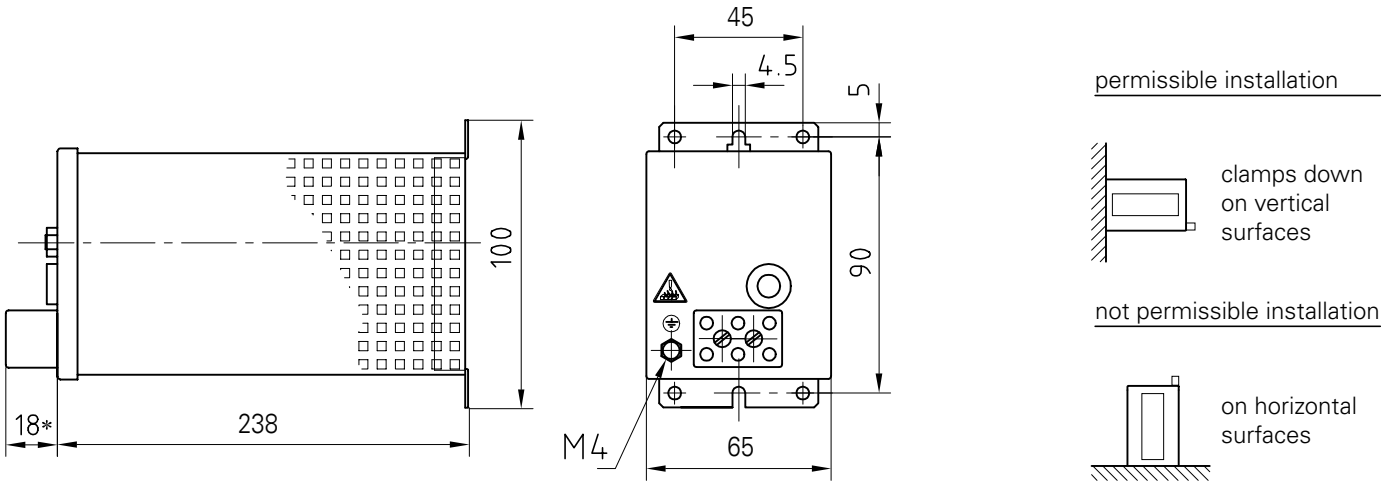
Technical specifications

- Typical power in W at 40 °C, 100% DCF: 100 W
- Resistance range Ω-value (single-phase): 0.33 Ω – 10 kΩ
- Approx. weight: 0.6 kg

Description

- Two connectors wired on a porcelain terminal, which is accessible without demounting the cover and protected against access to hazardous parts according to BGV A2.
- The terminal is fixed on the enclosure front plate.
- Adjustable clip not available

Dimensions



2.6 Tubular fixed resistor EPB200 (optional)

Cemented wirewound tubular fixed resistor in one-tube design, degree of protection IP20, in perforated steel sheet enclosure, mounting vertical to mounting surface, connections optionally at terminals or at screw or fast-on clips at the resistor. For integration into switch cabinets. When using the tubular fixed resistor, the matching power cables for external ballast resistor must also be ordered.

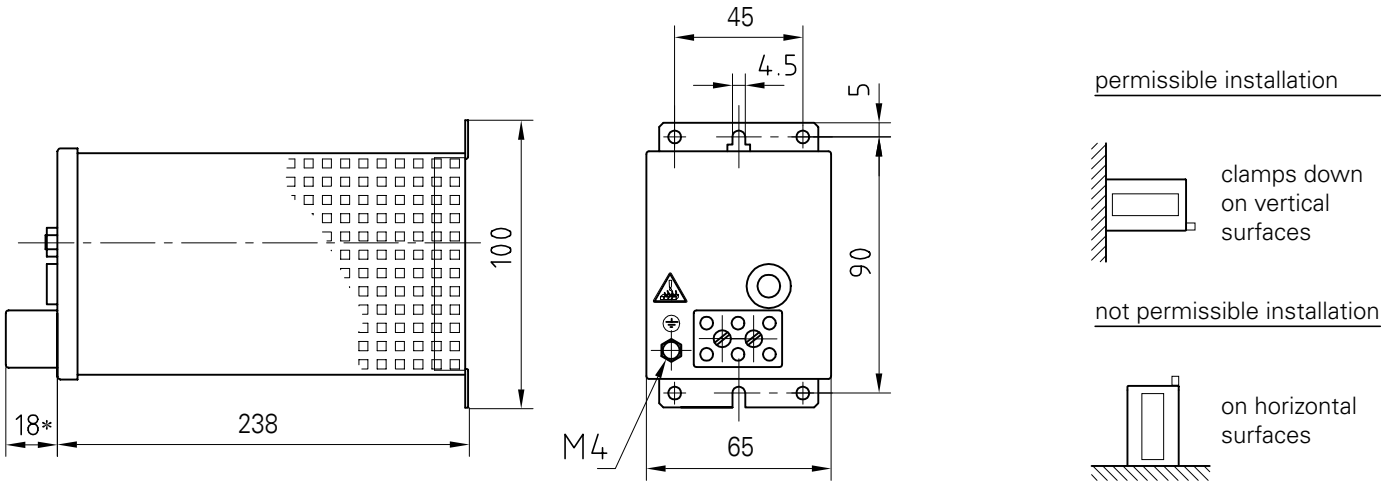
Technical specifications

- Typical power in W at 40 °C, 100% DCF: 180 W
- Resistance range Ω-value (single-phase): 0.68 Ω – 5.6 kΩ
- Approx. weight: 0.8 kg

Description

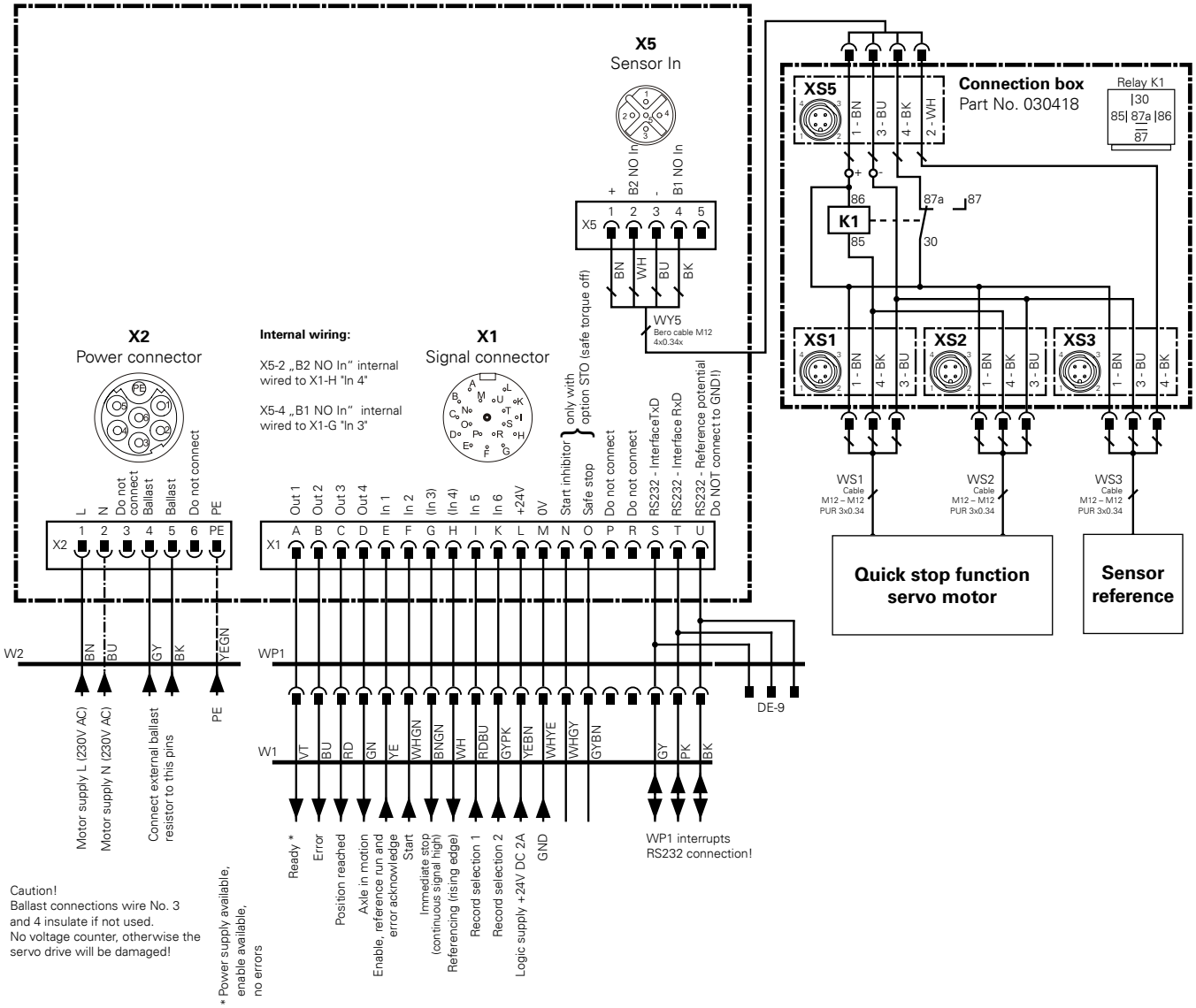
- Two connectors wired on a porcelain terminal, which is accessible without demounting the cover and protected against access to hazardous parts according to BGV A2.
- The terminal is fixed on the enclosure front plate.
- Adjustable clip not available

Dimensions



3 Circuit diagrams

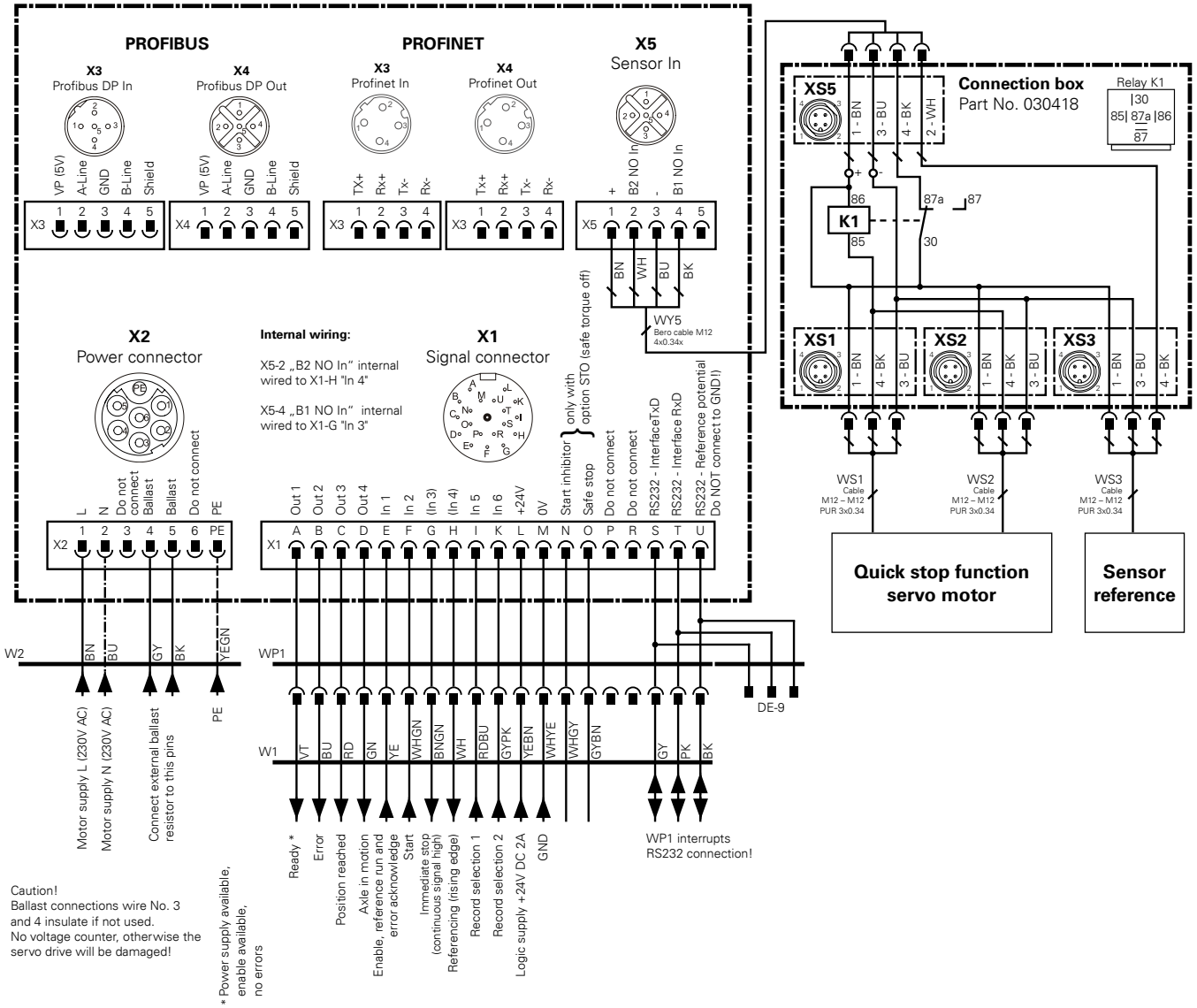
3.1 Circuit diagram servo motor 32/3 PLC interface



Input		Function	Information
In1 (24V)		Release	Permanent signal
In2 (24V)		Start	Positive edge starts selected set
In3 (24V)		Quick stop	Do not switch from outside (internal use)
In4 (24V)		Reference	Do not switch from outside (internal use)
In5 (24V)	+ In6 (0V)	Open	Record 1
In5 (0V)	+ In6 (24V)	Close	Record 2
In5 (24V)	+ In6 (24V)	Intermediate position	Record 3 (Standard = 1/2 stroke)

Note:
Connection to the safety switch and the optional position sensors are not detailed in this electrical connection plan.
The software ServoLink for the adaptation of the driving data sets and motor parameters is available together with the programming cable WP1 as an accessory.
The records are controlled via the PLC interface.

3.2 Circuit diagram servo motor 32/3 PROFIBUS DP or PROFINET interface



Input		Function	Information
In1 (24V)		Release	Permanent signal
In2 (24V)		Start	Positive edge starts selected set
In3 (24V)		Quick stop	Do not switch from outside (internal use)
In4 (24V)		Reference	Do not switch from outside (internal use)
In5 (24V)	+ In6 (0V)	Open	Record 1
In5 (0V)	+ In6 (24V)	Close	Record 2
In5 (24V)	+ In6 (24V)	Intermediate position	Record 3 (Standard = 1/2 stroke)

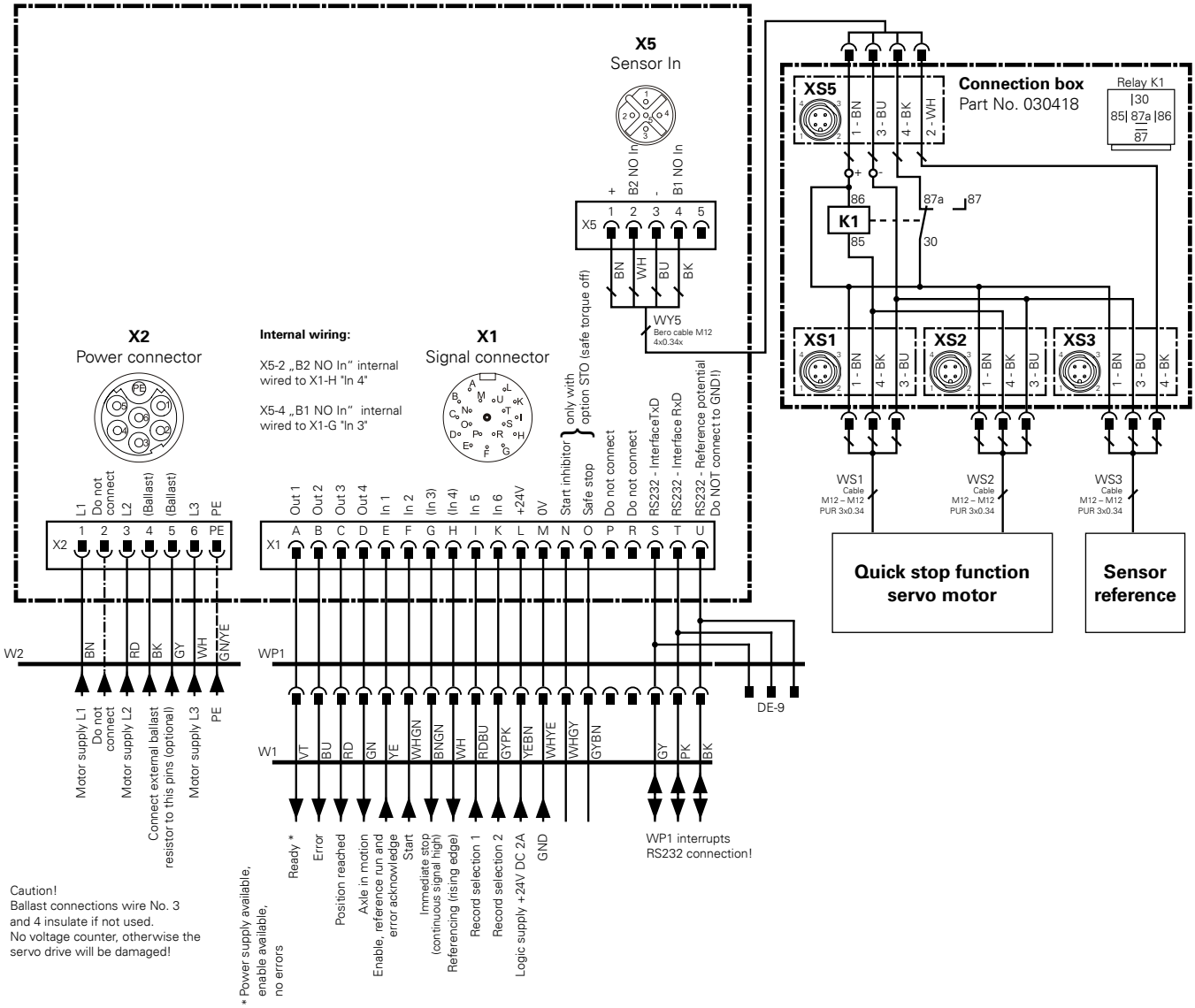
Note for control via PROFINET:
If the record selection and start functions are carried out via commands 286 and 287 via PROFINET, the input functions „Start" (In2) and „Record selection x" (In5 and In6) must not be set or used, as the input functions are prioritised over the PROFINET commands.

Note:
Connection to the safety switch and the optional position sensors are not detailed in this electrical connection plan.
The software ServoLink for the adaptation of the driving data sets and motor parameters is available together with the programming cable WP1 as an accessory.
The records are controlled via the PROFIBUS DP or PROFINET interface.

Programming via field bus:
To drive the motor and to integrate into the „Step 7" / „TIA portal" software, a library with a function block (FB) and programming examples are included in the scope of delivery. The documentation for the programming is available on request.

3 Circuit diagrams

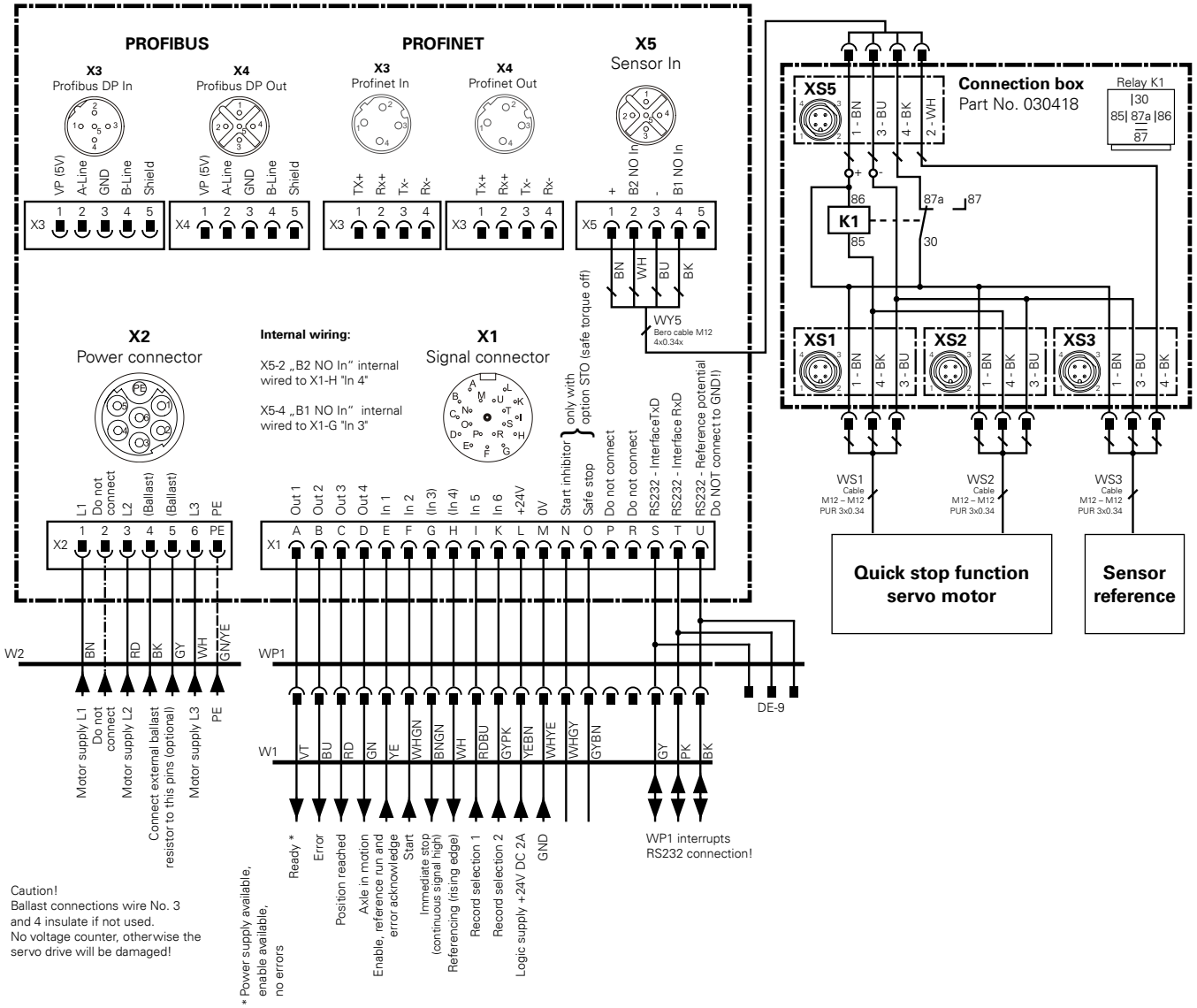
3.3 Circuit diagram servo motor 433/4 PLC interface



Input		Function	Information
In1 (24V)		Release	Permanent signal
In2 (24V)		Start	Positive edge starts selected set
In3 (24V)		Quick stop	Do not switch from outside (internal use)
In4 (24V)		Reference	Do not switch from outside (internal use)
In5 (24V)	+ In6 (0V)	Open	Record 1
In5 (0V)	+ In6 (24V)	Close	Record 2
In5 (24V)	+ In6 (24V)	Intermediate position	Record 3 (Standard = 1/2 stroke)

Note:
Connection to the safety switch and the optional position sensors are not detailed in this electrical connection plan.
The software ServoLink for the adaptation of the driving data sets and motor parameters is available together with the programming cable WP1 as an accessory.
The records are controlled via the PLC interface.

3.4 Circuit diagram servo motor 433/4 PROFIBUS DP or PROFINET interface



Input		Function	Information
In1 (24V)		Release	Permanent signal
In2 (24V)		Start	Positive edge starts selected set
In3 (24V)		Quick stop	Do not switch from outside (internal use)
In4 (24V)		Reference	Do not switch from outside (internal use)
In5 (24V)	+ In6 (0V)	Open	Record 1
In5 (0V)	+ In6 (24V)	Close	Record 2
In5 (24V)	+ In6 (24V)	Intermediate position	Record 3 (Standard = 1/2 stroke)

Note for control via PROFINET:
If the record selection and start functions are carried out via commands 286 and 287 via PROFINET, the input functions „Start" (In2) and „Record selection x" (In5 and In6) must not be set or used, as the input functions are prioritised over the PROFINET commands.

Note:
Connection to the safety switch and the optional position sensors are not detailed in this electrical connection plan.
The software ServoLink for the adaptation of the driving data sets and motor parameters is available together with the programming cable WP1 as an accessory.
The records are controlled via the PROFIBUS DP or PROFINET interface.

Programming via field bus:
To drive the motor and to integrate into the „Step 7" / „TIA portal" software, a library with a function block (FB) and programming examples are included in the scope of delivery. The documentation for the programming is available on request.

4 Program sequence

Switch on safeguard or restart and operation

	Action	Function	Result	In1	In2	(In3)*1	(In4)*1	In5	In6	Out1	Out2	Out3	Out4
1.	**Power supply on**	**Start motor electronics**	**Motor electronics started**	0	0	1	0	0	0	1	0	0	0
2.	**Enable the motor**	**Safeguard referenced**	**Safeguard ready for operation**										
2.1	Enable 6 sec. after power supply supplied	-	-	**1**	0	1	0	0	0	1	0	0	0
2.2	*Automatic sequence*	Reference movement	Safeguard moves	1	0	1	0	0	0	1	0	0	**1**
2.3	*Automatic sequence*	Reference point reached	Rising signal edge on (In4)	1	0	1	**0-1-0**	0	0	1	0	**1**	**0**
2.4	*Automatic sequence*	Move to initial position	Safeguard moves	1	0	1	0	0	0	1	0	**0**	**1**
2.5	*Automatic sequence*	Initial position reached	Motor stopped	1	0	1	0	0	0	1	0	**1**	**0**
3.	**Record selection for function**	**Load movement data records**	**Safeguard ready to move**										
3.1	Select Record 1 **or**	Open safeguard	Movement data record loaded	1	0	1	0	**1**	**0**	1	0	1	0
3.2	Select Record 2 **or**	Close safeguard	Movement data record loaded	1	0	1	0	**0**	**1**	1	0	1	0
3.3	Select Record 3	Safeguard in intermediate pos.	Movement data record loaded	1	0	1	0	**1**	**1**	1	0	1	0
4.	**Start the record**	**Safeguard moves**	**Safeguard in position**										
4.1	Start*2	Safeguard executes record	Safeguard moves	1	**0-1-0**	1	0	**Record**		1	0	**0**	**1**
4.2	*Automatic sequence*	Motor stop	Safeguard in position	1	0	1	0	**values**		1	0	**1**	**0**
4.3	Switch motor ready for oper.	Safeguard ready for operation	Safeg. rdy to select next record	1	0	1	0	**0**	**0**	1	0	1	0

Continue with step 3 "Switch on safeguard or restart" and new record selection

*1) These inputs deliver an output signal due to the internal motor wiring

*2) The start signal must only be applied to start the record as edge

Tab. 4.1: Function sequence switch on/ restart

Intrusion in the closing safeguard during operation

	Action	Function	Result	In1	In2	(In3)*	(In4)*	In5	In6	Out1	Out2	Out3	Out4
1.	**Contact strip triggered during the closing process**	**Mechanical stop function activated**	**Pane frame stopped**										
1.1	*Automatic sequence*	Motor stop	Motor stopped	1	0	1	0	0	0	1	0	0	**0**
1.2	*Automatic sequence*	Quick Stop sensor responds	sinking signal flank on (In3)	1	0	**1-0-1**	0	0	0	1	0	0	1
2.	**Auto reference movement**	**Safeguard referenced**	**Safeguard ready for operation**										
2.1	*Automatic sequence*	Reference movement	Safeguard moves	1	0	1	0	0	0	1	0	0	**1**
2.2	*Automatic sequence*	Reference point reached	Rising signal edge on (In4)	1	0	1	**0-1-0**	0	0	1	0	**1**	**0**
2.3	*Automatic sequence*	Move to initial position	Safeguard moves	1	0	1	0	0	0	1	0	**0**	**1**
2.4	*Automatic sequence*	Initial position reached	Motor stopped	1	0	1	0	0	0	1	0	**1**	**0**

Continue with step 3 "Switch on safeguard or restart and operation" and new record selection

*) These inputs deliver an output signal due to the internal motor wiring

Tab. 4.2: Stop function sequence

Reference safeguard and error acknowledgment (Switch release on/off):

- Reaching the reference point after a reference movement can be detected through a positive edge on "In4: reference" and a positive signal on "Out3: Position reached".
- Reaching the initial position after reaching the reference point can be detected through a positive edge on "Out3: Position reached" and no signal on "Out4: Drive moves".
- A reference run is automatically done after an interruption of the 24 VDC signal connection to customer on-site controls, when the reference position and after acknowledgment of errors, that result in the loss of the reference position. Interrupting the 230 VAC power supply to safety devices during maintenance of the 24 VDC connection does not initiate an automatic new referencing. Acknowledging errors that do not cause loss of the reference position also do not trigger an automatic reference run.

5 Option Safe Torque Off (STO)

5.1 Overview

There is no electrical insulation of the motor from the power supply. There is no protection against dangerous bodily currents. The complete equipment needs to be galvanically separated from mains during maintenance and repair works.

There is no control of the stand-still position.

The servo motor is optionally available with the "Safe Torque Off" function. In the "STO" state, the drive generates no torque and thus provides protection against unintentional startup. This condition is monitored internally within the drive.

In many applications there is no need of a mains contactor because of this integrated safety function. The shut-off is performed on two channels. The safety function is obtained even with the event of a single error, because of this redundancy.

The shutdown channels are completely checked on each shut-down / restart. Therefor no undetected error-burst may happen. In addition to this, the test can be forced through input operation. In the event of an error, the restart is inhibited until the freedom from errors is detected.

5.2 When do I need the option Safe Torque Off (STO)?

For MecLock safeguards, no safe drive concept is necessary as anti-pinch protection is ensured purely mechanically. For certain applications or scenarios, however, the STO engine option is a useful addition to the drive:

Example 1 - Emergency stop and bus communication:

In an emergency stop scenario, without the STO option, the entire fieldbus communication to the machine must be interrupted in order to prevent the transmission of control commands to the motor control.

With the STO option, it is sufficient to switch off only the two actively connected STO inputs. In the monitored state STO, the drive can no longer generate torque and thus offers protection against unwanted startup. So fieldbus communication can remain without risk here.

Example 2 - Laser application and light gap:

In the case of laser applications, unintentional opening of the safeguard during the machining process can result in serious injury or danger to the operating personnel. By means of the STO option and its direct control by the laser source, an unwanted opening can be ruled out since the opening of the safeguard is only possible after an active connection of the STO inputs.

5.3 Shut-down inputs

Diversity through two different shut-down inputs:

- Safe start inhibitor by interrupting the anode potential of the output-stage optocoupler
- Controller enable through microprocessor function

Design characteristics of shut-down inputs:

- Both inputs work according to the quiescent current principle, blackout of the control voltage causes the shut-down.
- Monitoring of the plausibility of the two inputs: both inputs need to be switched within a given time.
- Internal monitoring of the shut-down function through read-back function.
- Short-circuit detection can be activated.

5.4 Requirements on equipment side

- Grounding of the control circuit.
- Use of separated sheathed cables for the shut-down cables or activated short-circuit detection. For the use of the short-circuit detection, the switches need to be adjusted in such a way, that they switch with a timely difference.

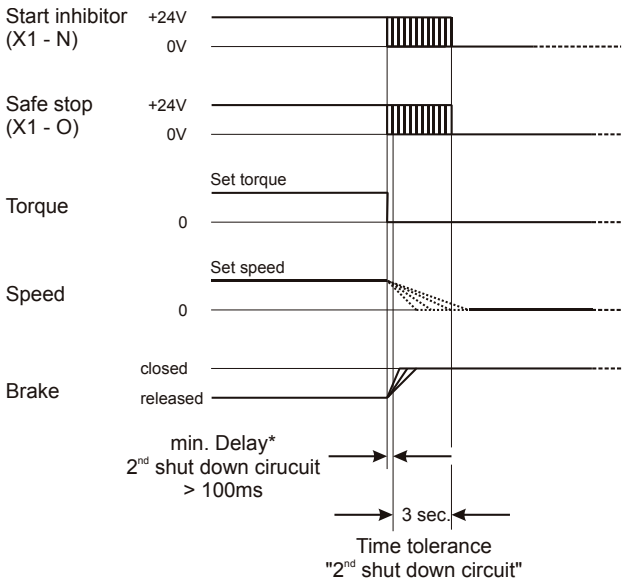
5.5 State transition Operational >> Safe stop

On motors with built-in standstill brake, the brake falls in. The standstill brake is not designed to brake down the motor frequently. Recurring load brake action may not be accomplished with the standstill brake.

It is irrelevant which of the two inputs "start inhibitor" or "safe stop" is switched first to 0V. The motor drive is disabled as soon as one of the inputs is switched.

There must be a delay time of min. 100ms between the two input transitions with activated short-circuit detection.

Error messages regarding the function "Safe torque off" can be acknowledged only when the state "STO" is reached distinct, i.e. inputs "Start inhibitor" and "Safe stop" both are 0V.



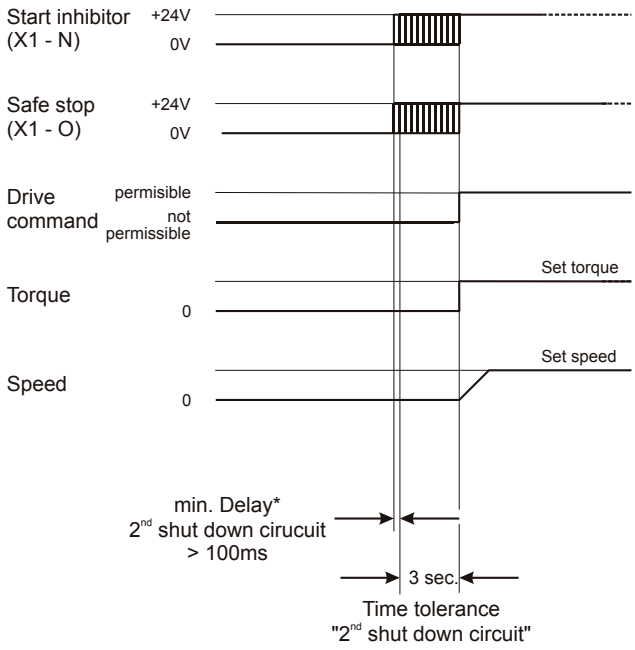
*) with activated short-circuit detection

Fig. 5.1: State transition Operational >> Safe stop

5.6 State transition “Safe stop” >> Operational

It is irrelevant which of the two inputs “start inhibitor” or “safe stop” is switched first to 24V. The motor drive is enabled as soon as both of the inputs are switched, if this happens within 3 seconds.

There must be a delay time of min. 100ms between the two input transitions with activated short-circuit detection.



*) with activated short-circuit detection

Fig. 5.2: State transition “Safe stop” >> Operational

Error messages regarding the function “Safe torque off” can be acknowledged only when the state “STO” is reached distinct, i.e. inputs “Start inhibitor” and “Safe stop” both are 0V.

5.7 Safety information about the servo motor with STO according to EN ISO 13849-1:2007

Category: 3

Performance level: e

PFH [1/h]: $3{,}71 \times 10^{-8}$

MTTFd [a]: 100 (high)

DC: 92,85 (medium)

Service life [a]: 20

6 Programming set and software ServoLink

The following instructions are intended as a help to the use of the programming set at electrically driven MecLock safeguards. If you have any questions, please contact Strasser GmbH!

DANGER

RISK OF ELECTRICAL SHOCK OR ELECTROCUTION!
Incorrect connection of cables and connectors can lead to electrical shock or electrocution with serious or fatal injuries.
Work on the electrical parts of the safeguard must only be done by qualified personnel.

Components of programming set:

Fig. 1: Programming set

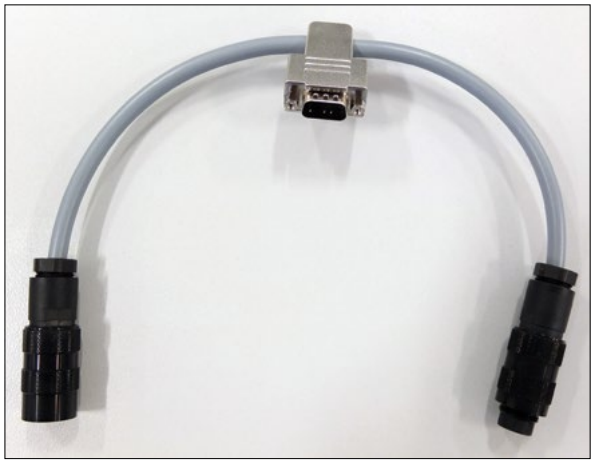


Fig. 2: T adapter

Fig. 3: USB serial adapter device

Fig. 4: Serial cable 9 pin

Fig. 5: ServoLink program CD

Do not use force! Take care that no pins of the connectors are broken or bent.

Connection Installation

Step 1:

1.1 Remove the signal cable from the signal connector X1 on the motor.

1.2 Connect the T adapter (see Fig. 2) with the signal connector X1 and the signal cable.

1.3 Connect the serial cable (see Fig. 4 - 2x 9 pin femal connectors) with the T adapter (see Fig. 2).

1.4 Connect the free serial cable connector (see Fig. 4) with the USB serial adapter device (see Fig. 3).

1.5 Connect the USB serial adapter device (see Fig. 3) with your notebook (not included). Take care that you use a USB port for connecting on your notebook because the USB plug will also fit in the network connection socket (funny error)!

1.6 The USB serial adapter device will be installed as COM port on your notebook.

1.7 Install the ServoLink software on your notebook (you will need administrator privileges for successful installation).

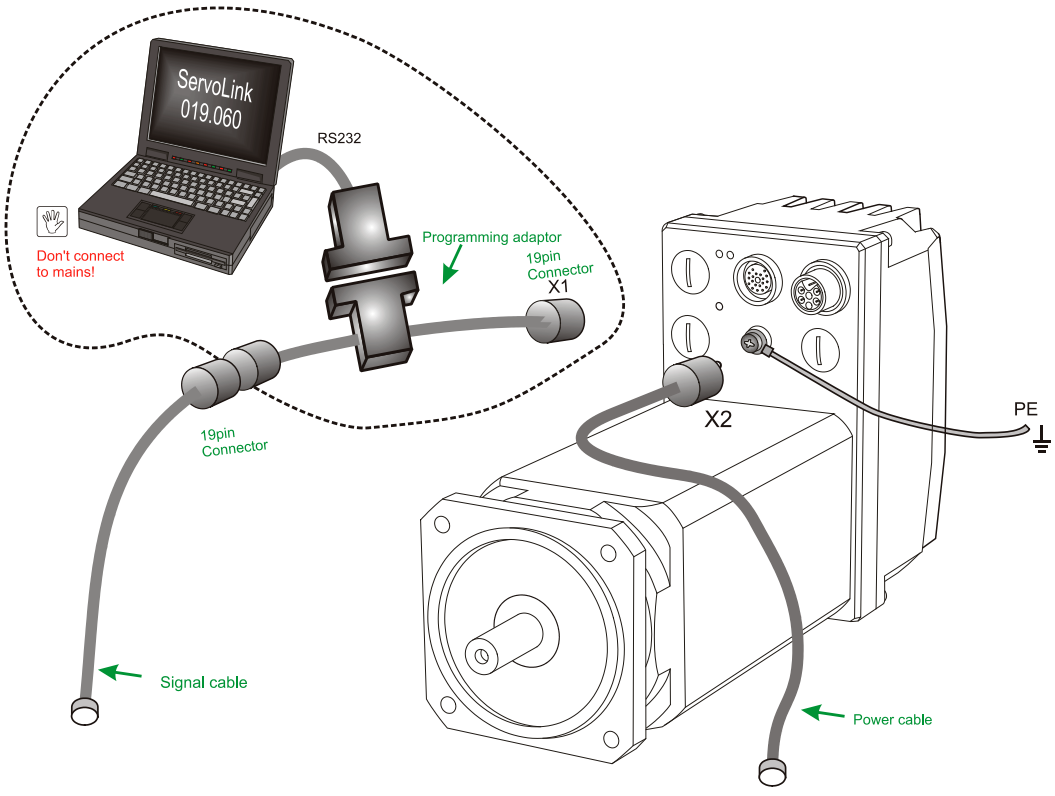


Fig. 6: Programming set installation

ServoLink software

Step 2:

Start the ServoLink program (see Fig. 7 and Fig. 8):

Fig. 7: Program icon

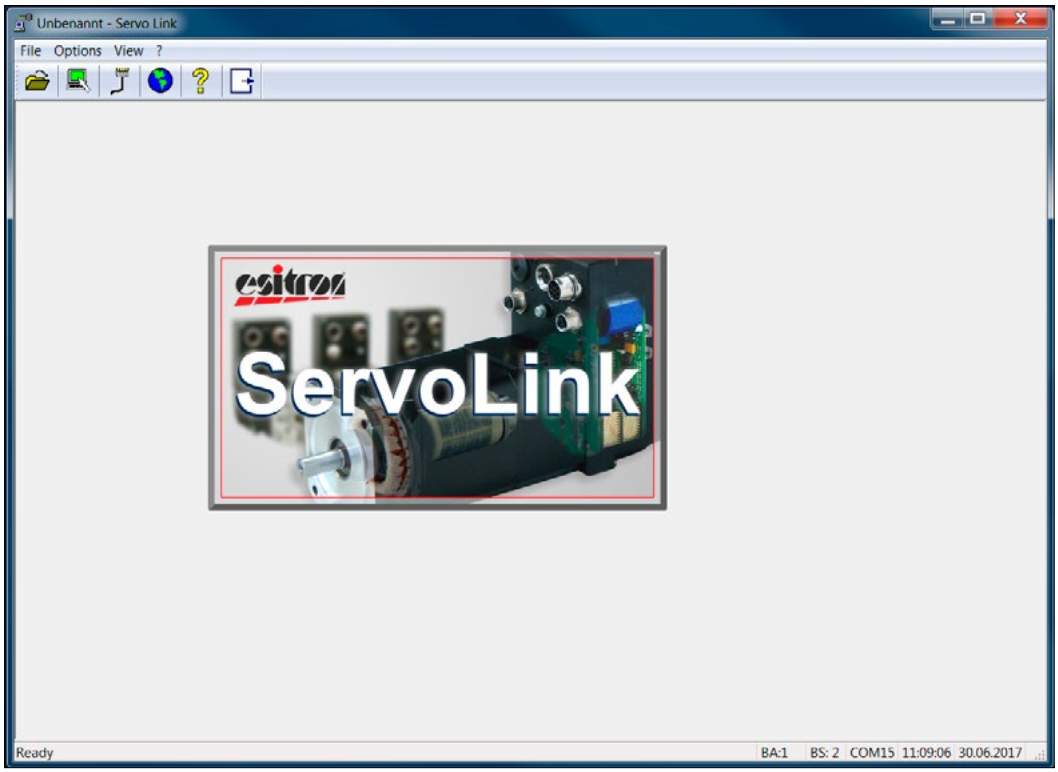


Fig. 8: Program interface

Step 3:

Change the default language from German to English (see Fig. 9 and Fig. 10):

Change default language

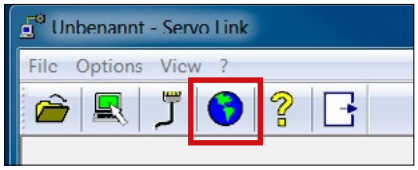


Fig. 9: Change language

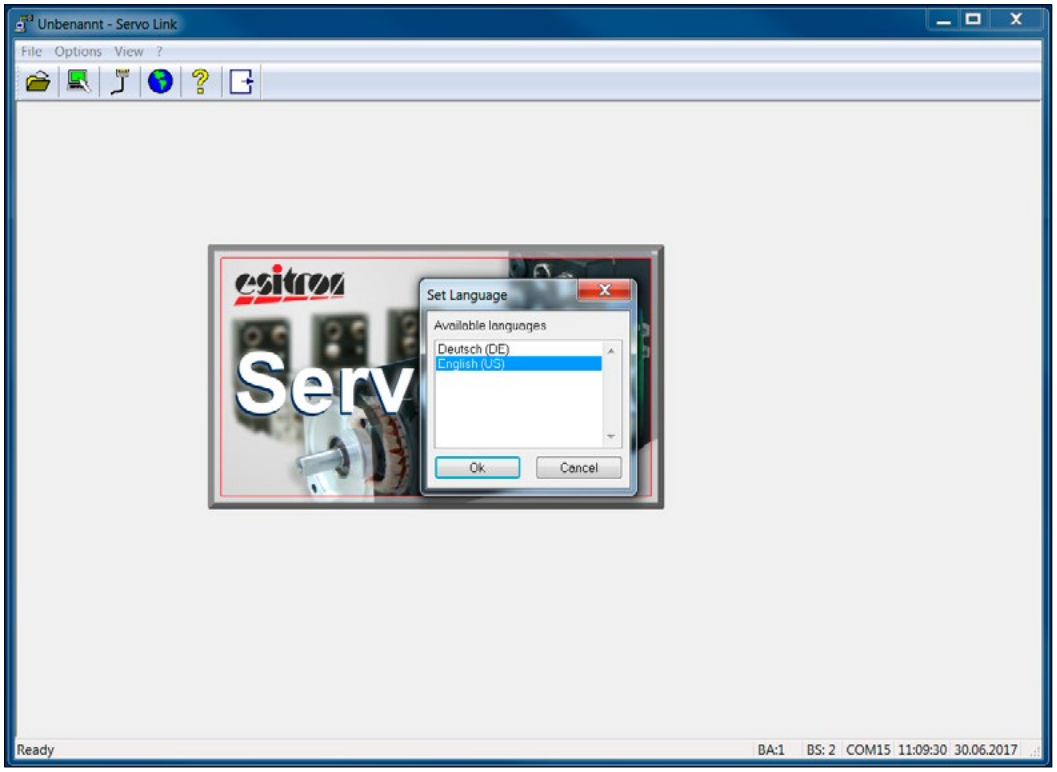


Fig. 10: Set language dialog box

COM port selection

Step 4:

4.1 Check the available COM ports and COM port numbers in the windows device manager (control panel) (see Fig. 11).

Fig. 11: Windows device manager - COM ports (example)

4.2 Check the choosen COM port number (see Fig. 12 and Fig. 13).

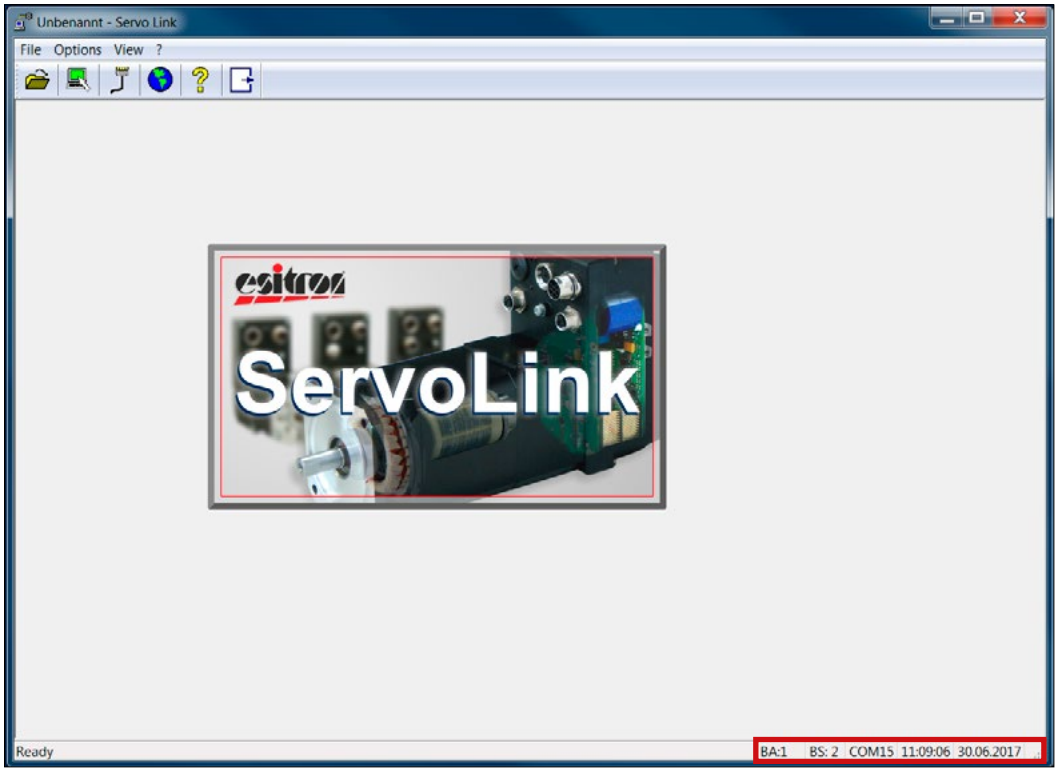


Fig. 12: Active COM port in ServoLink program interface

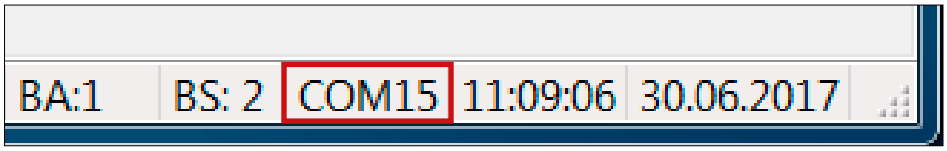


Fig. 13: Status bar: active COM port

4.3 Change the active COM port number (only if necessary) to one of the listed COM port numbers in windows device manager (see Fig. 14 and Fig. 15).

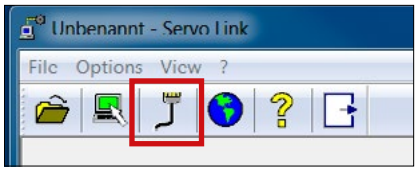


Fig. 14: Change active COM port

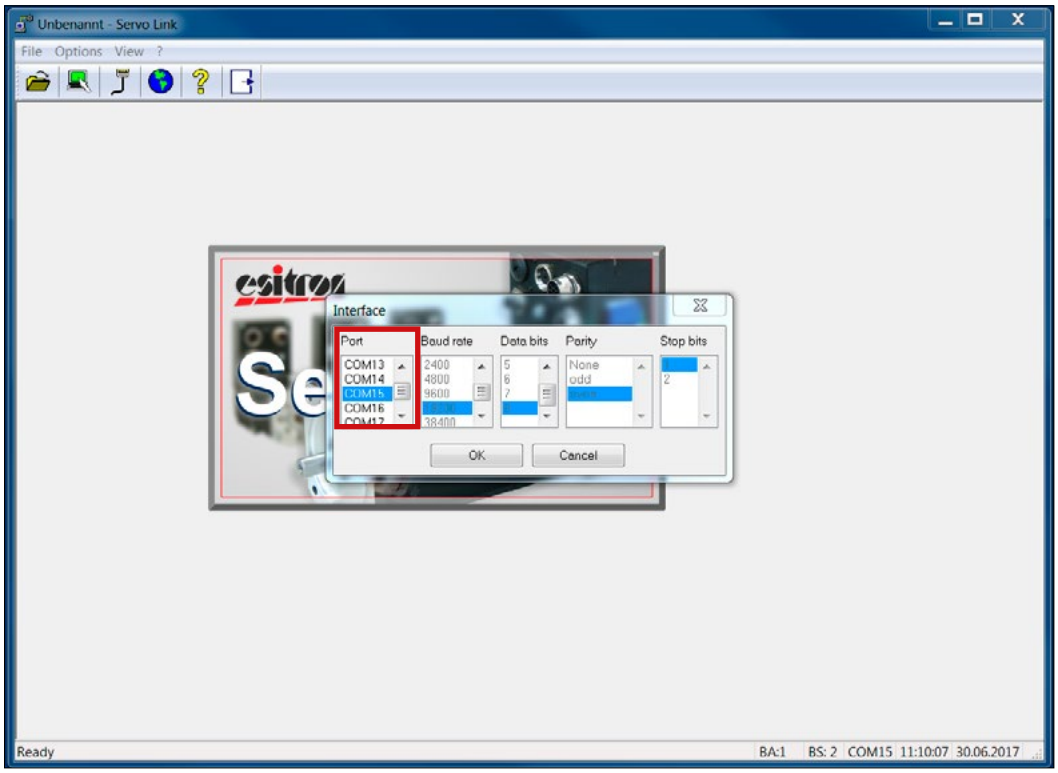


Fig. 15: Set active COM port number dialog box

Password for advanced settings

Step 5:

5.1 Open the menu `[File]` in the program interface (see Fig. 16).

5.2 Select the menu item `[Enter password ...]` (see Fig. 16).

Fig. 16: Password menu item for advanced settings

5.3 Enter and confirm the password **3467** for advanced settings (see Fig. 17).

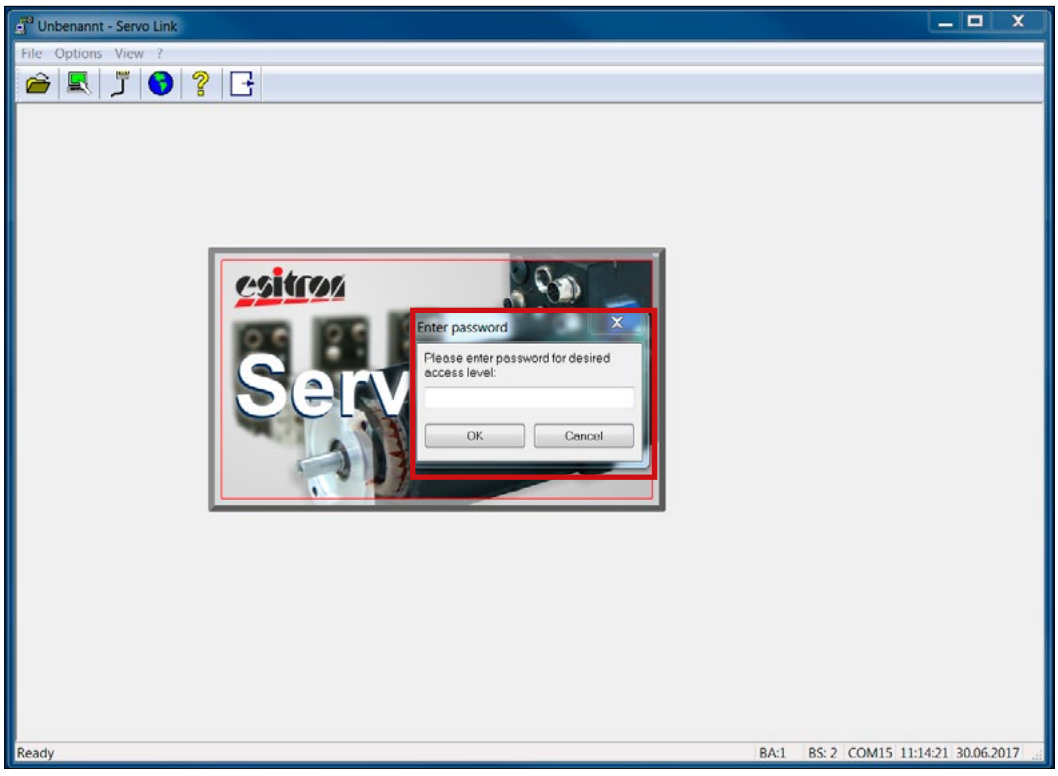


Fig. 17: Password dialog box

User interface (I)

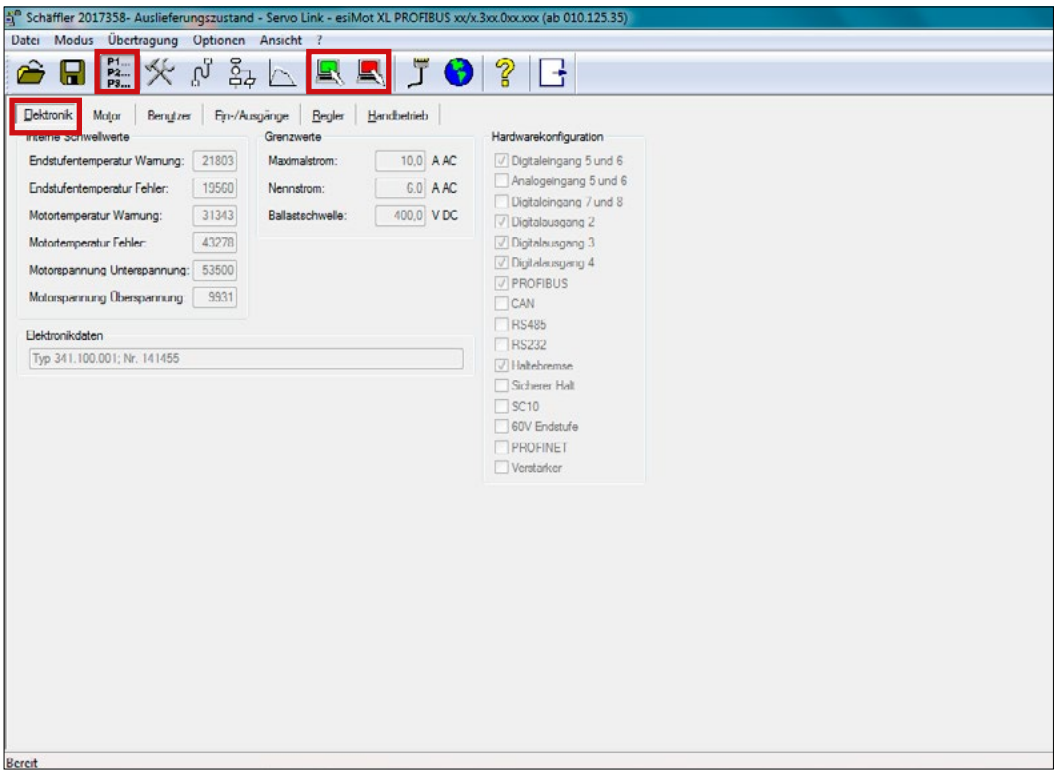


Fig. 18: 'Electronic'-tab in 'Parameter'-screen - parameter download to PC (green) and upload to motor control (red)

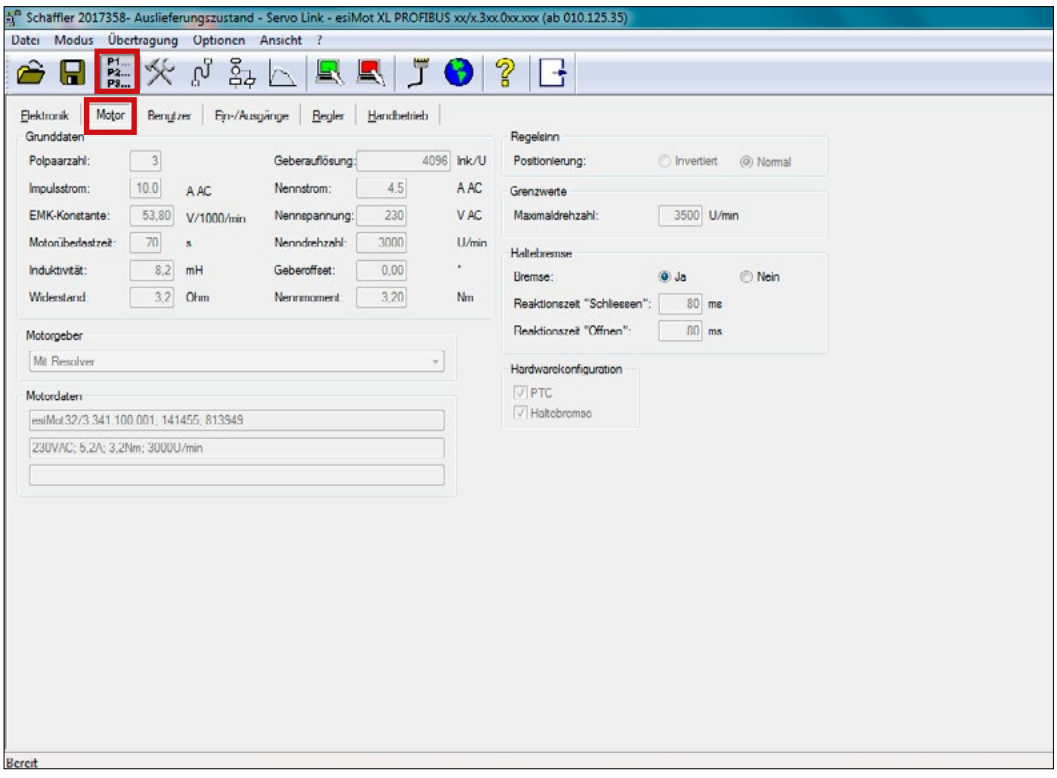


Fig. 19: 'Motor'-tab in 'Parameter'-screen

User interface (II)

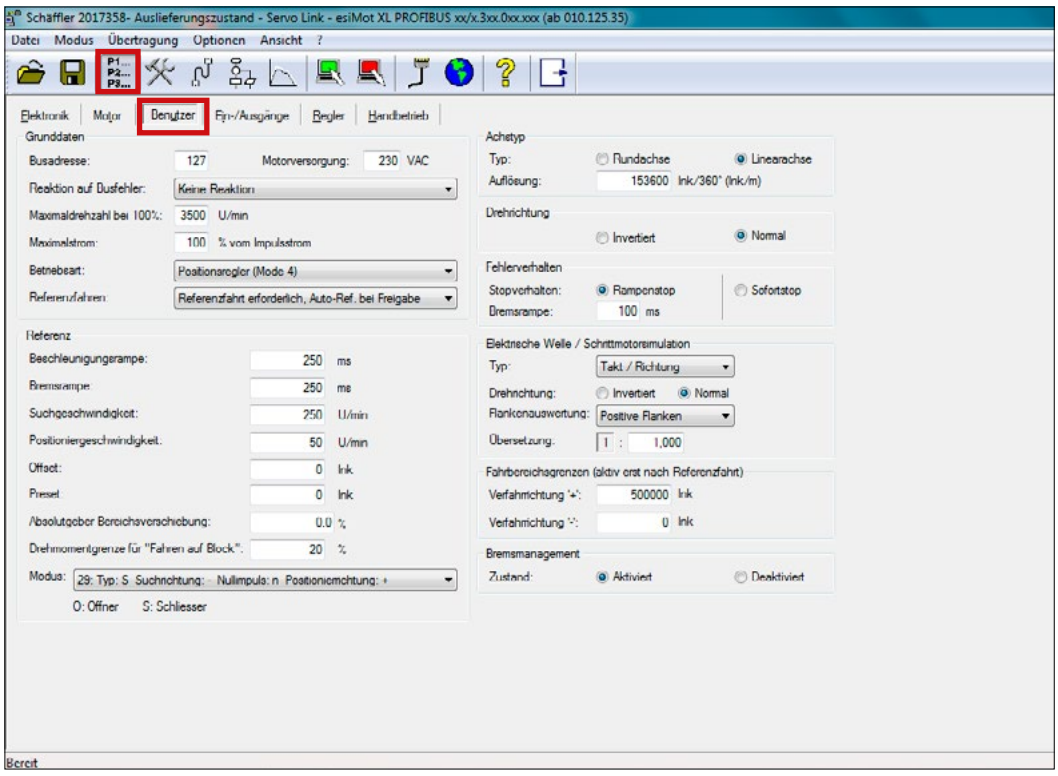


Fig. 20: 'User'-tab in 'Parameter'-screen

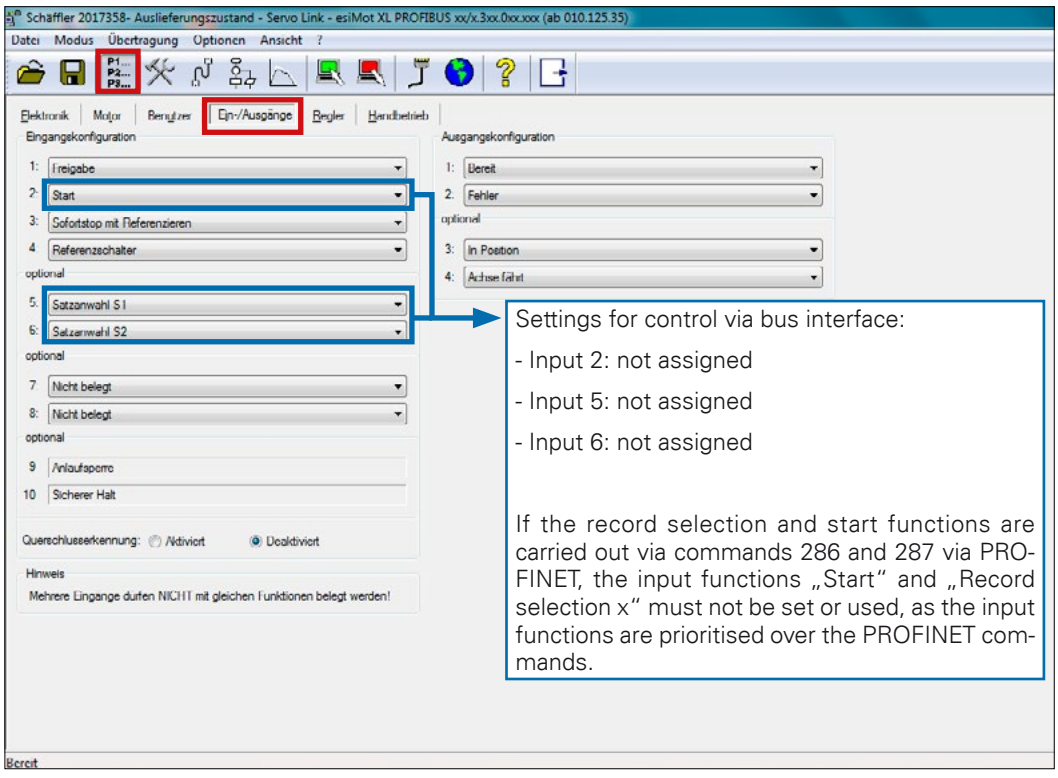


Fig. 21: 'Motor'-tab in 'Parameter'-screen

User interface (III)

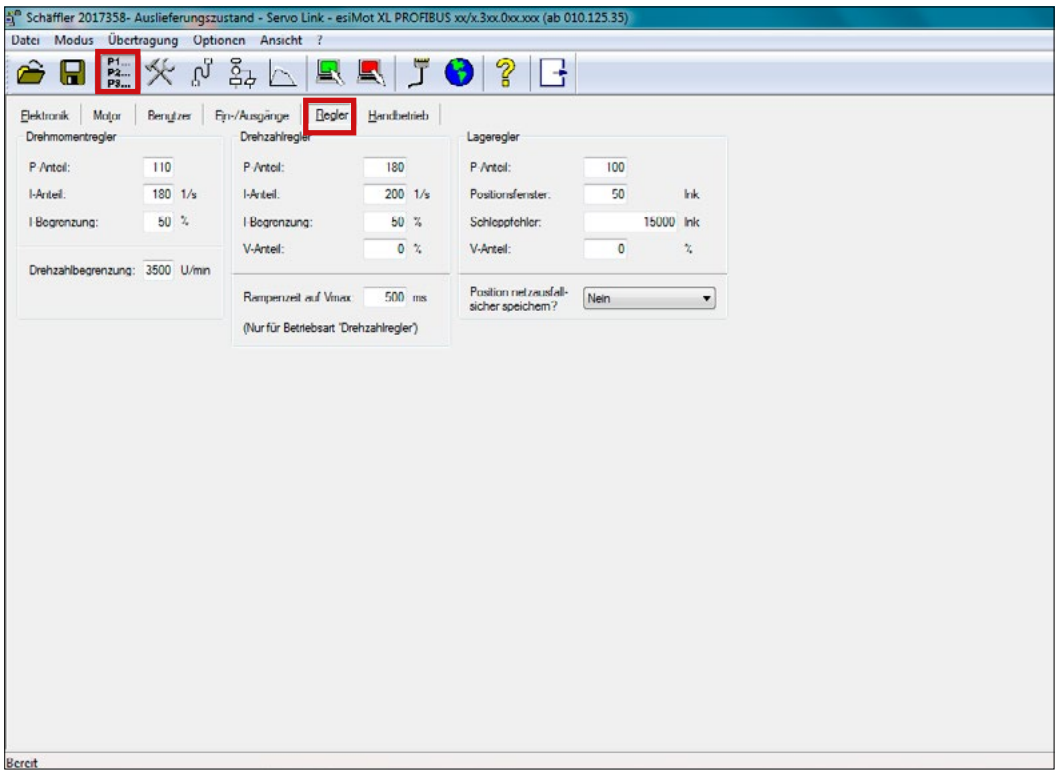


Fig. 22: 'Regulator'-tab in 'Parameter'-screen

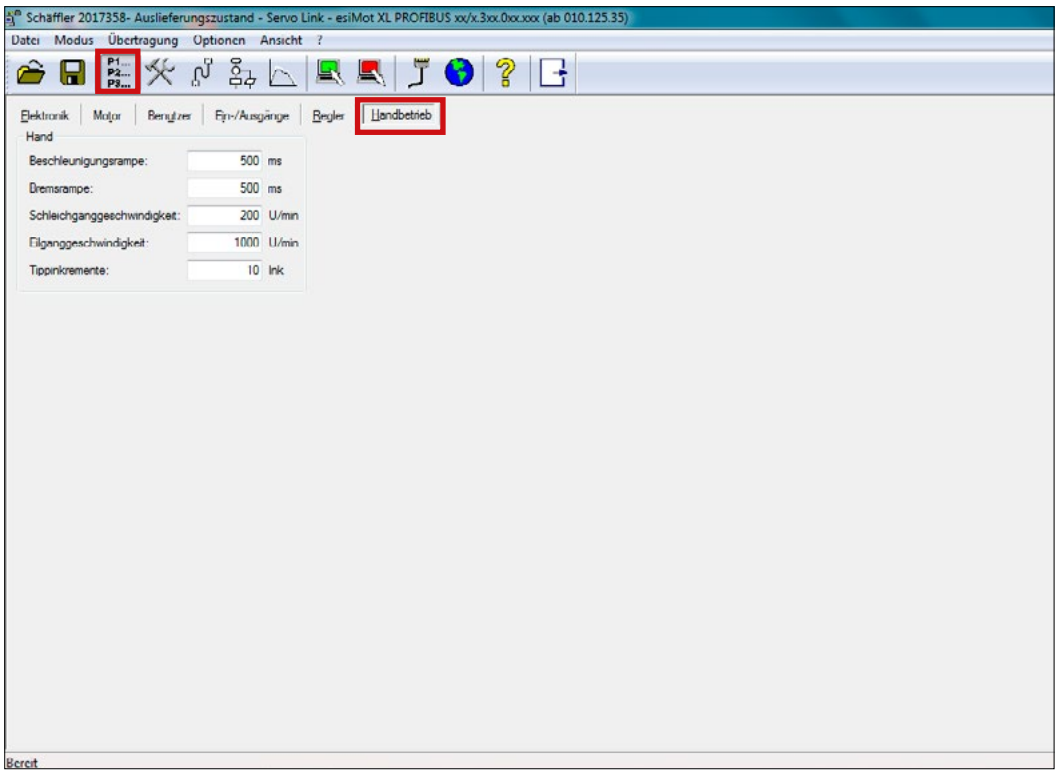


Fig. 23: 'Jog mod'-tab in 'Parameter'-screen

User interface (IV)

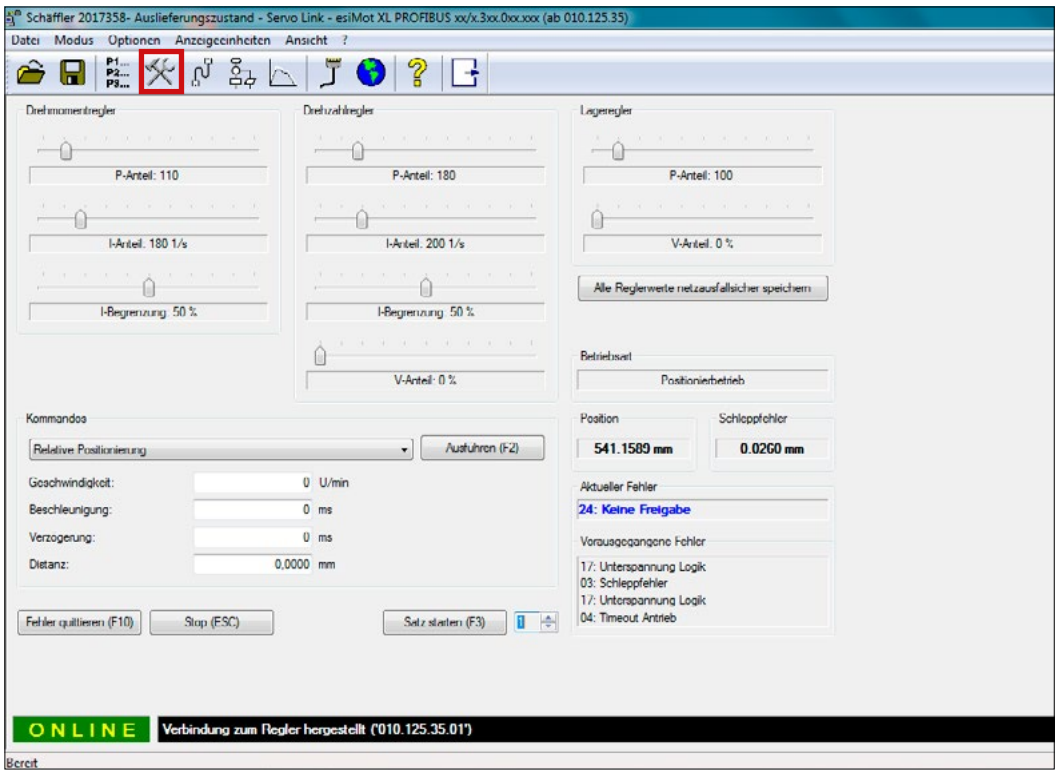


Fig. 24: Parameters for the torque, speed and position controller in 'Setup'-screen

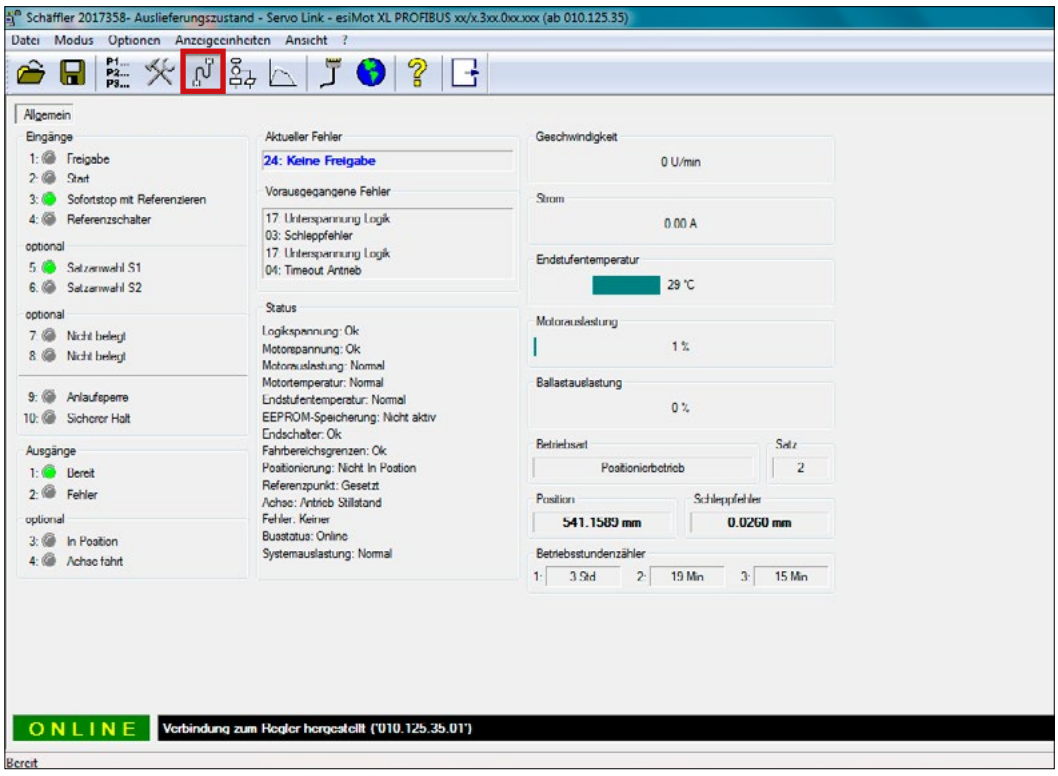


Fig. 25: 'General'-tab in 'Diagnostics'-screen

User interface (V)

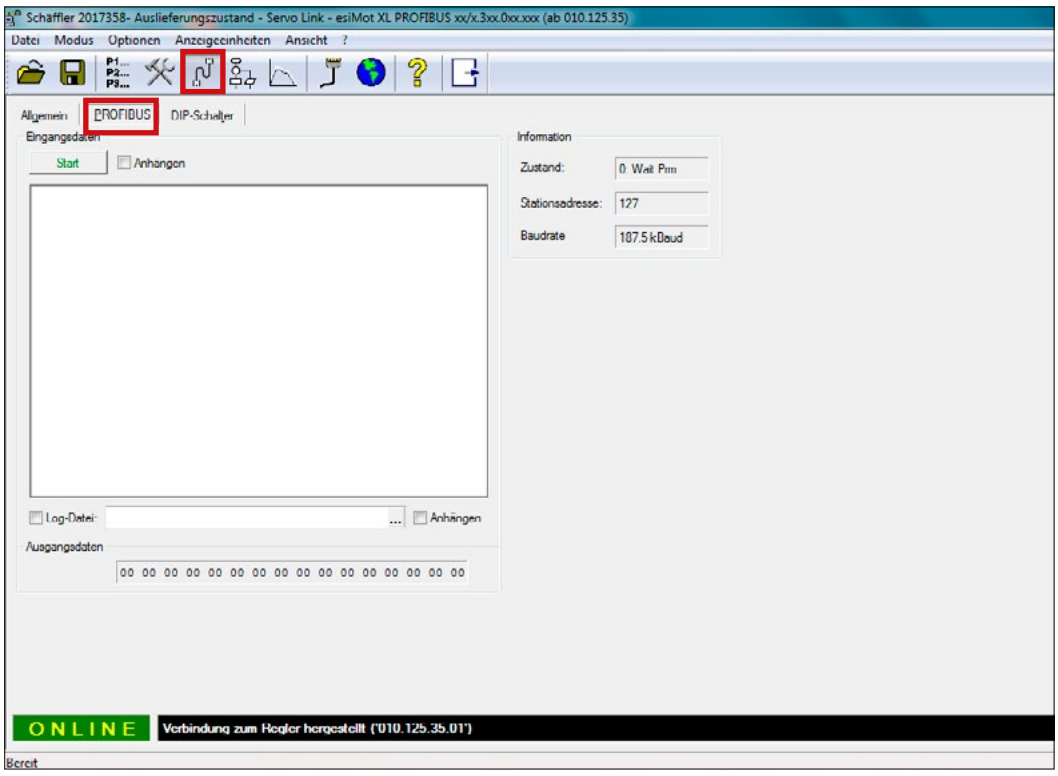


Fig. 26: 'PROFIBUS'-tab in 'Diagnostics'-screen (only with PROFIBUS DP interface)

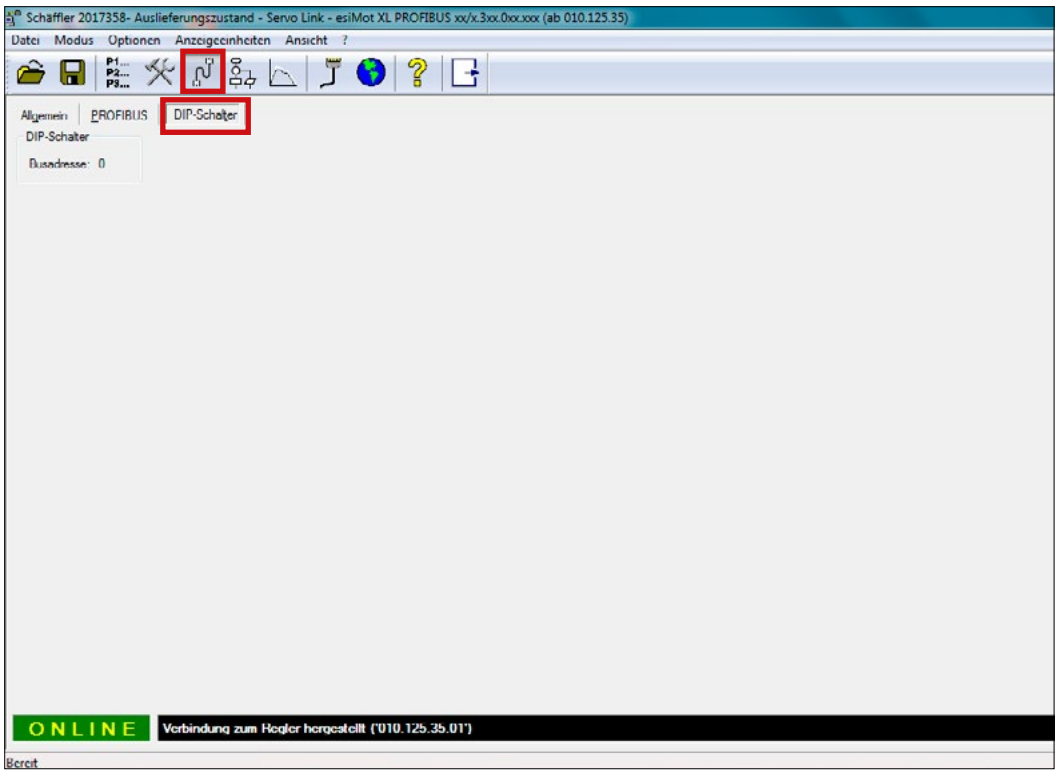


Fig. 27: 'DIP-Switch'-tab in 'Diagnostics'-screen (only with PROFIBUS DP interface)

User interface (VI)

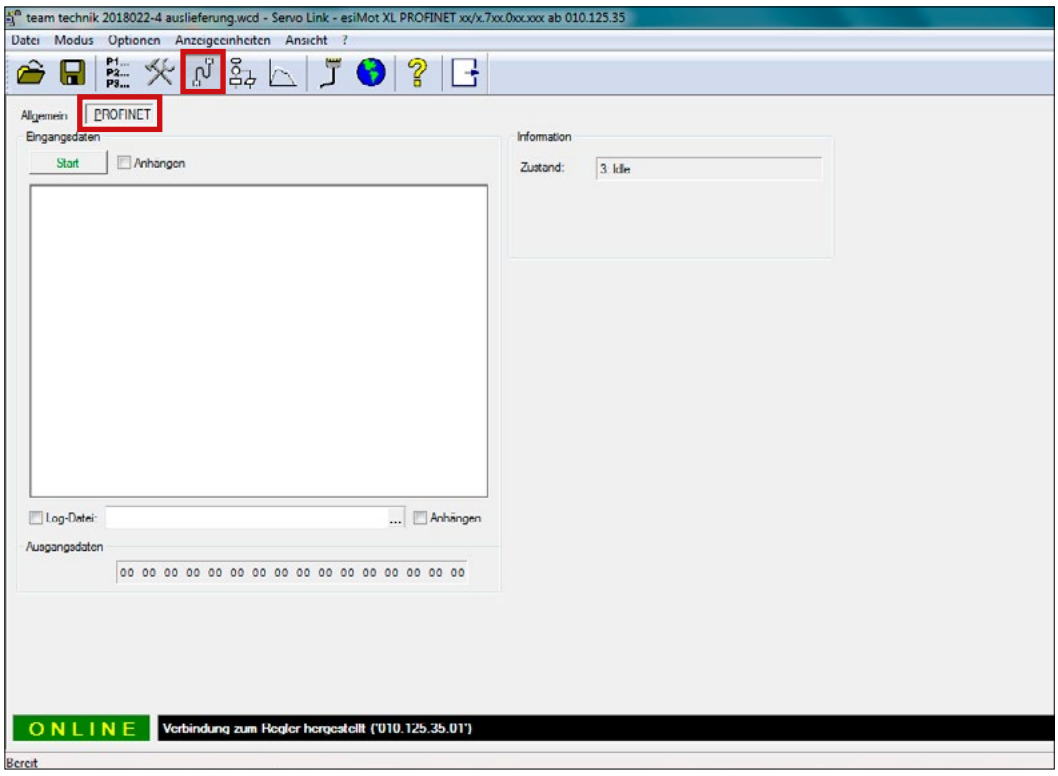


Fig. 28: 'PROFINET'-tab in 'Diagnostics'-screen (only with PROFINET interface)

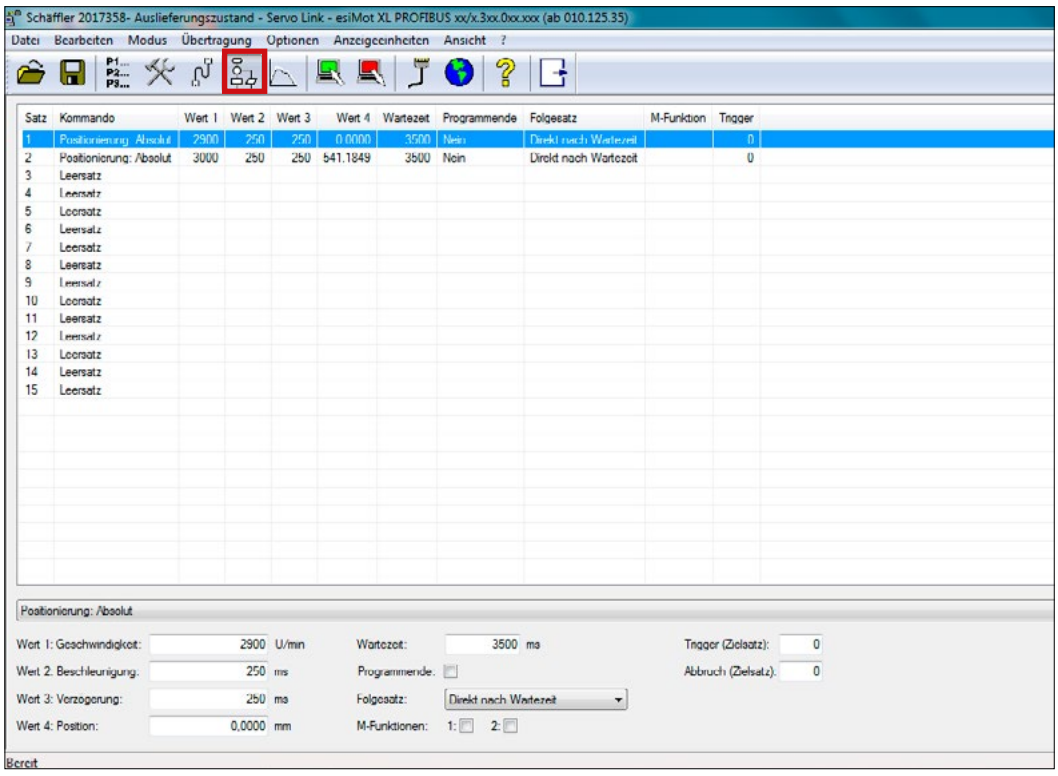


Fig. 29: 'Program'-screen

User interface (VII)

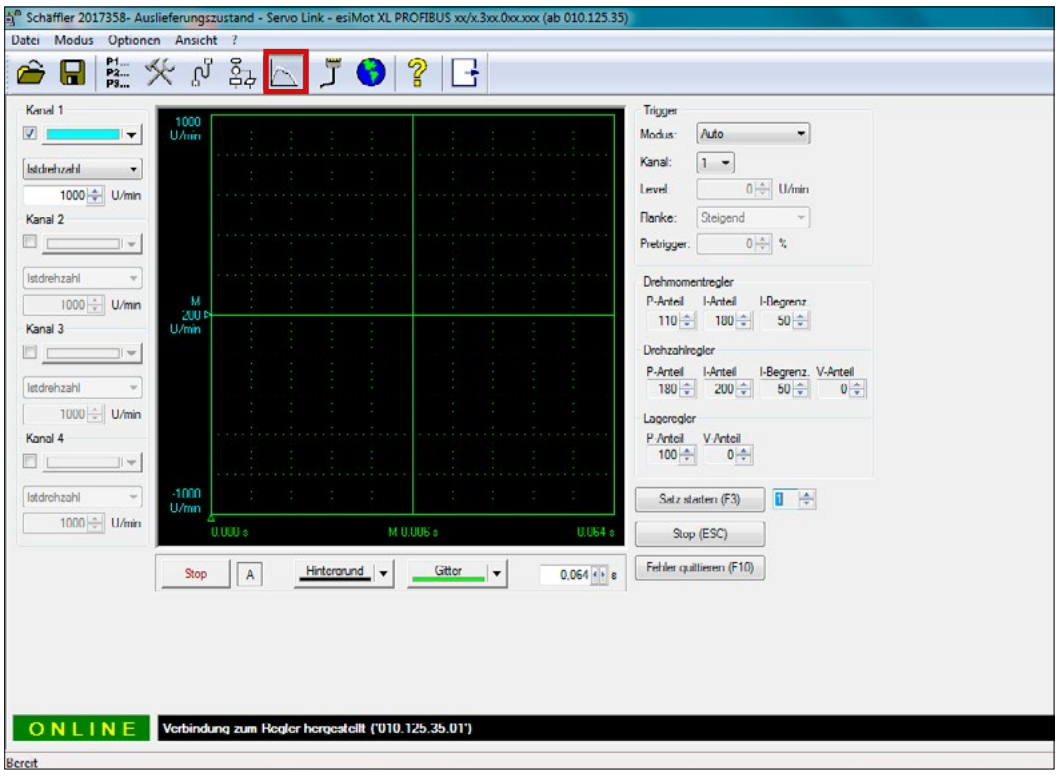


Fig. 30: 'Oscilloscope'-screen

7 Control via PROFIBUS DP / PROFINET

For servo motors with **PROFIBUS DP** interface, the **bus address must be set** using the ServoLink software in the parameter screen in the "User" tab. For servo motors with **PROFINET** interface, the **address is assigned automatically** via the fieldbus.

The **signal cable** of the servomotor **must be installed** for the power supply of the logic part on the servomotor, even when controlled via fieldbus. If the inputs on the logic part of the servomotor are controlled on the hardware side, then the **hardware-side circuit always has priority over a software-side control** via the fieldbus.

7.1 Integration in the IO system

The servo motor is integrated into the PROFIBUS DP / PROFINET system via the included GSD/GSDML-files. The name and IP address of the servo motor are projected here and stored inside the esiMot.

7.1.1 Communication via function blocks

There is a library of function blocks (FBs) provided in the scope of delivery for consistent control of the servo motor and integration to the Windows-Programming software "Step 7" and also "TIA Portal".

7.1.1.1 Installation of the library

The provided library is installed the following way:

1. Put the CD in your drive
2. Start the "SIMATIC Manager"
3. Dearchive the library "esimot.zip" from the CD with "Dearchive" in the menu "File"

The library "esiMotKonsistent" is now integrated in Step 7.

7.1.1.2 Integration of functions blocks and user data types into a project

Function block	Meaning
FB0 fbEsiMot	Function block for consistent control of the esiMot

The following steps are required to use the function block and the data structure in a STEP 7 project:

1. Open the target project
2. Open the library "esiMot" with "open" in the menu "file"
3. Copy the function block from the library "esiMotKonsistent" into the target project

Copy for each esiMot the entity-DB of the function block FB0 (fbEsiMot) to the project.

7.1.2 How to work with the function block fbEsiMot

7.1.2.1 Parameter definition

The following table shows the parameters of the function block fbEsiMot:

Name	Type	Data type	Meaning
iiAddress	IN	INT	Base address esiMot (peripheral area)
iiCommand	IN	INT	Command to be processed
biDisableTXData	IN	BOOL	Suppress copying the TX-datafields
iioCommError	IN OUT	INT	Current communication error

This function block needs to be called cyclic.

7.1.2.2 Static variables

The following table shows the static variables of an entity data block of the function block fbEsiMot.

Name	Type	Data type	Meaning
stEsiMotTX	stat	STRUCT	Send buffer to the esiMot (used internal)
stEsiMotTXPos	stat	STRUCT	Structure for send data field commands 3/4/12/13/303 “Start positioning to absolute position [Increments]” / “Start relative positioning [Increments]” / “Absolute positioning rotary axis + [Increments]” / “Absolute positioning rotary axis - [Increments]”
stEsiMotTXRef	stat	STRUCT	Structure for send data field command 5 “Start referencing”
stEsiMotTXStop	stat	STRUCT	Structure for send data field commands 7/279 “Stop with ramp“ and “Stop positioning (with brake ramp) with error message”
stEsiMotTXHand	stat	STRUCT	Structure for send data field commands 10/11 “Jog mode positive direction” and “Jog mode negative direction”
stEsiMotTXRegType	stat	STRUCT	Structure for send data field command 256 “Set regulator type”
stEsiMotTXSollPreset	stat	STRUCT	Structure for send data field command 258 “Position controller set position = preset position [Increments]”
stEsiMotTXSoll	stat	STRUCT	Structure for send data field commands 266/268/366/368 “Set value speed controller [rpm]” and “Set value current controller [1/10 Ampere]”
stEsiMotTXSollProz	stat	STRUCT	Structure for send data field commands 267/269/367/369 “Set value speed controller [%]” and “Set value current controller [%]”
stEsiMotTXSollDZLim	stat	STRUCT	Structure for send data field command 371 “Set value current regulator [1/10 Ampere], continuous setpoint acceptance, speed limited”
stEsiMotTXSollDZLimProz	stat	STRUCT	Structure for send data field command 372 “Set value current regulator [%], continuous setpoint acceptance, speed limited”
stEsiMotTXPar	stat	STRUCT	Structure for send data field command 275 “Write Parameter”

Name	Type	Data type	Meaning
stEsiMotTXRiSinn	stat	STRUCT	Structure for send data field command 283 "Adjust rotational direction" (see chapter 3.3.29)
stEsiMotTXPosRec1	stat	STRUCT	Structure for send data field command 284 "Write record 1" (see chapter 3.3.30)
stEsiMotTXPosRec2	stat	STRUCT	Structure for send data field command 285 "Write record 2" (see chapter 3.3.31)
stEsiMotTXProgCtrl	stat	STRUCT	Structure for send data field command 286 "Program control" (see chapter 3.3.32)
stEsiMotTXOutputs	stat	STRUCT	Structure for send data field command 288 "Set digital output free usage" (see chapter 3.3.34)
stEsiMotTXCfgStat	stat	STRUCT	Structure for the send data field command 292/293 "Configure status field iStat" / "Configure status field lStat" (see chapter 3.3.38)
stEsiMotTXDiag	stat	STRUCT	Structure for send data field command 400

7.1.2.3 Calling a function block in IL/SCL

The following variables and data blocks have been defined in the symbol table:

Name	Data type	Remarks
iEsiMotCommand1	INT	Flag word
iCommError1	BOOL	Flag variable for current communication error (access to P- area)
dbEsiMot	fbEsiMot/FB0	Entity block

Call in IL - view:

```
CALL fbEsiMot, dbEsiMot(
iiAddress := 0, // esiMot at P-Address 0
iiCommand := iEsiMotCommand1,
biDisableTXData := FALSE
iioCommError := iCommError1))
```

Call in SCL - view:

```
fbEsiMot.dbEsiMot(
iiAddress := 0, // esiMot at P-Address 0
iiCommand := iEsiMotCommand1,
biDisableTXData := FALSE
iioCommError :=iCommError1);
```

If this call is executed cyclic, the function block fbEsiMot process the data transfer at a change of the variable "iEsiMotCommand1" and the data is accessible for the user in dbEsiMot in the respective structures. After execution of a command "iEsiMotCommand1" should be set to command 0 (no command) to update the status of the esiMot continuously.

The variable iioCommError delivers the return value for system functions DPRD_DAT() and DPWR_DAT().

If more than one esiMot is to be controlled with a Profinet master, each esiMot needs an entity block of the type fbEsiMot and fbEsiMot needs to be called cyclic with each of these entity blocks.

7.1.2.4 Memory requirements

Type	Size	Meaning
Code memory	ca. 3.138 bytes	Function block fbEsiMot
Local data	30 Bytes	Function block fbEsiMot

Type	Size	Meaning
Global data	x Data blocks 420 Bytes	Entity block of fbEsiMot; Memory requirement per used esiMot !

All data from and to the esiMot are stored in entity block. Therefor no additional user specific buffers are necessary.

7.1.2.5 Examples

There is an example program “esiMotPNBsp” provided in the package.

Install the example program as followed:

1. Insert the CD into the drive
2. Start the “SIMATIC Manager”
3. Dearchive the file “esimotbs.zip” from the CD with “Dearchive” in the menu “File”

The example program is now available in STEP 7.

In the example program, the variable table “vatEsiMot” can be found where all the data structures of the entity data block dbEsiMot1 as well as all other necessary variables required to control the function module (command, command echo, etc.) are defined. All esiMot function can easily be tested using these variable table and the functions “Controlling / monitoring”.

7.2 Command interface

7.2.1 Protocol

The execution of a command is triggered by a change of the data field "command" in bytes 0 – 1. At this change, the parameter data in byte 2 – 15 are transferred. After interpretation of the command through esiMot, the command is answered-back to the input data field of esiMot and the respective data are provided in the status field.

When using working mode 1 (inconsistent data transfer), take care the parameter data are already written at the time of the command change. Furthermore the master must write the command with a word access in the data field.

7.2.2 Send data field to esiMot

The send data field to esiMot is arranged the following way:

Byte-No.	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Function	Command		Parameter to "Command"													

7.2.3 Description of the parameter data fields

7.2.3.1 Login (Command 1)

<no parameters>

This command is required to change parameters. The user must log in prior to a parameter write command (275).

7.2.3.2 Logout (Command 2)

<no parameters>

7.2.3.3 Start referencing (Command 5)

Byte-No.	**2-13**	**14-15**
Function	Reserved	Referencing mode
Range	-	29 Hex: 0x1D
Unit	-	-

7.2.3.4 Jog mode positive direction (Command 10)

Byte-No.	2-5	6-7	8-15
Function	Reserved	0: Creep speed 1: Rapid traverse	Reserved
Range	-	-	-
Unit	-	-	-

7.2.3.5 Jog mode negative direction (Command 11)

Byte-No.	2-5	6-7	8-15
Function	Reserved	0: Creep speed 1: Rapid traverse	Reserved
Range	-	-	-
Unit	-	-	-

7.2.3.6 Deactivate regulator (Command 20)

<no parameters>

7.2.3.7 Activate regulator (Command 21)

<no parameters>

7.2.3.8 Error acknowledge (Command 22)

<no parameters>

The acknowledgement of an error is done with rising and falling edge. That means the signal must toggle. A pulse is required!

7.2.3.9 Program control (Command 286)

This command is effective only if no input function is assigned to record selection.

Byte-No.	2	3	4	5-15
Function	Record pointer number	Reset	Store	Reserved
Range	1-15	-	-	-
Unit	-	-	-	-
Remarks	This record number determines the record which is taken for the next regular read-in of records.	If byte 3 is different to 0, the record pointer is set to 1, if the esiMot is stopped, e.g. with stop input or missing "Enable". Byte 3 has higher priority than byte 2.	If byte 4 is different to 0, the record data in the RAM are stored safe to power outage. Byte 4 has higher priority than byte 3.	

7.2.3.10 Start Program (Command 287)

<no parameters>

This command is effective only if not input function is assigned to START.

7.2.3.11 Error acknowledge without enabling regulator at falling edge (command 294)

<no parameters>

A rising edge acknowledges errors.

7.2.4 Input data field of esiMot

The input data field of the esiMot is assigned like follows:

Byte-No.	**0-1**	**2-15**
Function	Echo Command	Status to "Command"

The echo of the command can be read back through the variables "dbEsiMot.stEsiMotRX.iCommand".

7.2.5 Description of the status fields

7.2.5.1 Default data field

Byte-No.	**2-5**	**6-7**	**8**	**9**	**10**	**11**
Function	IStat	iStat	Status of inputs	Status of outputs	Error status	Actual current
Unit	see below	see below	-	-	-	-

Byte-No.	**12**	**13**	**14**	**15**
Function	Status 1	Status 2	Status 3	Status 4
Unit	-	-	-	-

7.2.5.1.1 Actual position [IStat] (Bytes 2-5)

Actual position as signed long value (+/- 31 Bit).

7.2.5.1.2 Actual speed [iStat] (Bytes 6-7)

Actual speed as signed integer value (+/- 15 Bit).

7.2.5.1.3 Status of inputs (Byte 8)

Bit-No.	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Function	Input 1	Input 2	Input 3	Input 4	Input 5	Input 6	Input 7	Input 8

7.2.5.1.4 Status of outputs (Byte 9)

Bit-No.	0	1	2	3	4	5	6	7
Function	Output 1	Output 2	Output 3	Output 4	Speed window reached	Reserved	Reserved	Reserved

7.2.5.1.5 Error status (Byte 10)

For meaning of error numbers see table.

Bit-No.	0	1	2	3	4	5	6	7
Function	Error number 0-255							

7.2.5.1.6 Actual current (Byte 11)

Bit-No.	0	1	2	3	4	5	6	7
Function	Actual current in 1/10 A							

7.2.5.1.7 Status 1 (Byte 12)

Bit-No.	0	1	2	3	4	5	6	7
Function	Ready	Homed	Regulator active	Position reached	Axle in motion	Ramp	Brake	Error (all)

Description of signals

Bit-No.	Signal name	Signal state and description	
0	Ready	1	Ready
		0	Not ready
1	Homed	1	Valid referencing performed
		0	No valid reference point present
2	Regulator active	1	The regulator is active
		0	The regulator is not active
3	Position reached	1	The target window has been reached and the regulator is still active.
		0	The axle is outside the target window.
4	Axle in motion	1	The axle moves with a speed • Greater 4 increments per 20ms (3rpm with 4096 incr./rev encoder; 6rpm with 2048 incr./rev encoder.) • Greater 1 increment per 100ms with hall sensors.
		0	"The axle is motionless or moves with a speed • Less than 4 increments per 20ms (3rpm with 4096 incr./rev encoder; 6rpm with 2048 incr./rev encoder.) • Less than 1 increment per 100ms with hall sensors.
5	Ramp	1	A positioning is running.
		0	No positioning is running. However, the axle could be moving in speed or torque mode.
6	Brake	1	The (optional) parking brake is closed.
		0	The (optional) parking brake is open.
7	Error (all)	1	An error exists. See error state (Byte10) and error list.
		0	No error exists.

7.2.5.1.8 Status 2 (Byte 13)

Bit-No.	0	1	2	3	4	5	6	7
Function	Reserved	Enable	Stop	Limit switch +	Limit switch -	Software limit of travel +	Software limit of travel -	Error (acknowledge required)

Description of signals

Bit-No.	Signal name	Signal state and description	
0	STO active (only drives with STO option)	1	STO active
		0	STO not active
1	Enable	1	The regulator is enabled
		0	The regulator is blocked
2	Stop	1	An input signal “stop” is present. i.e. the appropriate input is 0V.
		0	No input signal “stop” is present. i.e. the appropriate input is 24V.
3	Limit switch +	1	An input signal “limit switch +” is present. i.e. the appropriate input is 0V.
		0	No input signal “limit switch +” is present. i.e. the appropriate input is 24V.
4	Limit switch -	1	An input signal “limit switch -” is present. i.e. the appropriate input is 0V.
		0	No input signal “limit switch -” is present. i.e. the appropriate input is 24V.
5	Software limit of travel +	1	The maximal travel range in positive direction is reached.
		0	The position is inside the parametrized limits of travel.
6	Software limit of travel -	1	The maximal travel range in negative direction is reached.
		0	The position is inside the parametrized limits of travel.
7	Error (acknowledge required)	1	An error which requires an error acknowledge exists. See error state (Byte10) and error list. See also respective unit manual.
		0	No error exists.

7.2.5.1.9 Status 3 (Byte 14)

Bit-No.	0	1	2	3	4	5	6	7
Function	2: Torque regulation 3: Speed regulation 4: Positioning							

7.2.5.1.10 Status 4 (Byte 15)

Bit-No.	0	1	2	3	4	5	6	7
Function	Record pointer (0-15)				reserved			

The default data field is used for the esiMot answer-back, if not one of the following commands is sent.

7.2.5.1.11 Read record 1 (command 290)

A record consists of several parameters. Not all can be set with one command. An “empty set” is needed to delete a record.

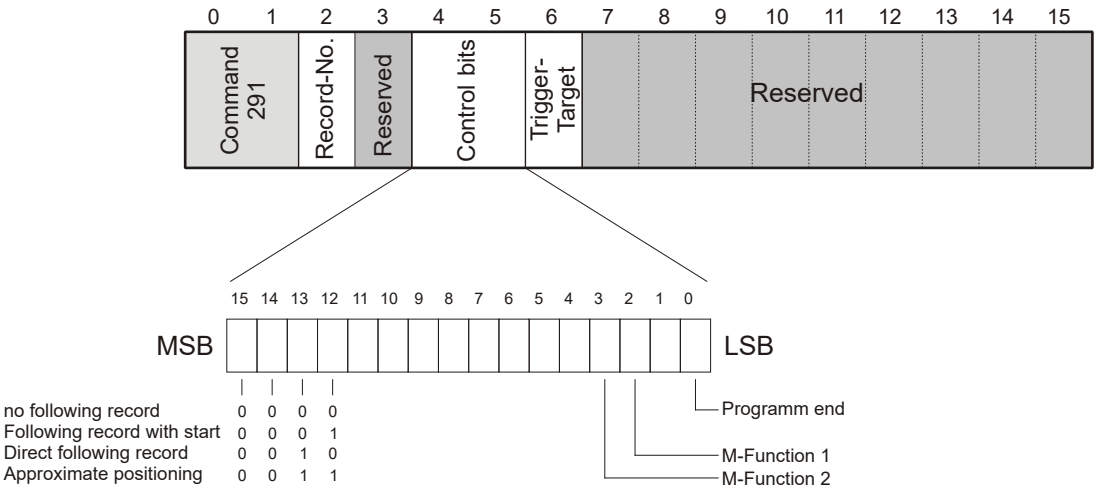
Byte-No.	**2**	**3**	**4-5**	**6-7**	**8-9**	**10-11**	**12-15**
Function	Record number	Type of record	Record value	Acceleration time	Deceleration time	Delay time	Position resp. Increments
Range	1-15 0x01 ... 0x0F	0: Absolute positioning	1-10.000 Hex: 0x01 ...2710 [rpm]	10-10.000 Hex: 0x0A ... 2710	10-10.000 Hex: 0x0A ... 2710	0-10.000 Hex: 0x00 ... 2710	-2.140 Mio. ... +2.140 Mio. Hex: F80723100 ... 7F8DCF00
		1: Relative positioning (Distance)	1-10.000 Hex: 0x01 ...2710 [rpm]				
		2: Set speed value in rpm	-10.000 ...10.000 Hex: D8F0...2710 [rpm]	-	-		
		3: Set speed value in %	-100 ... +100 Hex: 0x00...0x64 [%]				
		4: Set current value in 1/10 Ampere	-140 ... 140 Hex FF74 ...0x8C [A/10]				
		5: Set current value in %	-100 ... +100 Hex: FF9C...0x64 [%]				
		6: Absolute positioning rotary axis +	1-10.000 Hex: 0x01 ...2710 [rpm]	10-10.000 Hex: 0x0A ... 2710	10-10.000 Hex: 0x0A ... 2710		max. rotary resolution see parameter: 85
		7: Absolute positioning rotary axis -	1-10.000 Hex: 0x01 ...2710 [rpm]				
		127: Referencing	Referencing mode	-	-		-
		255: Empty record					
Unit	-		-	ms	ms	ms	Increments

7.2.5.1.12 Read record 2 (command 291)

Byte-No.	2	3	4-5	6	7-15
Function	Record number	Reserved	Control bits	Trigger a target record No.	Reserved
Range	1 - 15		LSB: Bit 0 – Program end Bit 3 – M Function 1 Bit 4 – M Function 2 Bit 12-15: 0: no following record 1: following record with start 2: direct following record (without start) 3: Approximate positioning	0-15 0x00 ...0x0F 0 means no trigger ☛ In a record started by trigger function a trigger address may not be set again	-
Unit			-	-	-

Write record 2nd command

Read record 2nd command



7.3 Examples

7.3.1 Execution of the command sequence “Activate regulator”, “Program control” and “Start program”

After switching on, the drive should be activated with feedback and a positioning block should be selected and executed. Currently “0” (no command) is transmitted in the output data field to the esiMot.

The following data field must be transferred:

7.3.1.1 Command “Activate regulator” (21)

Byte-No.	0-1	2-15
Value	Command = 21	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 21	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit “Regulator active” status 1, bit 2 can now be queried whether the command was executed correctly.

7.3.1.2 Record selection with command “Program control” (286)

Byte-No.	0-1	2	3	4	5-15
Value	Command = 286	Record number: 1-15	Reset = 0	Store = 0	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 286	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bits 0-3 of status 4, the selected record number can now be queried and checked.

7.3.1.3 Command “Start program” (287)

Byte-No.	0-1	2-15
Value	Command = 287	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 287	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit 3 of status 1, “Position reached” of the drive can now be queried and checked.

If you want to select a different record, you can continue with record selection via command “Program control” (286).

7.3.1.4 Execution of the command “Error acknowledge”

If an error requiring acknowledgment is signaled in status field 7 of status 2 in the default data field, the error must be acknowledged and the drive must be reactivated.

The value “500” shall be written to parameter number 100. Currently, a value other than “Fault” (22) is in the output data field transferred to esiMot (edge triggered command transfer).

7.3.1.5 Command “Error acknowledge without enabling regulator at falling edge” (294)

Byte-No.	0-1	2-15
Value	Command = 294	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 294	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

Status bit 7 of status 2 can now be queried as to whether the error requiring acknowledgment has been acknowledged.

7.3.1.6 Command “Activate regulator” (21)

Byte-No.	0-1	2-15
Value	Command = 21	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 21	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit “Regulator active” status 1, bit 2 can now be queried whether the command was executed correctly.

7.3.2 Modification of records

Records can be processed via fieldbus with the commands “Write record 1” (284) and “Write record 2” (285) and “Read record 1” (290) and ”Read record 2” (291).

This example shows the reading and changing of the data of record 1.

7.3.2.1 Command “Login” (1)

In order to be able to store changed data or parameters protected against mains failure, a logon must be made via fieldbus.

Byte-No.	**0-1**	**2-15**
Value	Command = 1	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 1	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

The login at the drive is now done.

7.3.2.2 Command “Read record 1” (290)

Byte-No.	**0-1**	**2**	**3-15**
Value	Command = 290	Record 1	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2**	**3**	**4-5**	**6-7**	**8-9**	**10-11**	**12-15**
Function	Echo command = 290	Record number: 1	Type of record	Record value	Acceleration time	Deceleration time	Delay time	Position resp. Increments

7.3.2.3 Command “Write record 1” (284)

The modified data are transferred to the servo motor as follows:

Byte-No.	**0-1**	**2**	**3**	**4-5**	**6-7**	**8-9**	**10-11**	**12-15**
Function	Command = 284	Record number = 1	Type of record	Record value	Acceleration time	Deceleration time	Delay time	Position resp. Increments

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 284	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

7.3.2.4 Store record data safe to power outage with command “Program control” (286)

Byte-No.	0-1	2	3	4	5-15
Value	Command = 286	-	Reset = 0	Store = 1	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 286	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

7.3.2.5 Command “Logout” (2)

Byte-No.	0-1	2-15
Value	Command = 2	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 2	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

The logout at the drive is now done.

7.3.3 Safeguard switching on, restarting and operation

7.3.3.1 Command “Activate regulator” (21)

Byte-No.	**0-1**	**2-15**
Value	Command = 21	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 21	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit “Regulator active” status 1, bit 2 can now be queried whether the command was executed correctly.

7.3.3.2 Command “Start referencing” (5)

If automatic homing is parameterized after switching on, this command need not be executed!

Byte-No.	**0-1**	**2-13**	**14-15**
Value	Command = 5	-	Referencing mode = 29

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 5	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In the status bits status 1 “Homed” bit 1 and “Position reached” bit 3, it is now possible to query whether the command was executed correctly.

7.3.3.3 Record selection record 1 (starting position) with command "Program control" (286)

Byte-No.	0-1	2	3	4	5-15
Value	Command = 286	Record number: 1	Reset = 0	Store = 0	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 286	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bits 0-3 of status 4, the selected record number 1 can now be queried and checked.

7.3.3.4 Command "Start program" (287)

Byte-No.	0-1	2-15
Value	Command = 287	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 287	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit 3 of status 1, "Position reached" of the drive can now be queried and checked.

7.3.3.5 Record selection record 2 (open), record 3 (close) or record 4 (intermediate position) with command "Program control" (286)

Byte-No.	0-1	2	3	4	5-15
Value	Command = 286	Record number: 2, 3, 4	Reset = 0	Store = 0	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 286	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bits 0-3 of status 4, the selected record number 2, 3 or 4 can now be queried and checked.

7.3.3.6 Command “Start program” (287)

Byte-No.	0-1	2-15
Value	Command = 287	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 287	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit 3 of status 1, “Position reached” of the drive can now be queried and checked.

7.3.4 Intervention in the closing safeguard during normal operation

The triggering of the closing safeguard during operation can be detected via status bit "Input 3" in the data field "Status of inputs" and the status bit "Axle in motion" in Status 1, Bit 4.

7.3.4.1 Command "Start referencing" (5)

If automatic homing is parameterized after switching on, this command need not be executed!

Byte-No.	**0-1**	**2-13**	**14-15**
Value	Command = 5	-	Referencing mode = 29

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 5	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In the status bits Status 1 "Homed" Bit 1 and "Position reached" Bit 3, it is now possible to query whether the command was executed correctly.

7.3.4.2 Record selection record 1 (starting position) with command "Program control" (286)

Byte-No.	**0-1**	**2**	**3**	**4**	**5-15**
Value	Command = 286	Record number: 1	Reset = 0	Store = 0	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	**0-1**	**2-5**	**6-7**	**8**	**9**
Function	Echo command = 286	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	**10**	**11**	**12**	**13**	**14**	**15**
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bits 0-3 of status 4, the selected record number 1 can now be queried and checked.

7.3.4.3 Command “Start program” (287)

Byte-No.	0-1	2-15
Value	Command = 287	-

After a recognized command, esiMot sends the following data field (default data field):

Byte-No.	0-1	2-5	6-7	8	9
Function	Echo command = 287	Actual position	Actual speed	Status of inputs	Status of outputs

Byte-No.	10	11	12	13	14	15
Function	Error status	Actual current	Status 1	Status 2	Status 3	Status 4

In status bit 3 of status 1, “Position reached” of the drive can now be queried and checked.

Continue with record selection record 2 (open), 3 (close) or 4 (intermediate position) via command “Program control” (286).

8 Diagnosis

8.1 Status LED's

Axis State and Errors						
Error	**Error message / Status**	**Red Led (H1)**	**Multicolour LED (H2)**	**Ready – Signal**	**Acknowledge-ment required**	**Brake mode**
0	"Enable" present, no errors (nor-mal operating mode)	Off	Green On	On	No	-
24	Enable missing	Off	fast blink Green	On	No	A
47	STO - Status information -	Off	fast blink Green	On	No	A
19	Stop without error message	Off	Blink orange 2x	On	No	-
3	Lag error	On	Blink green 1x	Off	Yes	B
5	Axle moving	On	Blink green 2x	Off	Yes	-
8, 9, 10, 11	Limit switch, Limit of travel range	On	Blink green 3x	Off	Yes	B
4	Timeout drive	On	Blink orange 1x	Off	Yes	-
20	Stop with error message	On	Blink orange 2x	Off	Yes	-
25, 28, 31-34	Wrong regulator state	On	Blink orange 3x	Off	Yes	-
6, 7	Erroneous ramp parameter	On	Blink orange 4x	Off	Yes	-
43, 44	Limit switch, direction deviant	On	Blink 3x green	Off	Yes	B
45, 46, 48	2. shutdown circuit, STO	On	Orange on	Off	Yes	B

Voltage supply and Temperature error						
Error	**Error message / Status**	**Red Led (H1)**	**Multicolour LED (H2)**	**Ready – Signal**	**Acknowledge-ment required**	**Brake mode**
12, 21	Motor temperature	On	Blink red 1x	Off	Yes	A
13, 41	Controller temperature	On	Blink red 2x	Off	Yes	B
14, 15	Power supply motor	On	Blink red 3x	Off	Yes	B
16, 17, 30	Power supply logic	On	Blink red 4x	Off	Yes	B

Communication Errors						
Error	**Error message / Status**	**Red Led (H1)**	**Multicolour LED (H2)**	**Ready – Signal**	**Acknowledge-ment required**	**Brake mode**
26, 35	Profibus/CAN-Bus off	Blink	Blink orange 1x	Off	Yes	A
22	Profibus login missing	Blink	Blink orange 2x	Off	Yes	-
27	RS232/RS485 time out	Blink	Blink orange 3x	Off	Yes	A
The LED H3 flashes if the fieldbus communication is not available or faulty.						

Internal errors						
Error	**Error message / Status**	**Red Led (H1)**	**Multicolour LED (H2)**	**Ready – Signal**	**Acknowledge-ment required**	**Brake mode**
23, 38-40, 42, 49	Error at initialisation	On	Red on	Off	Yes	B
18	EEPROM Checksum error	On	Red on	Off	Yes	B
36	Encoder error	On	Red on	Off	Yes	B

Interface status LED H3 (PROFINET only)		
LED colour	**State**	**Description**
Green	On	PROFINET – connection available
Green	Flashing	Searching for PROFINET
Orange	On	No PROFINET connection found and the parameter "reaction to bus error" is set to "no reaction".

8.2 Error and status messages

Error	Hex	Error message / Status	Error cause / remedy
0	0x00	No error	-
3	0x03	Lag error	The drive is unable to follow the set value. The drive may be sluggish or even blocked. The parameter for the allowable tracking error is set too small. The acceleration is too high. Encoder signals erroneous. Incorrect direction of control. Loop gain insufficient or excessive (drive is oscillating).
4	0x04	Timeout drive	Desired position was not reached within the given time. Tune regulator, increase position window.
5	0x05	Axle moving	It was attempt to send a new drive or positioning command while the drive was in motion.
6	0x06	Erroneous ramp parameter	The drive can't comply with the given data. The acceleration time is too long.
7	0x07	Travel for ramp to far	It's not possible to calculate the ramp for the given data.
8	0x08	Limit switch +	The drive has reached the limit switch in positive direction.
9	0x09	Limit switch –	The drive has reached the limit switch in negative direction.
10	0x0A	Limit of travel range +	The machine has reached the admissible limit of the travel of 2.140 million increments in "+". ☛ After error acknowledge the position value is set to 0.
11	0x0B	Limit of travel range –	The machine has reached the admissible limit of the travel of 2.140 million increments in "–". ☛ After error acknowledge the position value is set to 0.
12	0x0C	Motor temperature (PTC sensor)	Possible reasons for high temperature: The drive may be sluggish or even blocked, high acceleration values, high motor load with high ambient temperature … ☛ The temperature error can be acknowledged when the temperature of the motor went down again.
13	0x0D	Output-stage temperature too high	Possible reasons for high temperature: The drive may be sluggish or even blocked, high acceleration values, high motor load with high ambient temperature … ☛ The temperature error can be acknowledged when the temperature of the controller went down again.
14	0x0E	Over-voltage motor	When the motor source exceeds one of the limits defined in the settings this error is given. This can happen due to spikes in the supply voltages. In most cases, the over-voltage can also occur if the motor needs to slow down a heavy load. Check machine and state. An external ballast resistor can help.
15	0x0F	Under-voltage motor	The motor voltage undercuts the limits given by parameter. This may be caused by spikes on the supply voltage. In most cases undervoltage is a result of a heavy duty. A more powerful power supply may help. ☛ If the enable input is low, the motor-voltage is not monitored. There is no error message if the voltage is switched on ahead of a start. ☛ It takes several seconds to completely charge the DC bus. If the enable input is set too early, this error is detected as well.
16	0x10	Over-voltage logic	The logic voltage supply exceeds 30 V. In most cases this can happen due to spikes in the supply voltages.
17	0x11	Under-voltage logic	The logic supply falls below 18V. Maybe there are drop-outs in the supply voltage. Insufficient filtering of the supply voltage. A more powerful power supply may help.
18	0x12	EEPROM checksum error	Unit was switched off, during parameter input. Check parameters.
19	0x13	Stop without error message (status information)	There was a falling edge on the stop input. This can be caused by user or by a power loss on the stop input. No acknowledgement is required. To resume a start in necessary. The residual travel is proceeded.
20	0x14	Stop applied (error message)	"Stop with error message" present. There was a falling edge on the stop input. This can be caused by user or by a power loss on the stop input. Check machine and state. If no error or dangerous situation can be discovered try to acknowledge the "Stop"-error with a raising edge on the enable input. Be aware that acknowledgement can only be successful when the error which caused this state no longer exists.
21	0x15	Overload motor	Demanded motor power too high, The drive may be sluggish or even blocked, high acceleration values, high motor load with high ambient temperature …
22	0x16	Profibus login missing	Parameter setting is allowed only after login. Attempt to write parameters without prior login causes this error-message.

8 Diagnosis

Error	Hex	Error message / Status	Error cause / remedy
23	0x17	Error at initialisation	Internal initialisation error. No access to EEPROM. Please consult your dealer.
24	0x18	Enable missing	All configuration and controller data are correct. The motor control is able to operate and waits for a raising edge on the enable input.
25	0x19	Working mode wrong	A function was selected which is in the current working mode not possible. For example a position value was given to the speed regulator.
26	0x1A	Bus off-line	The connection to the fieldbus has failed. ☛ The LED H3 is flashing if the bus connection couldn't be established after power-on.
27	0x1B	RS232/RS485-Trigger timeout	RS232/RS485 interface link was disconnected, while the axle was moving.
28	0x1C	Axle not referenced	A drive command was given whilst no reference run was done.
29	0x1D	Wrong value (status Information)	An incorrect value was transmitted e.g. speed > n_{max}.
30	0x1E	Reference voltage to low	The internal reference voltage is too low. Please check the logic supply.
31	0x1F	Range of travel exceeded	The maximum countable number of increments has been exceeded.
32	0x20	Reference lost	The motor moved after the voltage was switched off. The procedure for position storage safe against power outage was not accomplished. Unsettled motor supply voltage.
33	0x21	Set value mode wrong	Clock/direction inputs were defined and it was tried to start a record or another set value was given.
34	0x22	Regulator state wrong	The response time of the brake was not taken care for and a new drive command given.
35	0x23	CANopen Error	Reserved error
36	0x24	Encoder error	A not existing encoder was selected in the parameters.
37	0x25	Reference type not supported	A not supported referencing mode was selected. No error message, only status information.
38	0x26	Resolver error	Resolver or processing faulty. Signals corrupted. Cable broken.
39	0x27	Resolver error	
40	0x28	Resolver error	
41	0x29	Output-stage overload	Possible cause: The drive may be sluggish or even blocked, high acceleration values, high load with high ambient temperature, motor faulty, cables broken …
42	0x2A	Error temperature sensor	The value is not plausible. There is a malfunction of the sensor.
43	0x2B	Limit switch + direction deviant	The limit switch in + direction was activated while the motor was moving in – direction.
44	0x2C	Limit switch – direction deviant	The limit switch in – direction was activated while the motor was moving in + direction.
45	0x2D	Missing second shutdown circuit	Possible short circuit between signals on inputs 9 + 10 (STO – Safe Torque Off).
46	0x2E	STO internal	There is an internal fault (STO).
47	0x2F	STO (status information)	The drive is in the state "STO" (no error message, status information only). No error acknowledgement is required to leave this state.
48	0x30	Short circuit STO	The switching time difference between the two STO-inputs is too short.
49	0x31	Temperature sensor motor	The temperature sensor delivers values outside the admissible range.
52	0x34	Configuration fault	There is an internal malfunction (configuration).
55	0x37	EEPROM error (Queue)	There is an internal malfunction (EEPROM).
56	0x38	EEPROM error (write)	
57	0x39	EEPROM error (read)	
58	0x3A	Ballast resistor overload	The ballast power is too big.
60	0x3C	STO not available	At least one input for STO control was activated but the drive isn't equiped with STO function.

Note: All error messages must be acknowledged by a rising edge on the enable input respectively on the acknowledge input (if assigned) or via appropriate command through fieldbus.

Note: No errors can be acknowledged while the status is "STO". Before an error is acknowledged, it is necessary to leave the "STO" state correctly. That means that the two inputs "STO 1" and "STO 2" must have been at 0 V and must be switched to 24 V while observing the time delay (when cross-circuit detection is active) and the tolerance time. Only then error messages can be acknowledged.

9 Revision overview

Date	Changes
2023-05-04	Status 4 (Byte 15) update
2023-05-04	Record selection with command “Program control” (286) update (4-7 --> 0-3)
2023-05-04	Record selection record 1 (starting position) with command “Program control” (286) update (4-7 --> 0-3)
2023-05-04	Record selection record 2 (open), record 3 (close) or record 4 (intermediate position) with command “Program control” (286) update (4-7 --> 0-3)
2023-05-04	Record selection record 1 (starting position) with command “Program control” (286) update (4-7 --> 0-3)

A Annex

STRASSER
inspired by ideas

inspired by ideas

Since 1996 Strasser GmbH has stood for new ideas in the area of automation technology and machine construction. Our guiding principle since the very beginning has been to offer our customer both economical solutions and also innovative products that inspire them. The pioneer spirit of the company founder, Karl-Heinz Strasser, led to a unique product line that reflects itself in over 25 patents.
Strasser currently offers, in addition to MecLock Safeguards, comprehensive automation solutions from components to profile systems based on carbon steel and stainless steel modular profiles. The spectrum of products ranges from base profiles, to preassembled machine frames up to complete conveyor systems or handling units.

Strasser GmbH

Heinrich-Hertz-Straße 6
88250 Weingarten
Germany

Phone + 49 (0) 7 51 / 5 61 61 - 0
Fax + 49 (0) 7 51 / 5 61 61 - 8

sales@strasser-gmbh.com
www.strasser-gmbh.com

MecLock© Safeguards - Control of electrical driven safeguards with PLC or BUS interface - Information brochure
Electrical_driven_safeguards_PLC_BUS_MAN_2307-01_EN_INT
Status: 05.07.2023