Application Note

FESTO

Synchronisation

Synchronisation of two controllers via I / O, Profibus and CAN

CMMS-AS;
CMMS-ST;
CMMP-AS

100019

Title ..... Synchronisation
Version 1.00
Document no. 100019
Original ..... en
Author ..... Festo

Last saved 28.01.2016

Copyright Notice

Legal Notice

Internet: http://www.festo.com
E-Mail: service_international@festo.com

Table of contents

Table of contents

1 Components/Software used

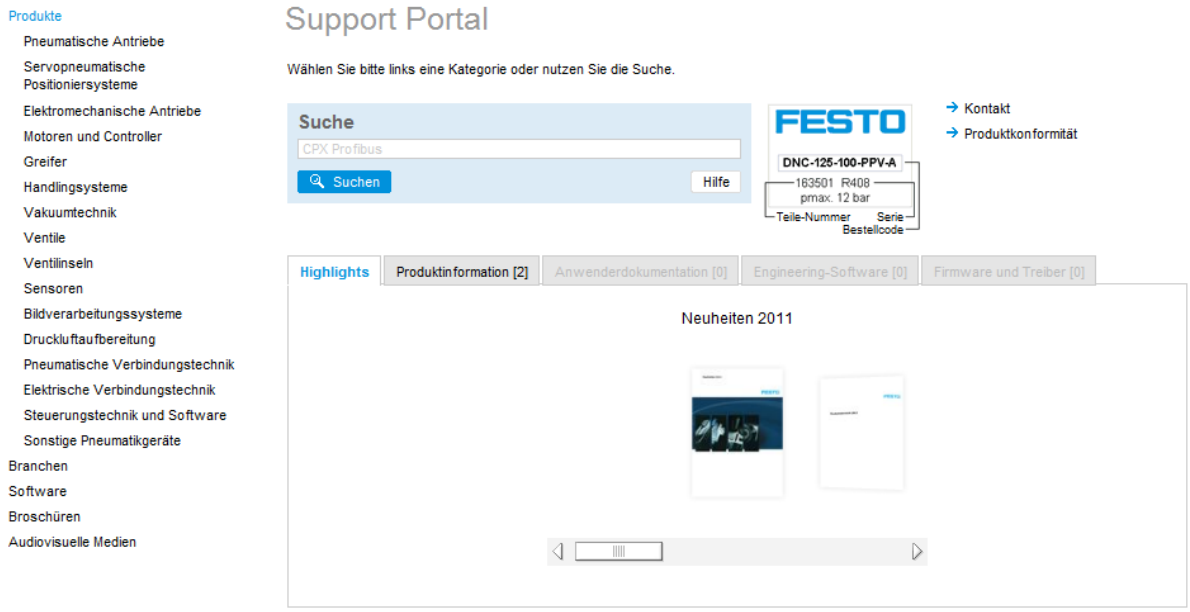
Type/Name	Version Software/Firmware	Date of manufacture
CMMP-AS-...-M3/M0	V4.0.1501.2.1	General
CMMD AS	V1.4.0.3.4	General
CMMS-AS	V1.4.0.2.4	General
CMMP-AS	V3.5.1501.5.3	General
FCT	CMMP-AS-...-M3/M0 V2.5.0.479 CMMP-AS V1.4.2.5 CMMS-AS V1.2.3.85 CMMD-AS V1.0.3.34	General

Table 1.1: 1 Components/Software used

2 Documentation

All required documentation (manuals, software, firmware, plug-ins, device description files, function blocks) for your utilised components can be found on Festo website at www.festo.com/cms/et_ee/index.htm -› “Support” tab -› select “All product information”, or click the following link:

https://www.festo.com/net/et_ee/SupportPortal/



3 Electrical installation

3.1 CMMS-ST/-AS and CMMD-AS

Carry out electrical installation of the motor controller and the motors as described in the manual.

The X10 connections are additionally required for the synchronisation mode. Connect the two interfaces using an appropriate cable. The pins are wired analogous to each other (pin 1 to pin 1, pin 2 to pin 2 ...).

Minimum wiring requirements: In order to implement synchronous operation at least the following pins must be connected between the two controllers: A, A#, B, B# and GND (pin 4). N and N# can also be connected. The interface parameters must be set up in accordance with the wiring configuration.

Ideally, the synchronisation cable has an outer shield which is connected to the housing. Additional inner shields around the twisted pairs, which are connected to GND (pin 4).

A stepper-motor encoder cable (e.g. NEBM-M12G8-E-5-S1G9 [part no. 550749]) with plug (NECC-A-S-S1G9-C2M [part no. 564264]) is recommended for use as a cable, in which the incremental encoder signals are twisted in pairs and the individual pairs are shielded.

Selection as to whether the interface will function as a master or a slave is made later via the software.

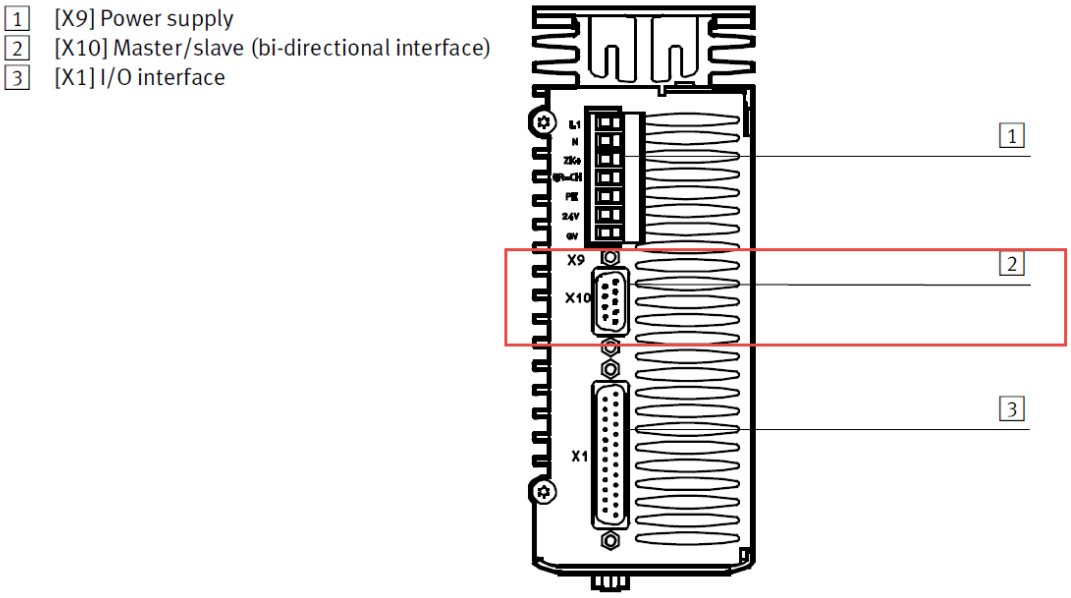


Figure 1: CMMS-AS as example

The master generates tracking signals A and B, as well as the zero pulse of an incremental encoder.

In addition to the A/B tracking signals, slave input X10 can also process CLK/DIR – pulse/direction and the CW/CCW pulse in the form of 5 V DC.

The X1 input can also process CLK/DIR – pulse/direction and the CW/CCW pulse in the form of 24 V DC.

Due to the fact that synchronisation of two Festo controllers is dealt with here, synchronisation via X10 master and X10 slave with EMMS motors will be worked through in the following.

Other synchronisation options can be looked up in the manual.

3.2 CMMP-AS

Carry out electrical installation of the motor controller and the motors as described in the manual.
The X10 and X11 interfaces are required in order to synchronise two CMMP controllers.

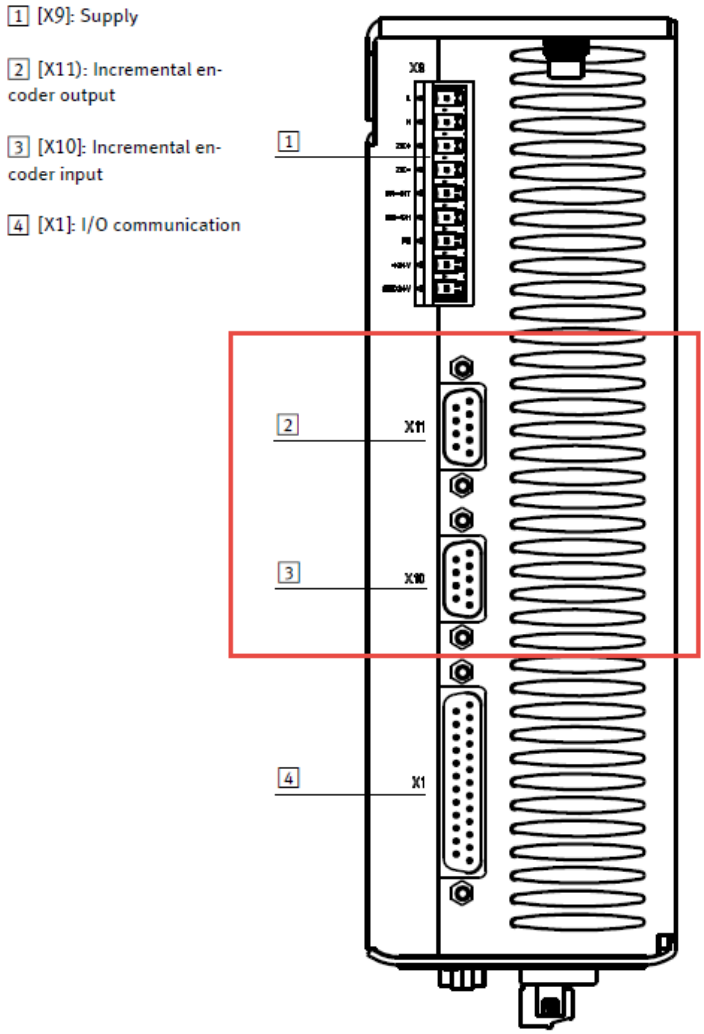


Figure 2: CMMP-AS-...-3A as example

Interface X11 is the master's incremental encoder output and X10 is the slave's incremental encoder input.

Connect the two interfaces using an appropriate cable. The pins are wired analogous to each other (pin 1 to pin 1, pin 2 to pin 2 ...).

Minimum wiring requirements: In order to implement synchronous operation at least the following pins must be connected between the two controllers: A, A#, B, B# and GND (pin 4). N and N# can also be connected. The interface parameters must be set up in accordance with the wiring configuration.

Ideally, the synchronisation cable has an outer shield which is connected to the housing. Additional inner shields around the twisted pairs, which are connected to GND (pin 4).

Pin 5 may not be connected because the controllers (as of M3/M0) supply each other with logic voltage and shutting down a controller has no effect.

A stepper-motor encoder cable of (e.g. NEBM-M12G8-E-5-S1G9 [part no. 550749]) with plug (NECC-A-S-S1G9-C2M [part no. 564264]) is recommended for use as a cable, in which the incremental encoder signals are twisted in pairs and the individual pairs are shielded.

The master generates tracking signals A and B, as well as the zero pulse of an incremental encoder.

In addition to the A/B tracking signals, slave input X10 can also process CLK/DIR – pulse/direction and the CW/CCW pulse in the form of 5 V DC.

The following is based on synchronisation of two CMMP-AS controllers with EMMS-AS motors.

Refer to the manual for further information.

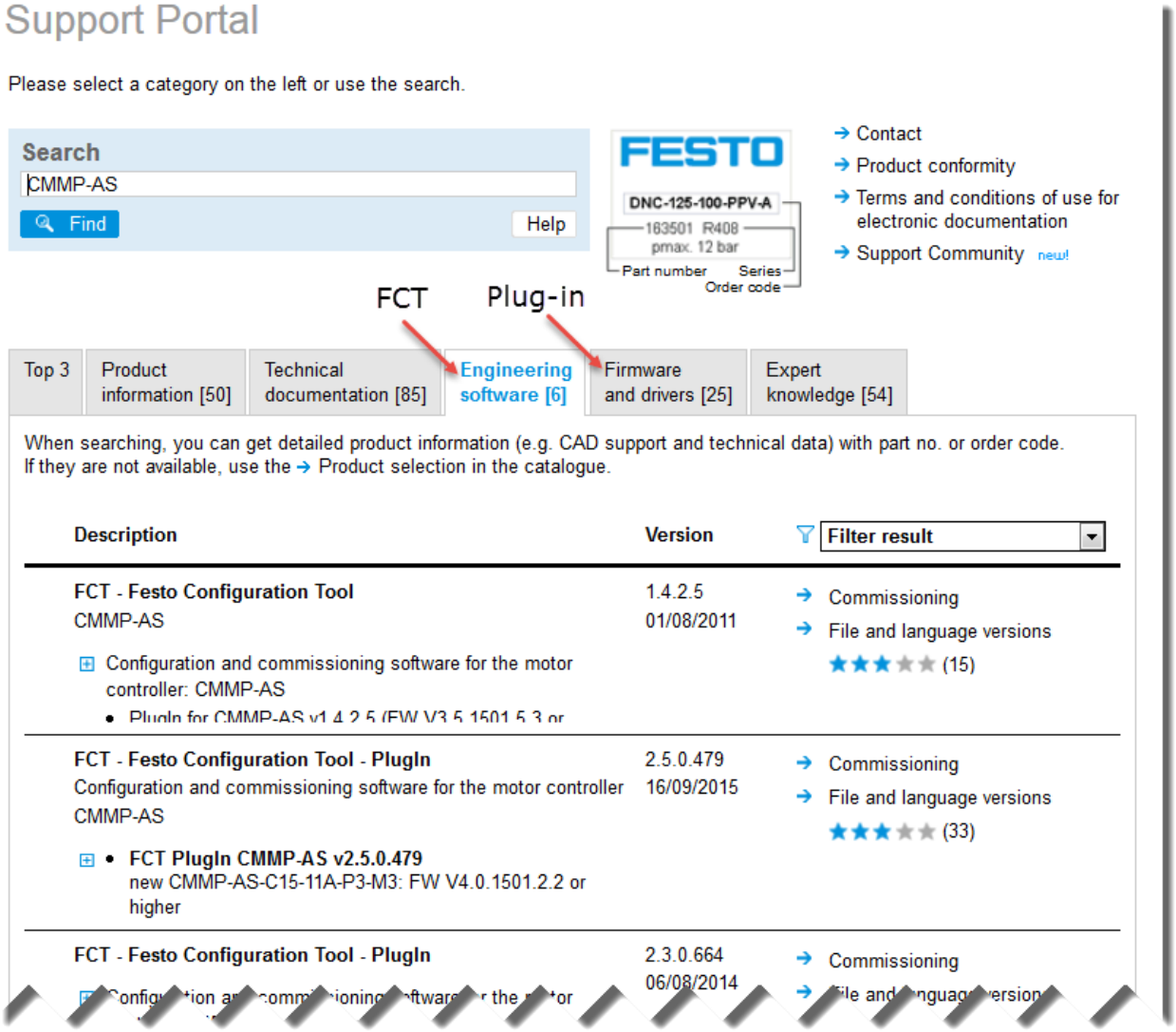
4 Configuring the parameters of the CMMP-AS-x controller

Note: Configuration of representative controllers is depicted in the following sections. All settings for which no explicit configuration is stipulated retain their default values.

These points must be adapted in accordance with your individual requirements.

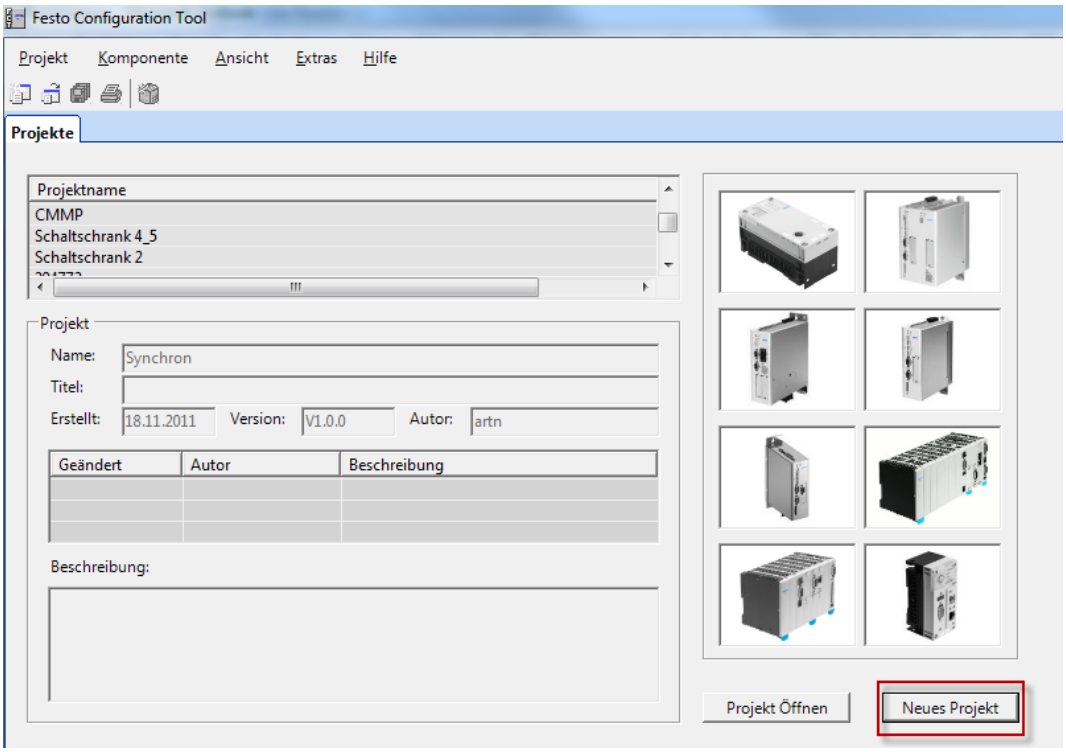
In order to configure the controller's parameters, you'll need the Festo Configuration Tool (FCT) and the corresponding plug-in for your controller. First install FCT, and then the plug-in.

You'll find the plug-in in the Support Portal after searching for the utilised controller, for example "CMMP-AS".

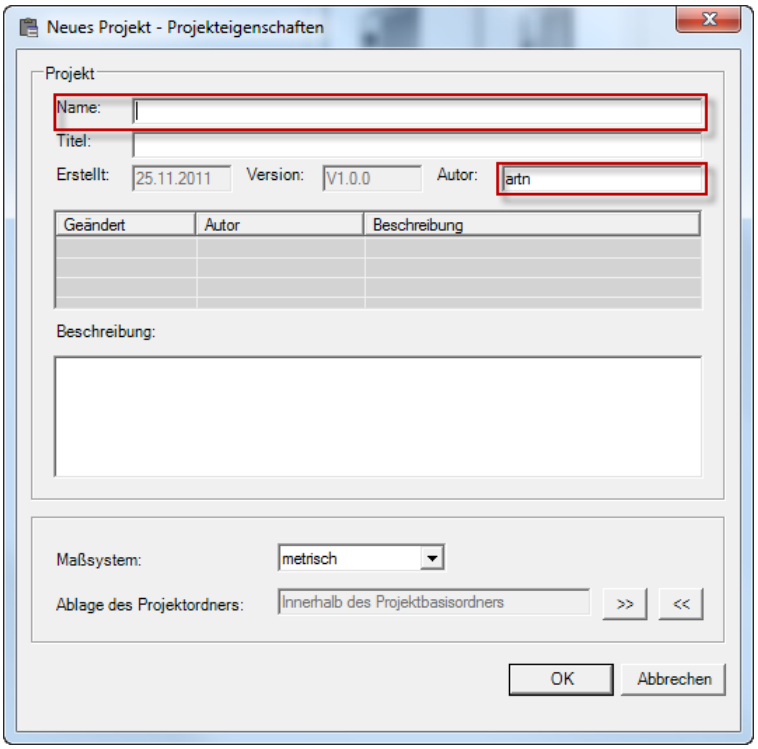


4.1 Configuring the master at the CMMP-AS

Start the Festo Configuration Tool and create a new project:

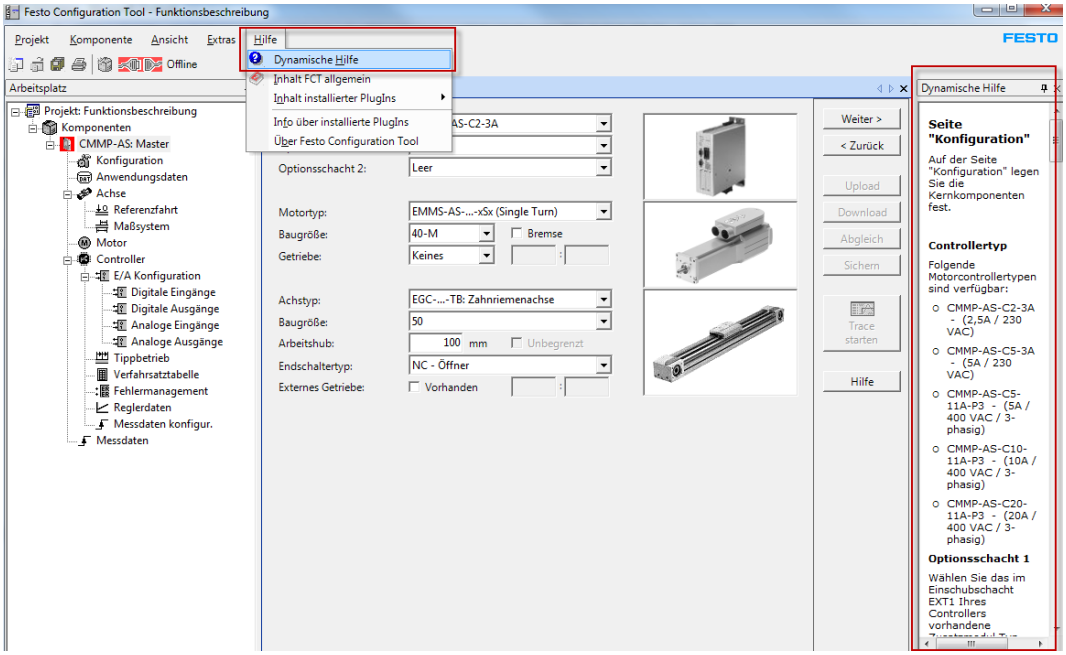


Create a new project and enter its name and the author:



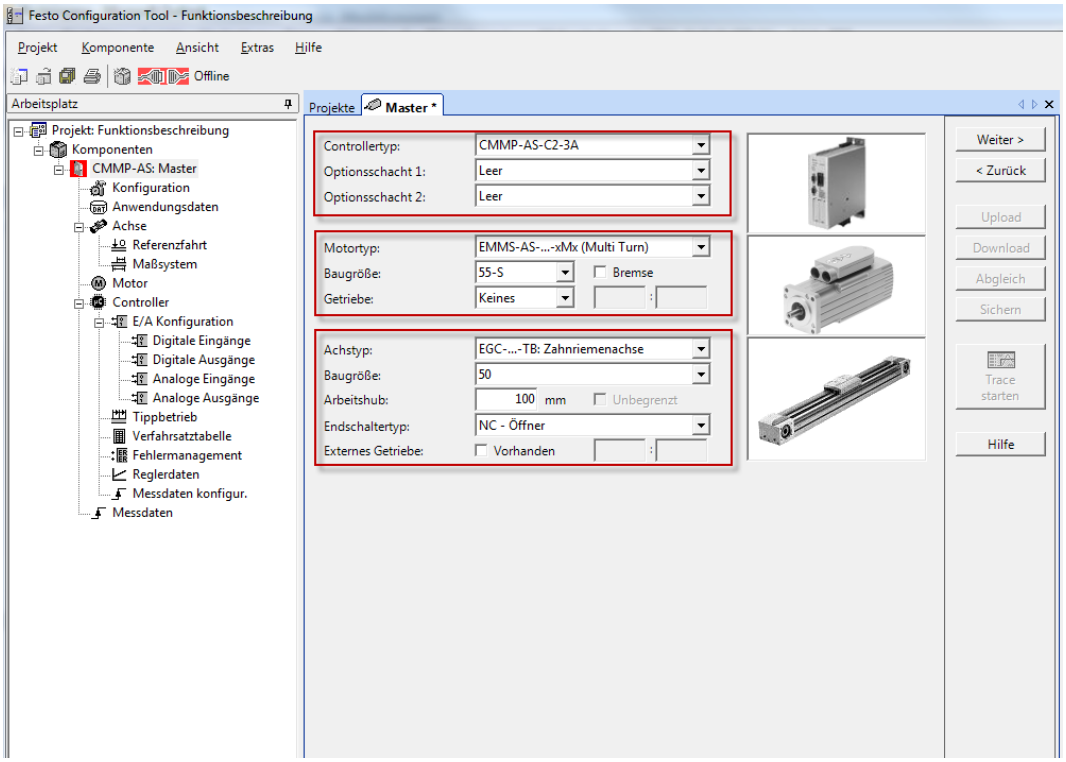
Select CMMP-AS, assign a component name to the controller and acknowledge by clicking OK:

Open the dynamic help function by selecting the Help menu and clicking Dynamic Help.
You are then always provided with notes concerning the points you're processing at the moment.



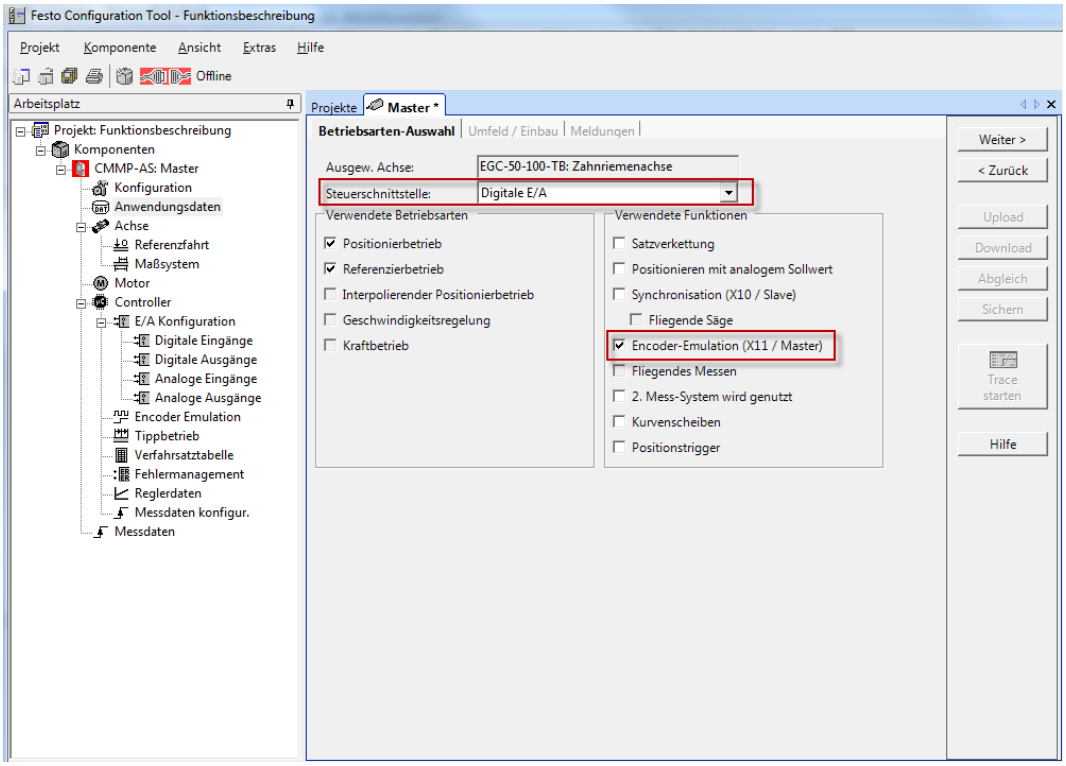
Configure your controller, motor and axis type according to your components.

For example, if you additionally use a Profibus card which is inserted into an options slot, this must be selected. Only in this way can the Profibus also be configured.

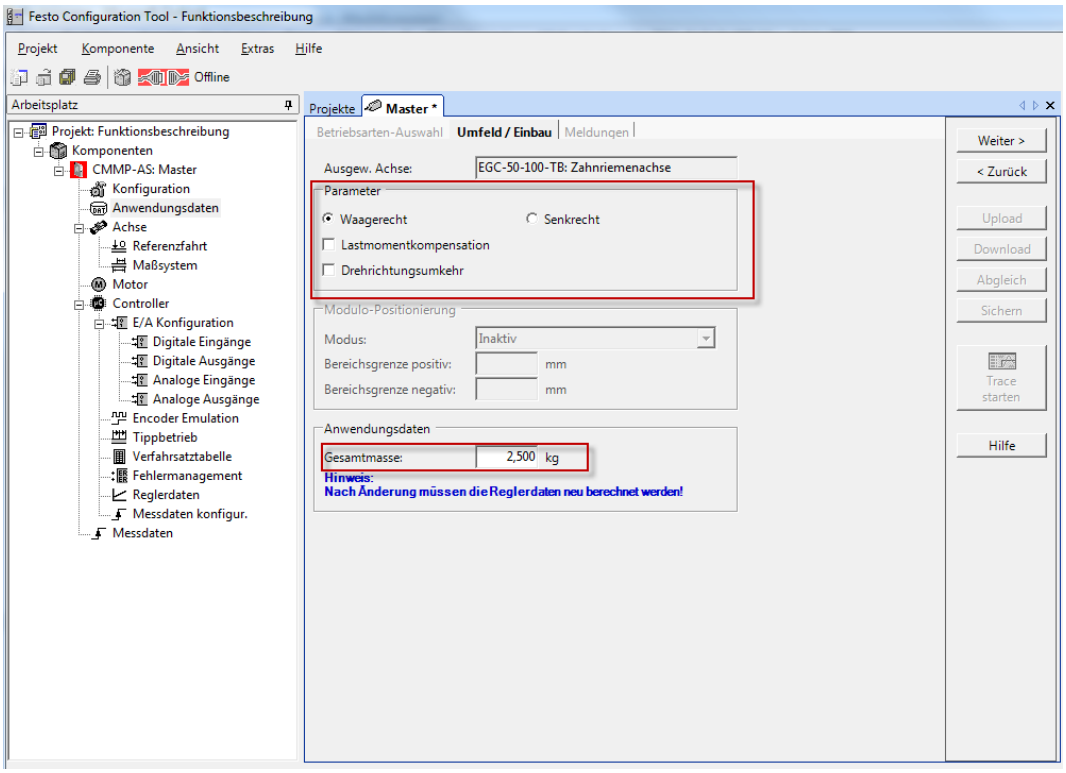


Select the desired control interface at your master controller and mark the “Encoder Emulation (X11/master)” function.

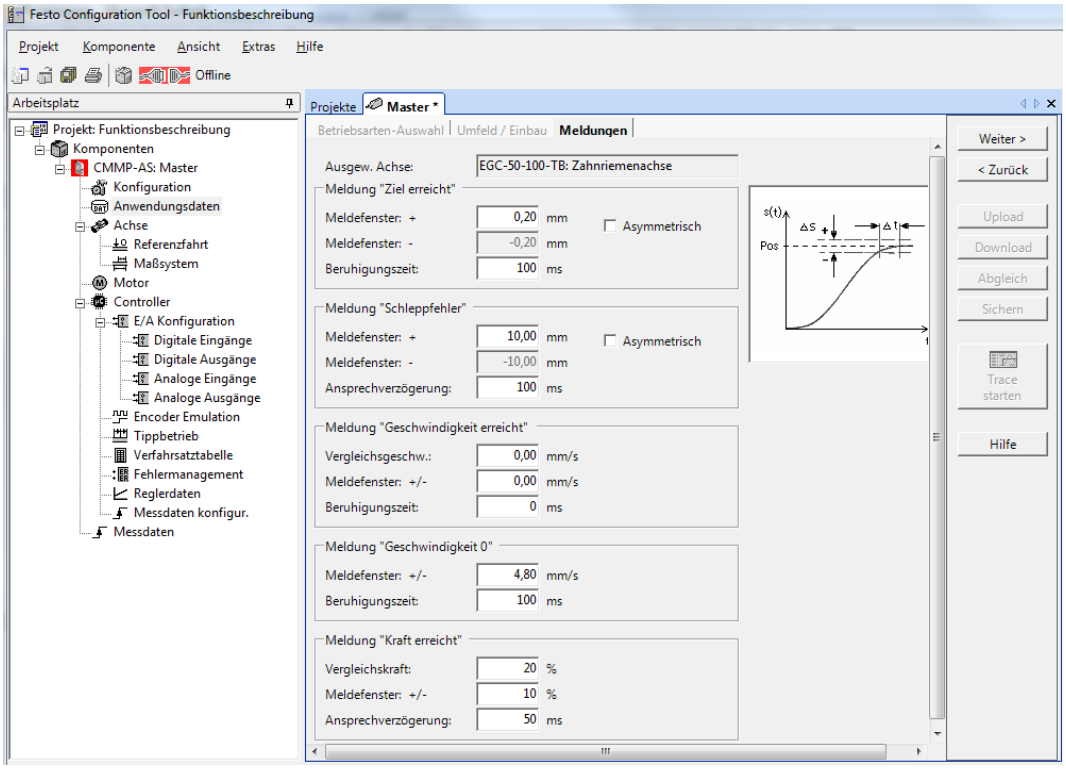
If you use Profibus or CAN bus, observe the supplements in sections 3.1.1 and 3.1.2.



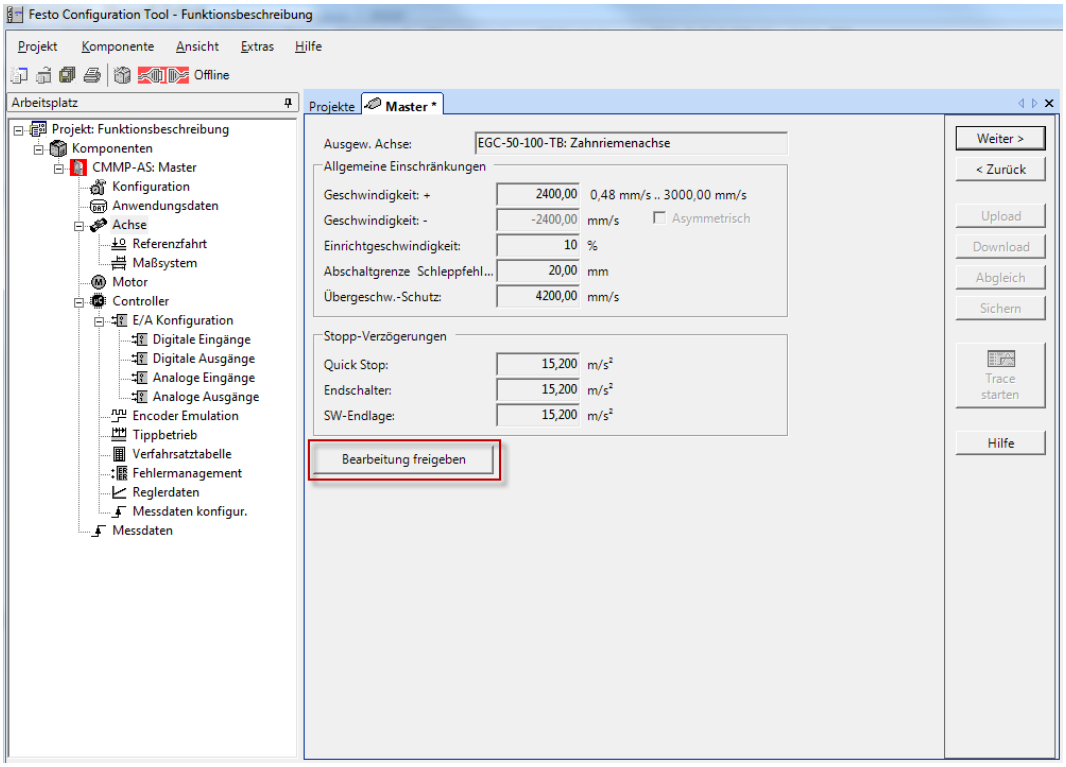
Set the parameters and enter the overall load which the master will have to set into motion.



Set the messages in accordance with your application.



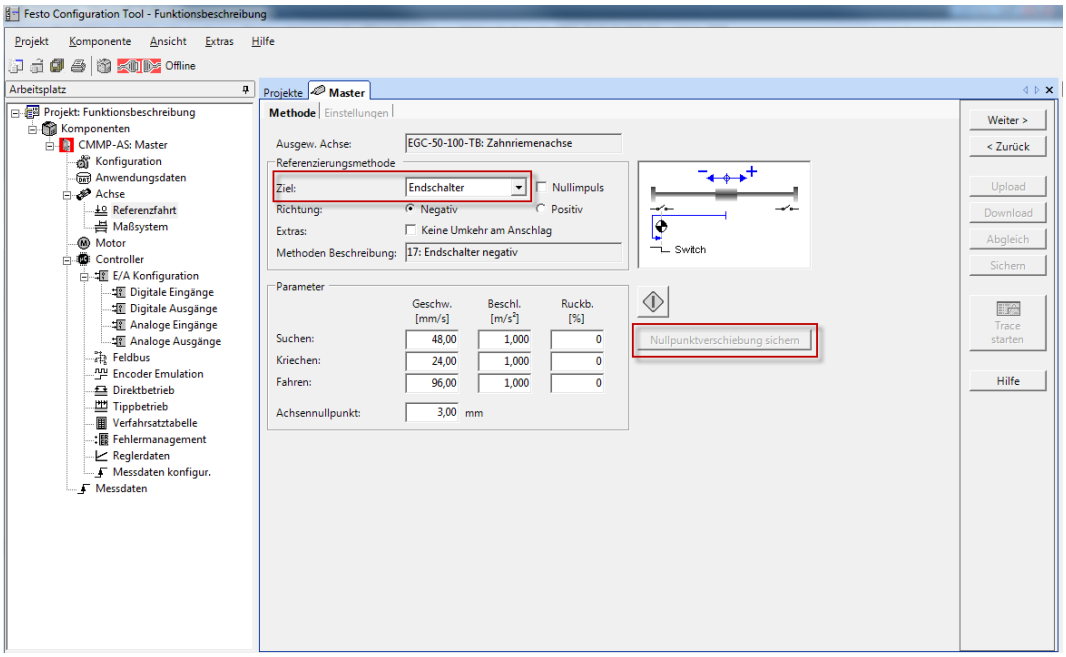
You can edit the default values for the axis after enabling editing.
If required, adapt the values in accordance with the calculated limits, for example with PositioningDrives.



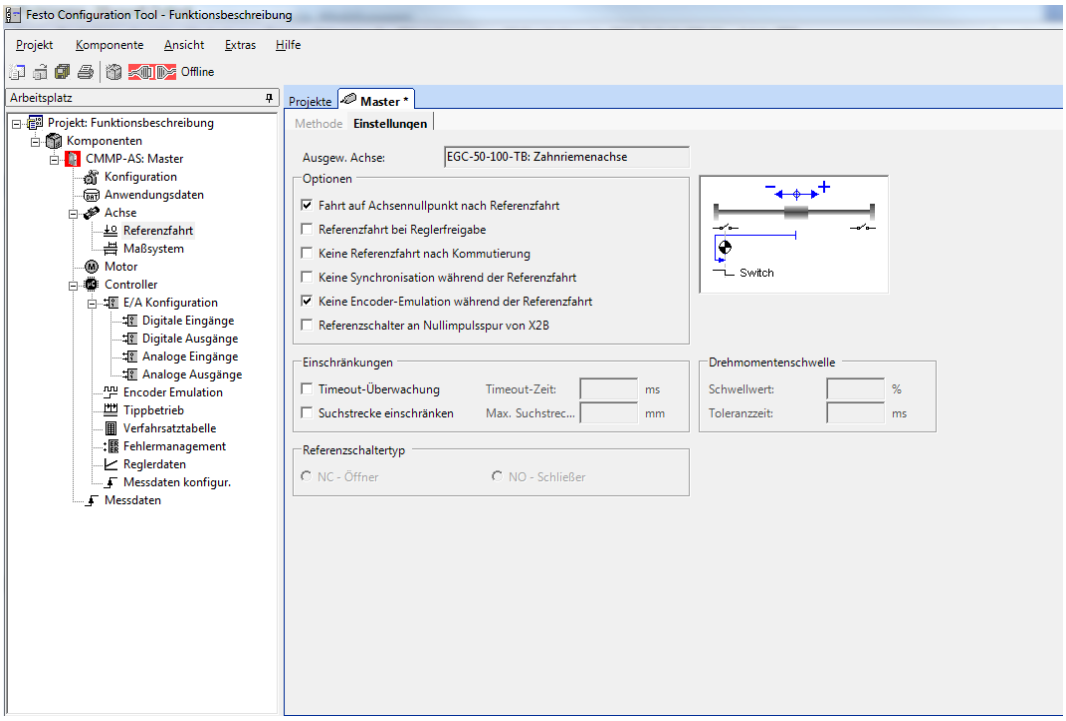
Select the homing method and set the homing parameters.

The master is referenced to a limit switch in this example. The axis then travels 3.00 mm (axis zero point: 3.00 mm).

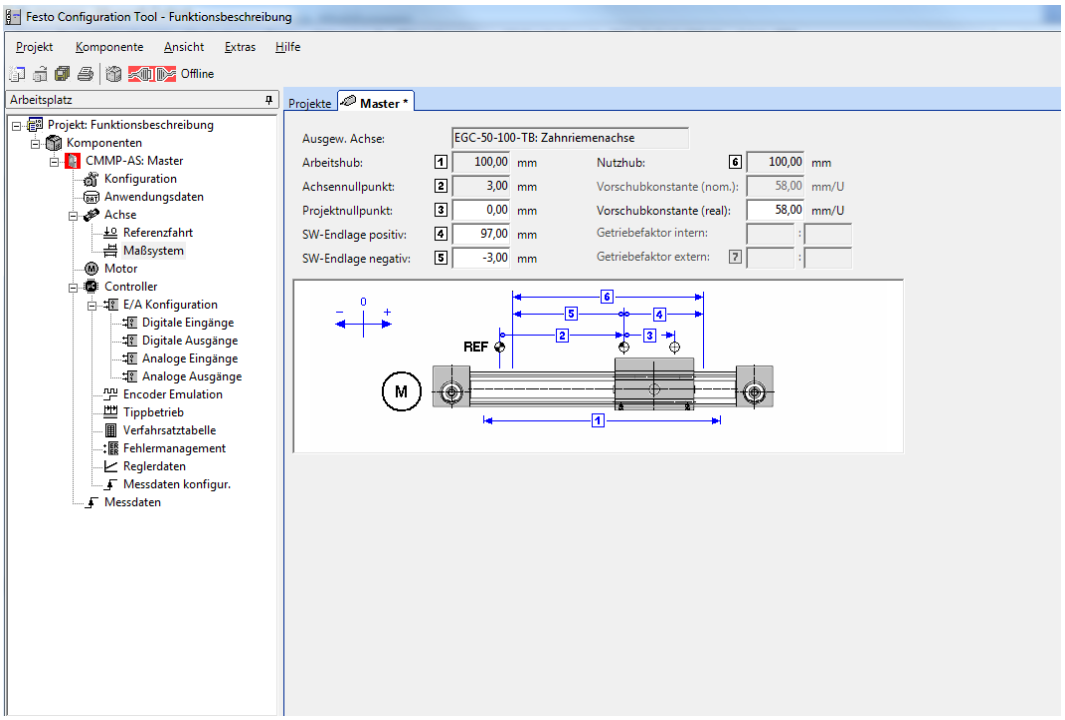
After successful homing, null shift has to be saved to the controller. The controller must not be enabled at this point in time.



Certain homing settings can be selected here. In our example it may make sense under certain circumstances to deactivate encoder emulation during homing.



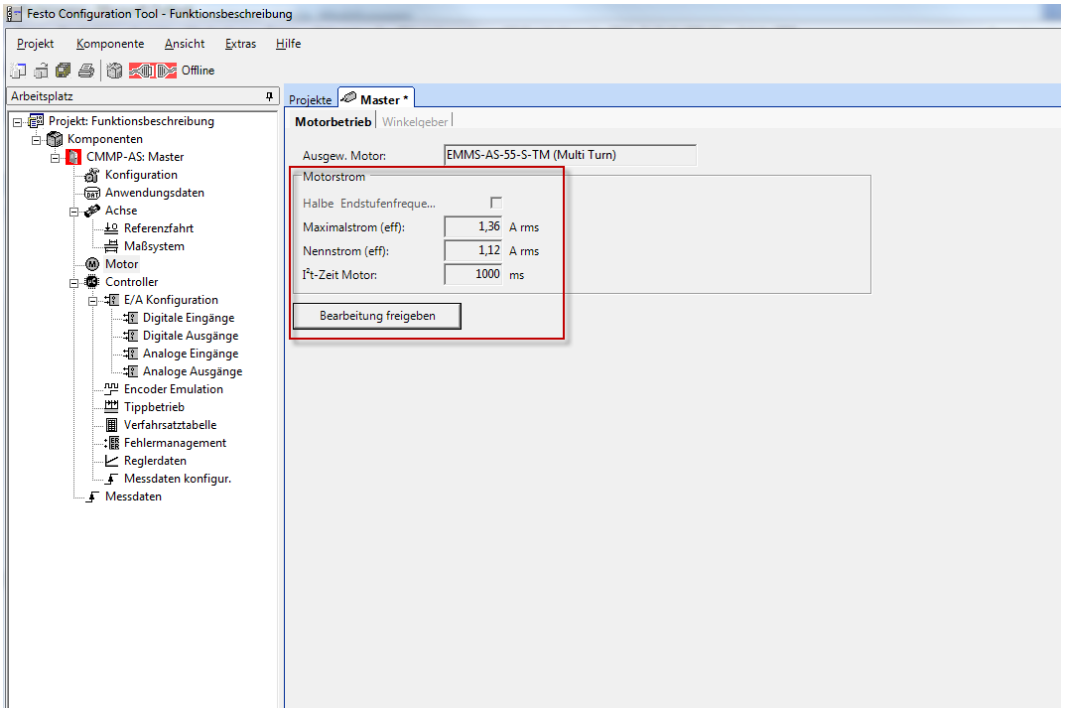
Complete the settings for the system of measures as required for your application.
The default values are used for the application depicted here.
The drive advances 3 mm after detection by the limit switch.



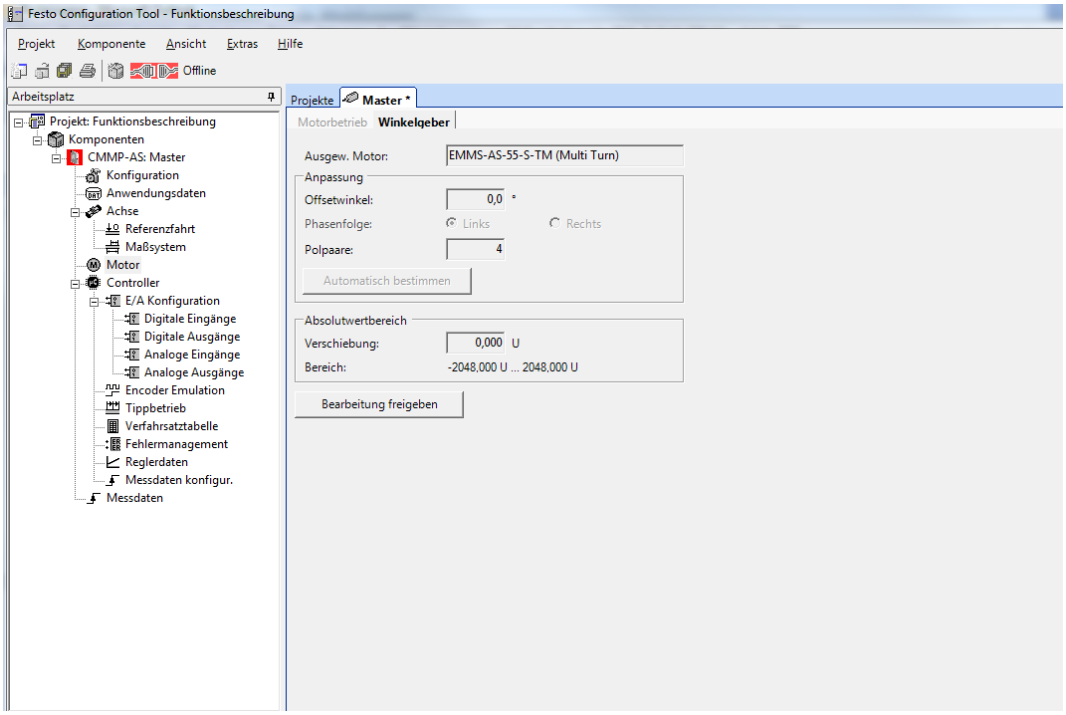
This window provides you with the opportunity of limiting motor current. This may be necessary in order to prevent damage to the axis due to excessive applied torque.

The values are adapted automatically in the case of motor-axis combinations from Festo.

The values can be changed after clicking “Enable Editing”.

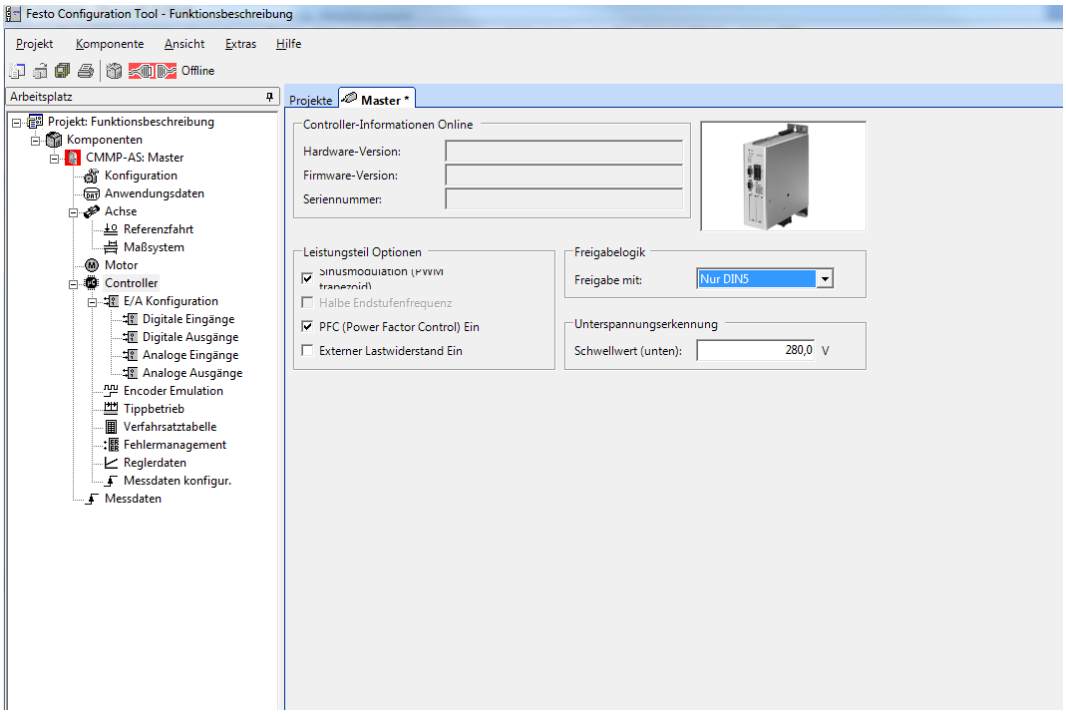


Changes to the angle encoder can be made in this tab. As a rule, no settings need to be entered here. The offset angle is determined automatically when switched on.



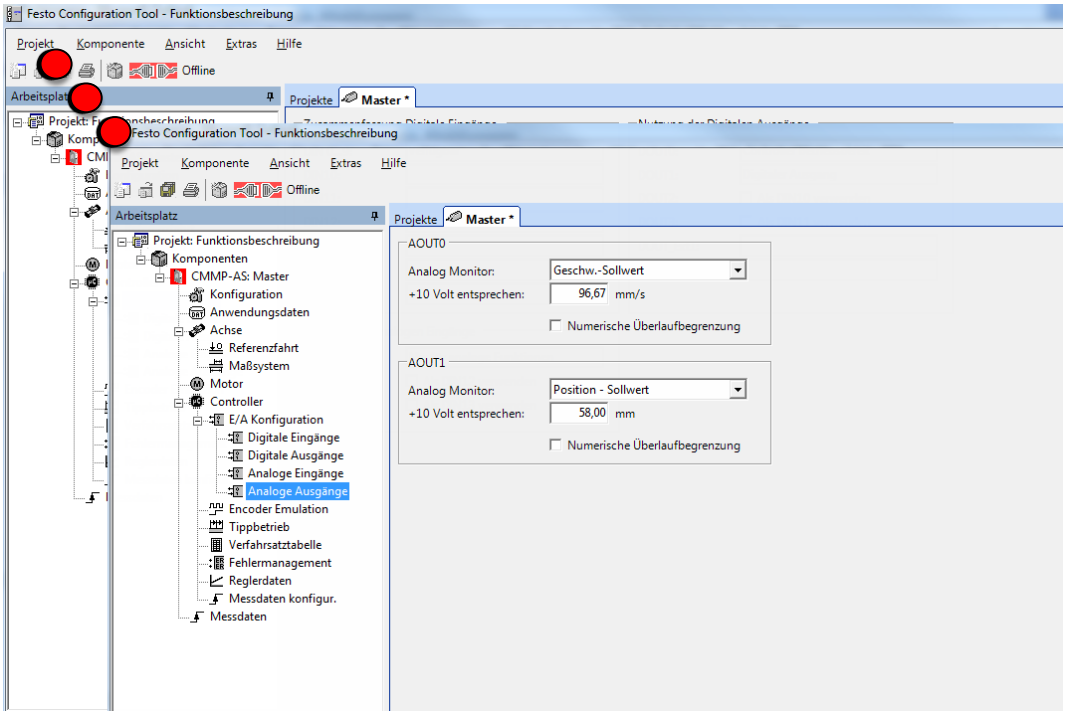
The revision level of the currently installed firmware can be viewed and basic settings can be entered for the controller in the “Controller” menu.

Settings are entered here including sinusoidal modulation, PFC, enabling with DIN5 and a threshold value of 280 V for undervoltage detection.



The I/Os can be set on the following pages. Adapt them to your circuit.

Acknowledge each page by clicking “Next” until you arrive at the encoder emulation settings.



Configure the encoder emulation parameters in accordance with your application.

The line count refers to one revolution of the encoder and can lie within a range of 1 to 8192, or can be precisely 16,384.

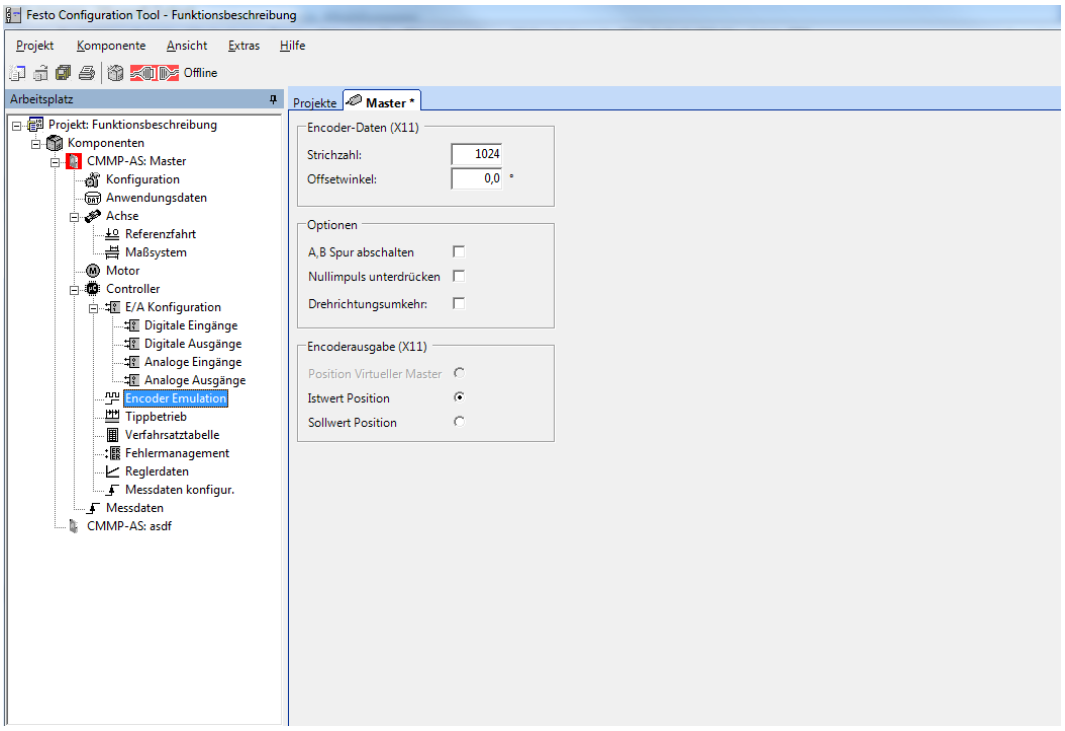
A correction factor can be set by entering an offset angle.

The line count is set to 1024 in our example.

None of the options are selected, so that all data will be transferred initially.

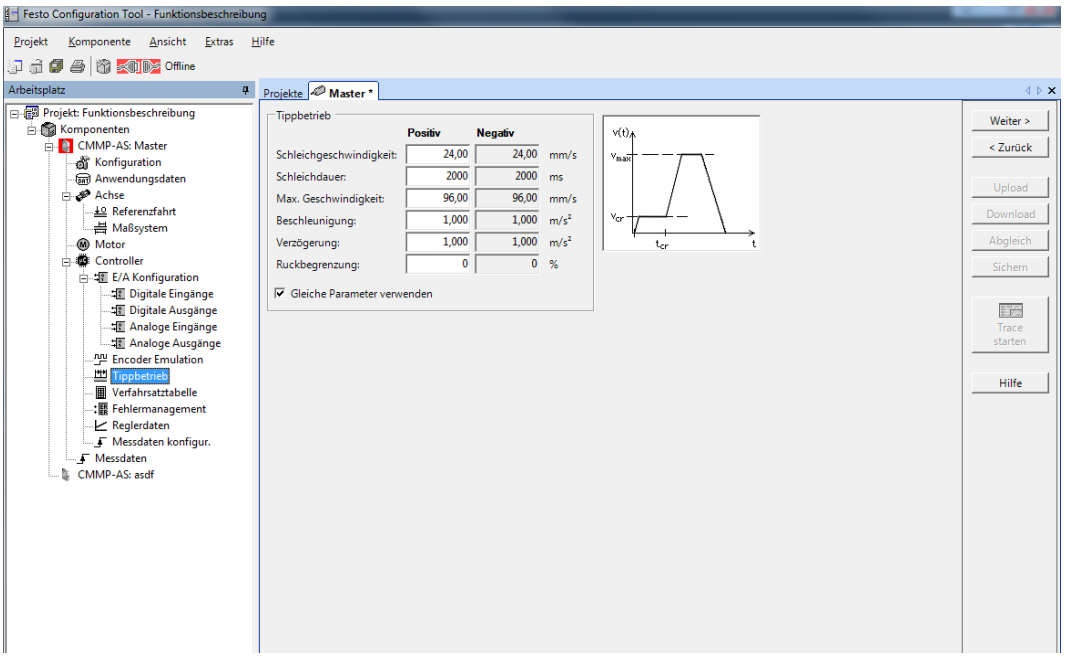
The master's actual position is read out. This is the desired function in most synchronisation applications.

Reading out the setpoint has the advantage of providing the slave with smoother setpoints, but it only follows the master if controller enabling is set there.

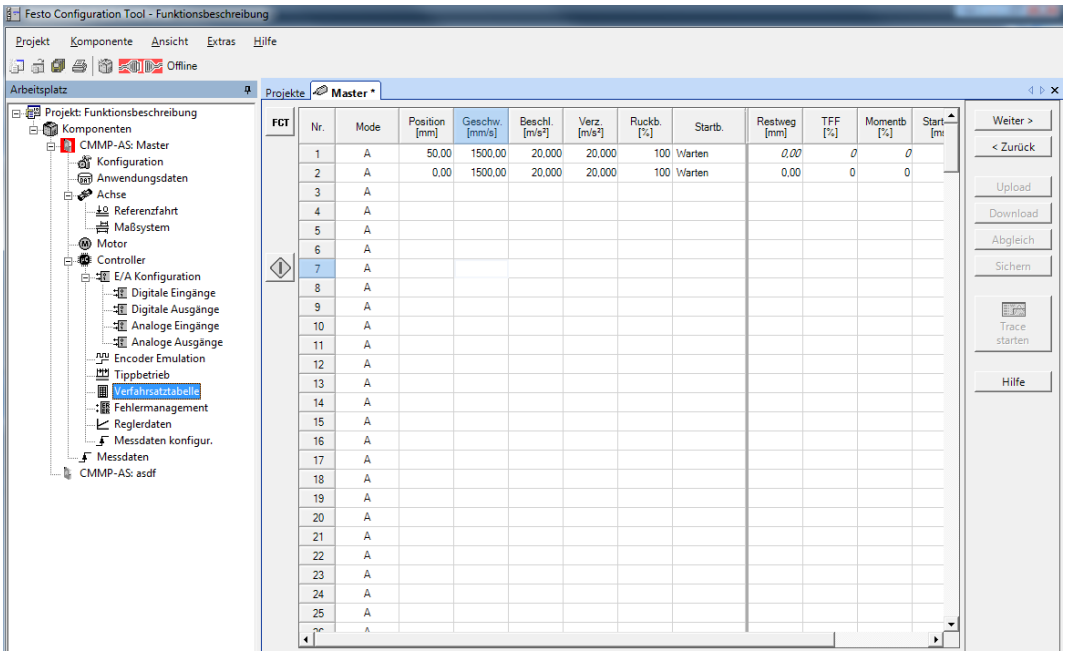


Settings for jogging operation can be selected in this window. Jogging operation is important for the setup mode, for example.

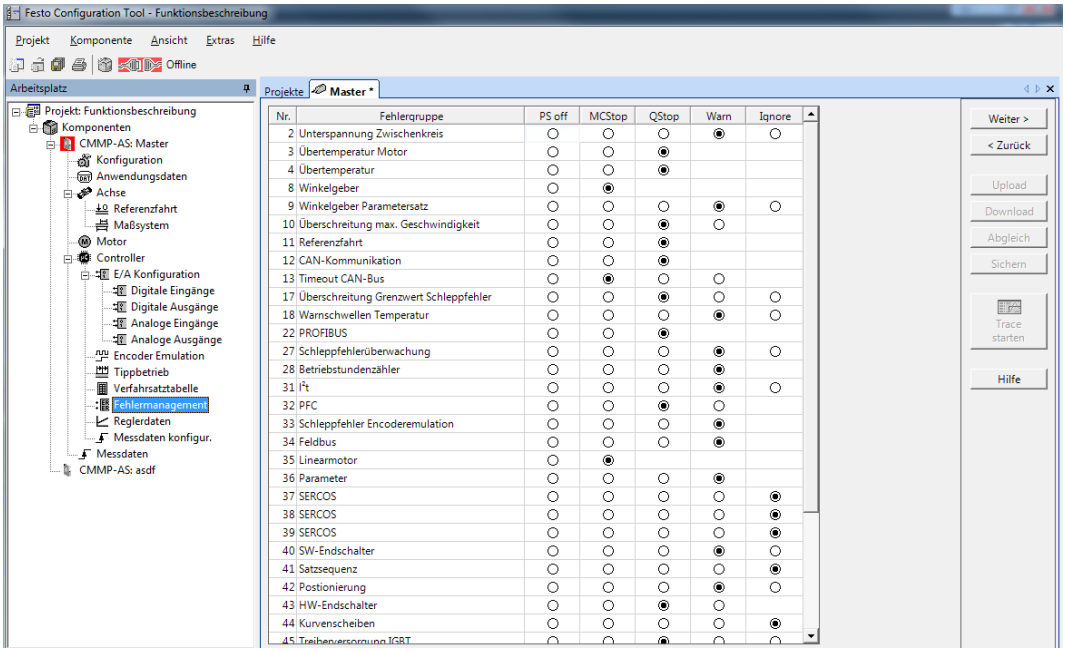
Select the desired values.



Different positioning records can be created for the master with the help of the positioning records table, which can then be retrieved via the I/O interface or the bus.



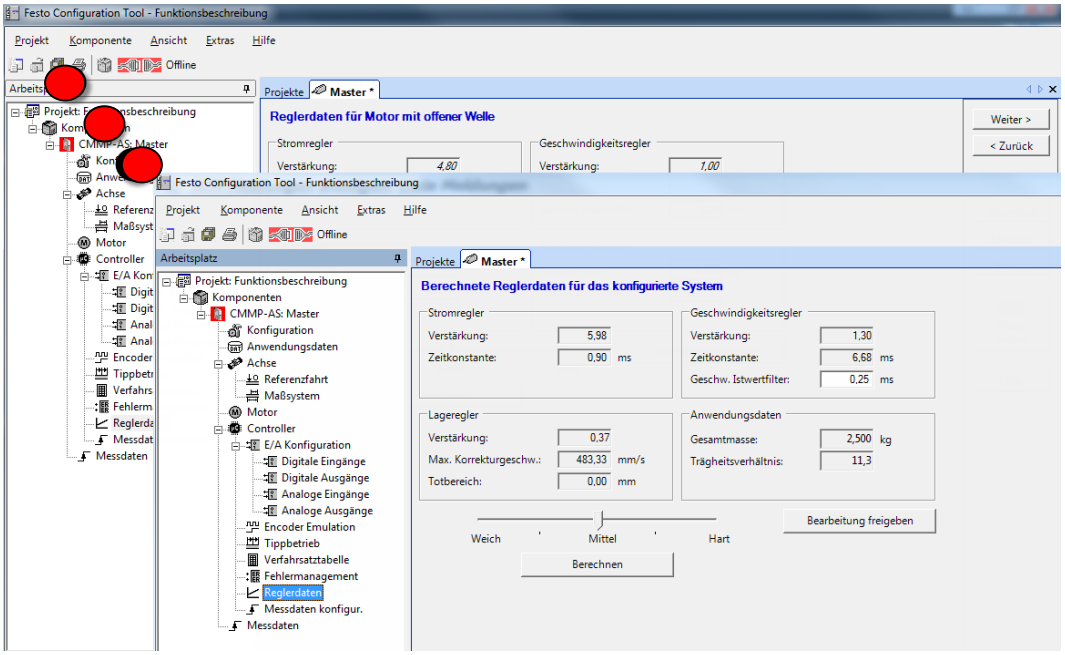
You can specify how the controller will react to various errors in the errors management window. This depends on your application and can be accordingly adapted.



The controller data are calculated by FCT depending on your previous entries. You can select controller data within a range from soft to hard.

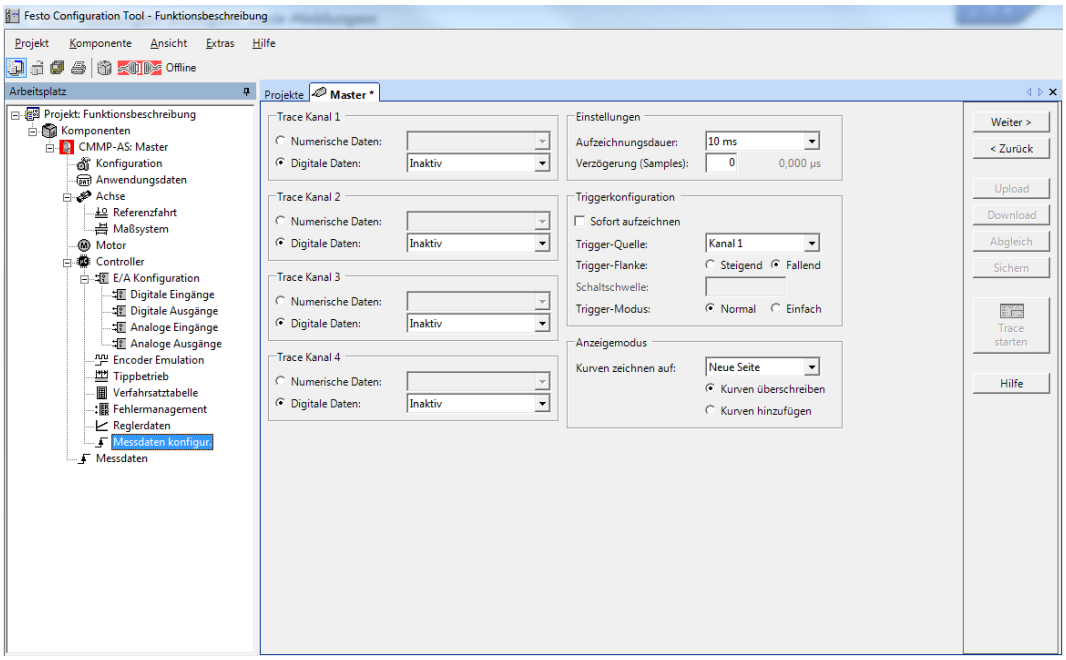
Before the controller data are calculated, the data for a motor with open shaft are parameterised.

The controller data are recalculated after clicking the “Calculate” button.

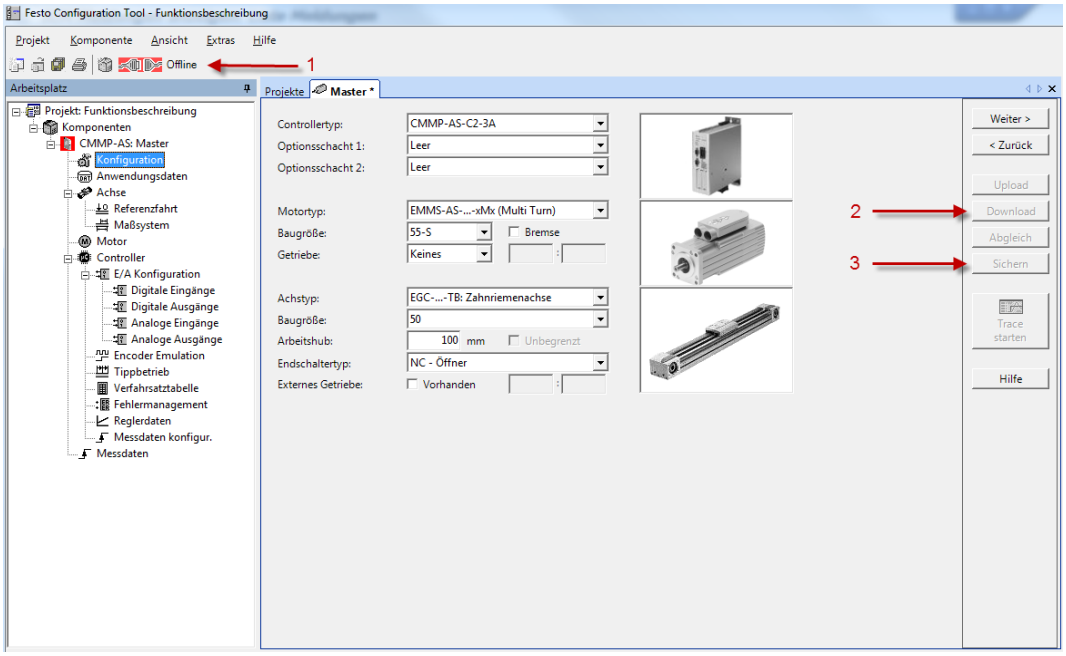


The controller provides you with the option of recording measurement data. This function must first be configured. You can optimise your controller parameters with the help of the measurement data.

The data can be viewed in the “Measurement Data” window.



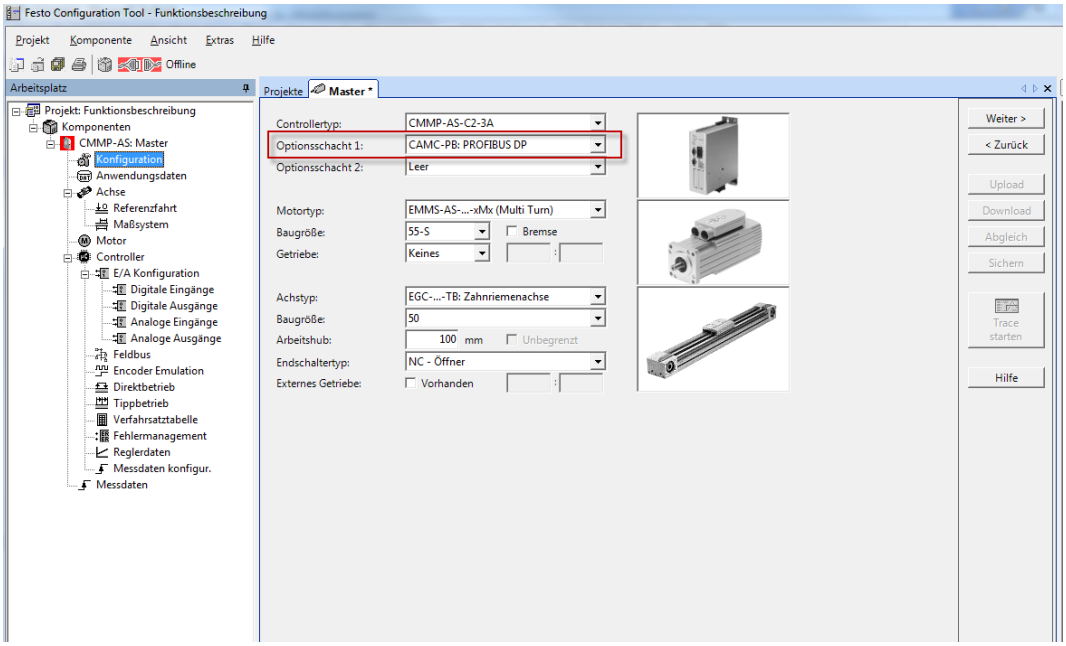
After configuration has been completed, the data have to be transferred to your controller.
Go online with FCT to this end and confirm by clicking the “Download” button.
Then click the “Save” button so that the data will be retained, even after restarting the controller.



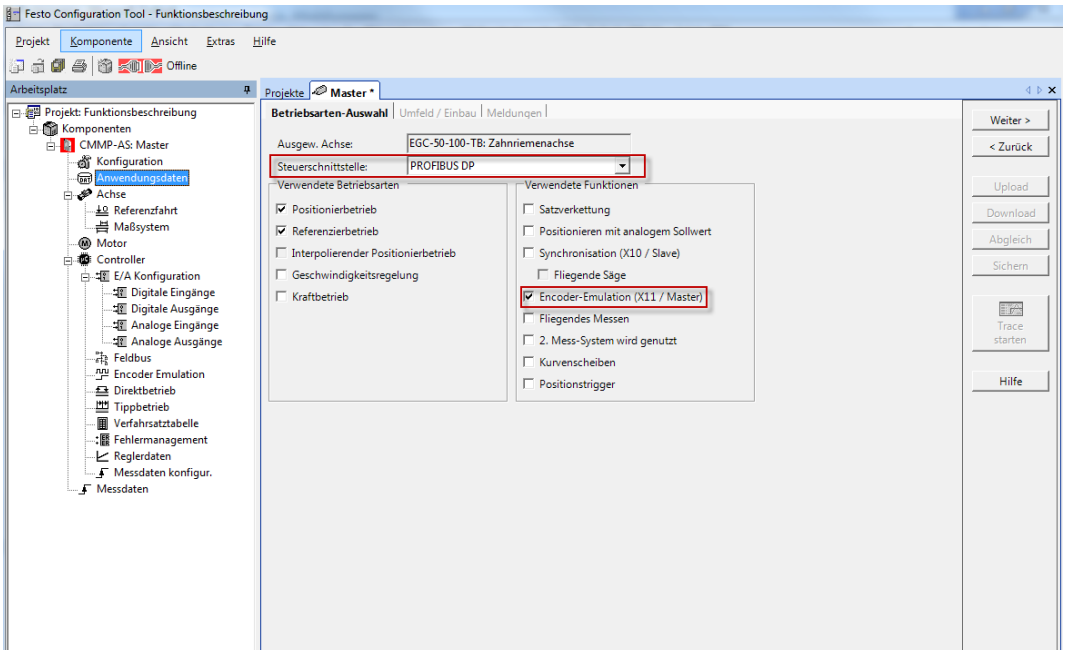
4.1.1 Supplement – Profibus control interface

In order to be able to use the Profibus at the controller, you'll need an additional plug-in module for insertion into slot 1 or 2.

This must be correspondingly configured in FCT.

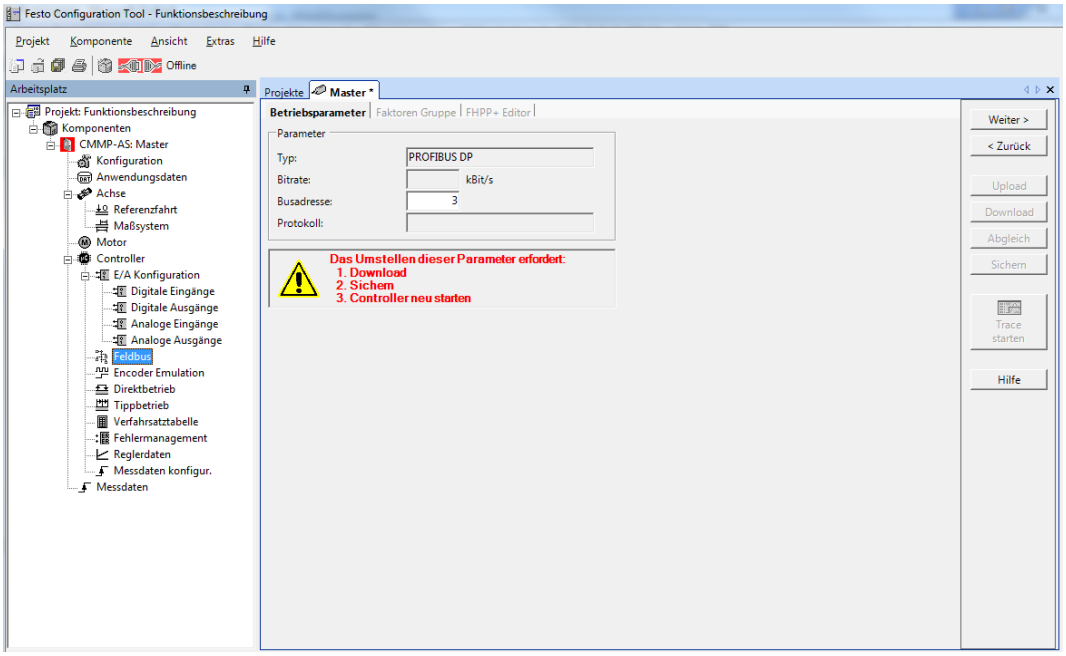


The control interface must be changed to Profibus DP in the application data menu and "Encoder Emulation (X11/master)" has to be activated.

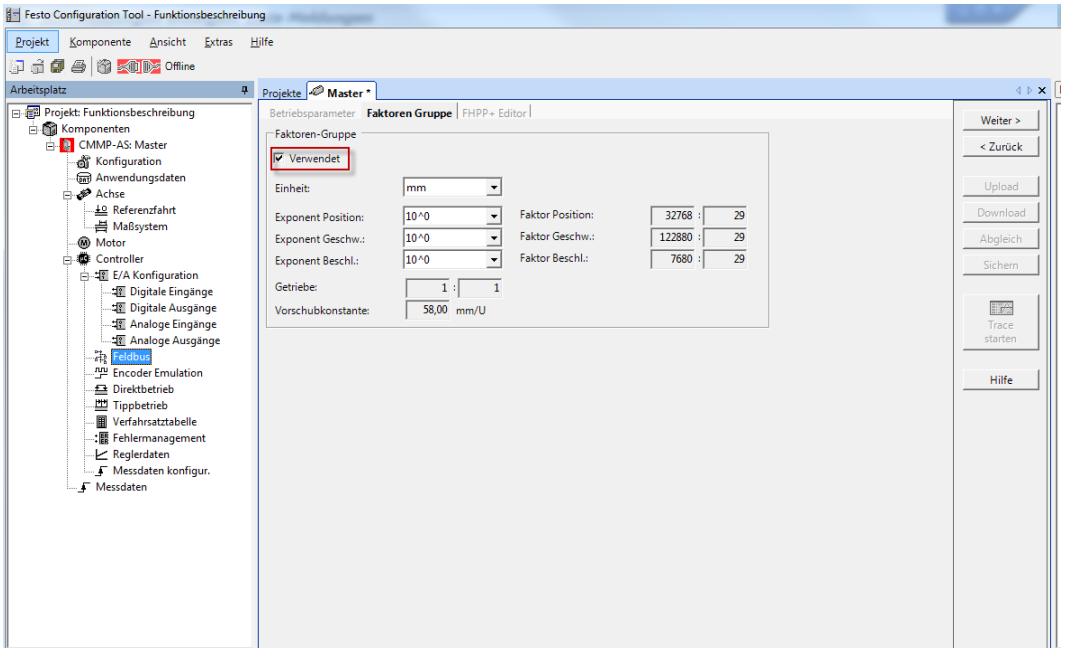


A new menu item then appears in the tree, namely “Fieldbus”.

The corresponding bus settings must be entered here, so that the controller can communicate with the control system.

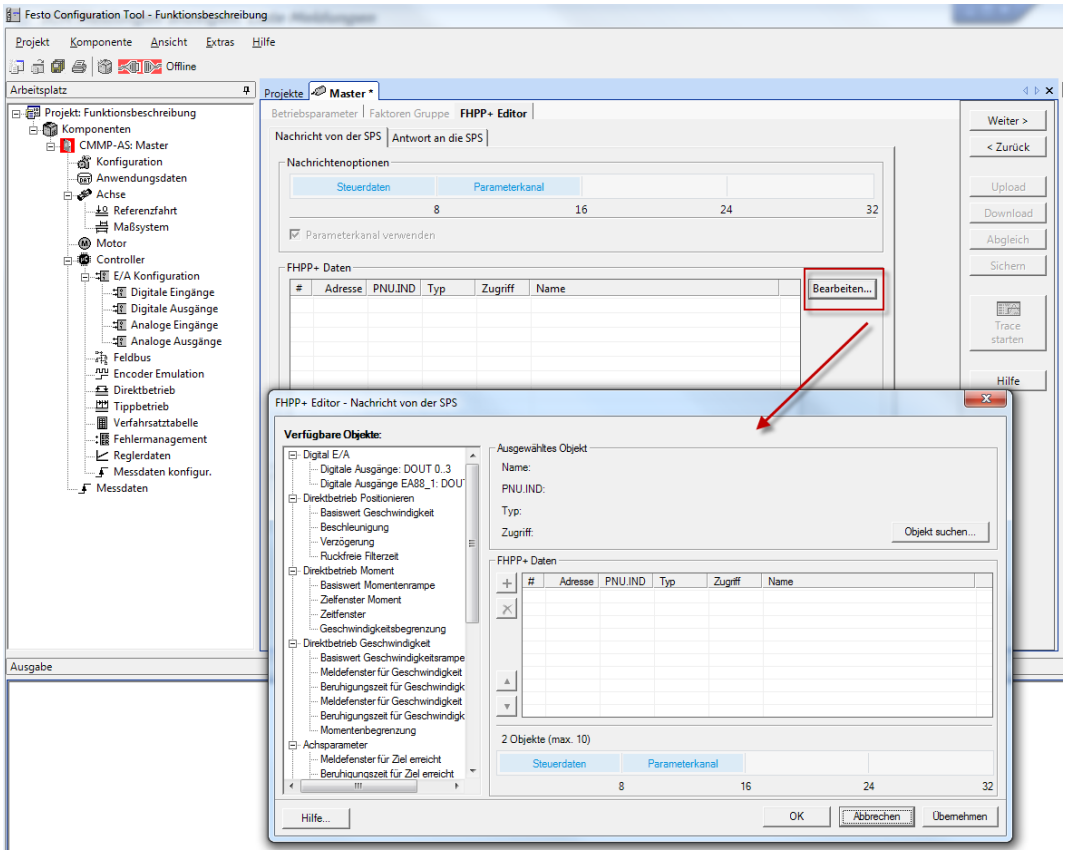


Use of the factors group can be activated in the “Factors Group” tab. This provides you with the option of, for example, displaying position at the module by means of a user-defined unit of measure instead of as an incremental value.



If you use the “Festo FHPP” protocol, cyclic values can be configured in the FHPP+ Editor which are also transmitted to or from the PLC for each cycle.

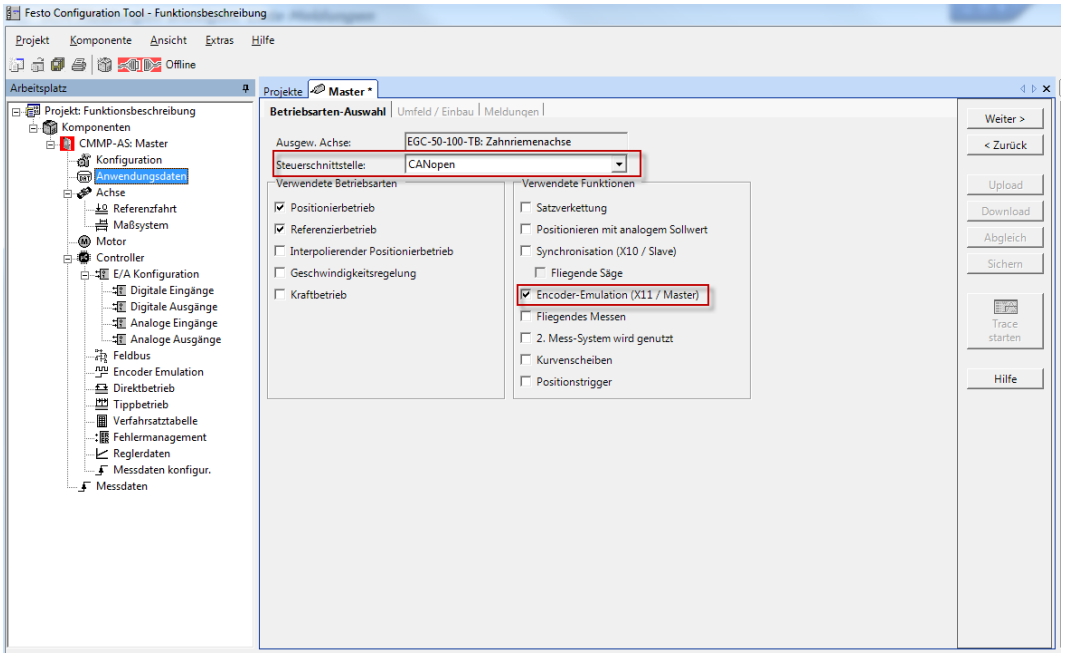
After clicking the “Edit ...” button, a new window appears at which the values can be selected.



4.1.2 Supplement – CAN bus control interface

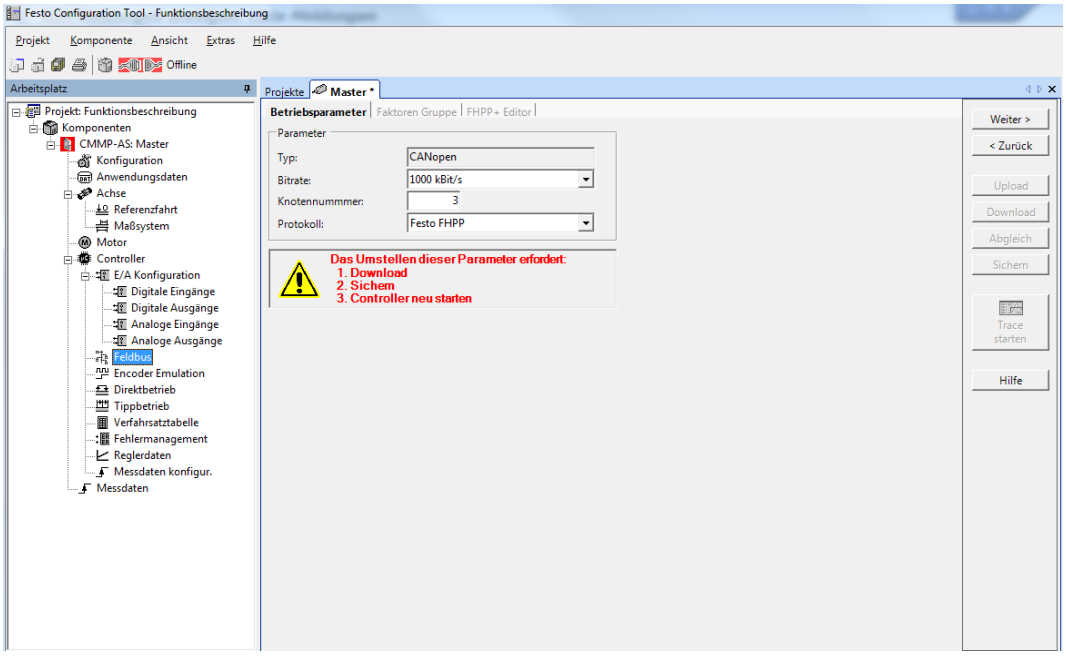
If you use the CAN bus control interface, change the control interface to CANopen under operating mode selection in the application data menu.

Then select encoder emulation.

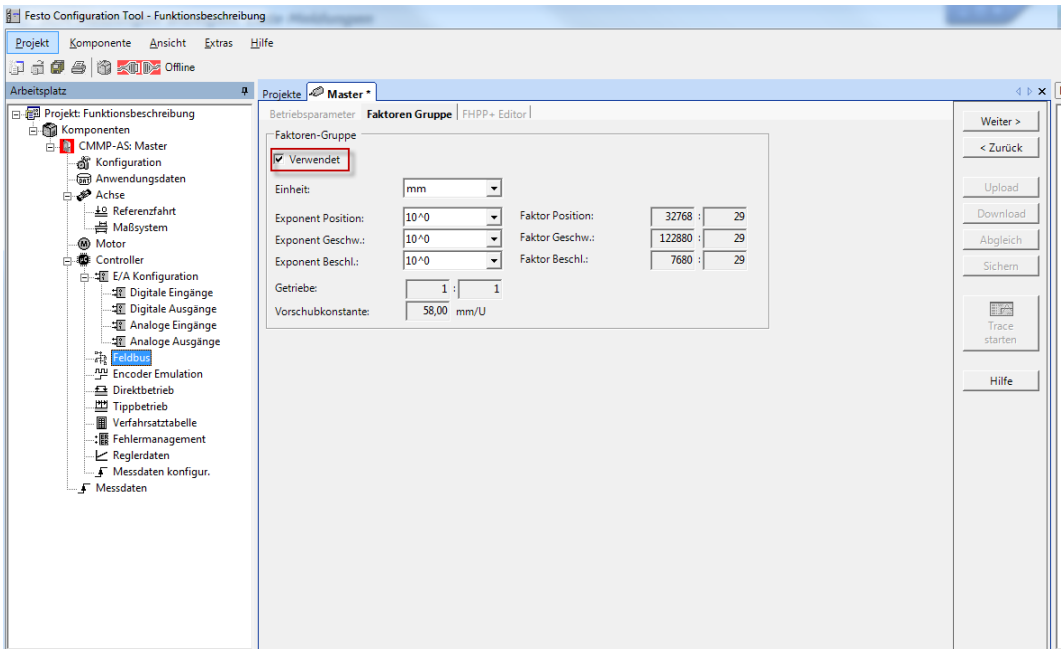


A new menu item then appears in the tree, namely “Fieldbus”.

The corresponding bus settings must be entered here, so that the controller can communicate with the control system.

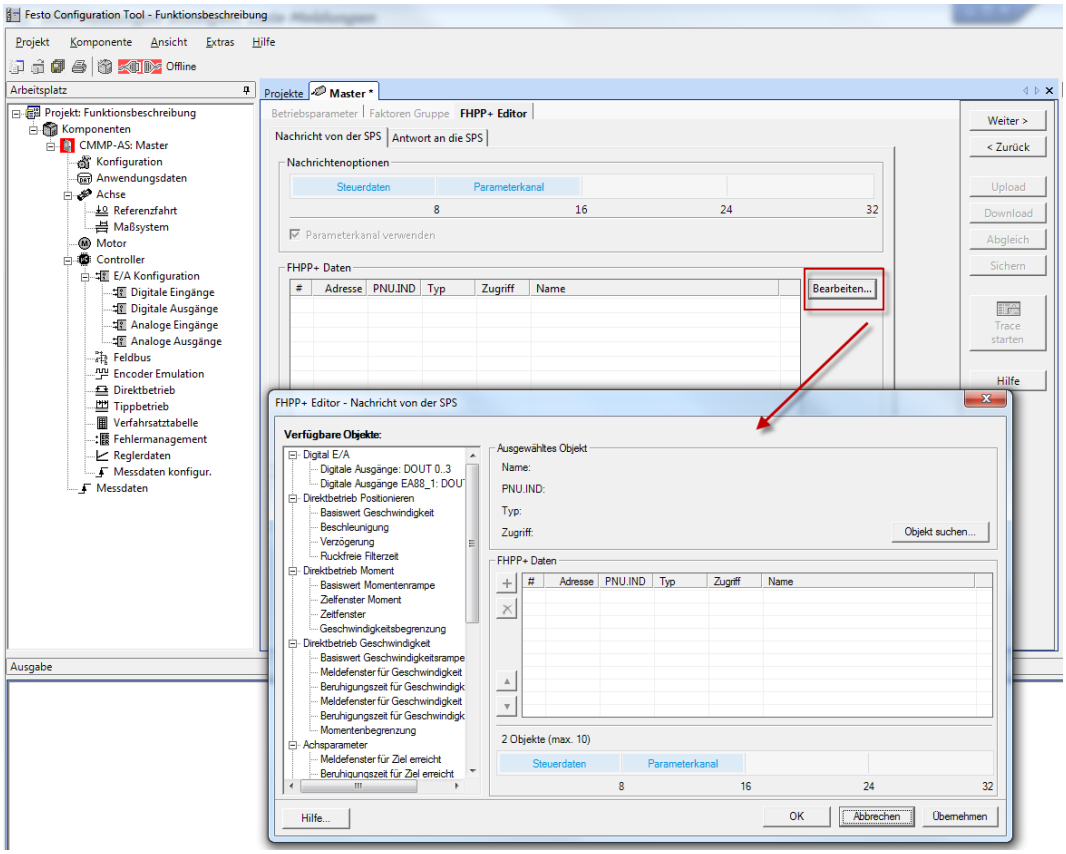


Use of the factors group can be activated in the “Factors Group” tab. This provides you with the option of, for example, displaying position at the module by means of a user-defined unit of measure instead of as an incremental value.



If you use the “Festo FHPP” protocol, cyclic values can be configured in the FHPP+ Editor which are also transmitted to or from the PLC for each cycle.

After clicking the “Edit ...” button, a new window appears at which the values can be selected.



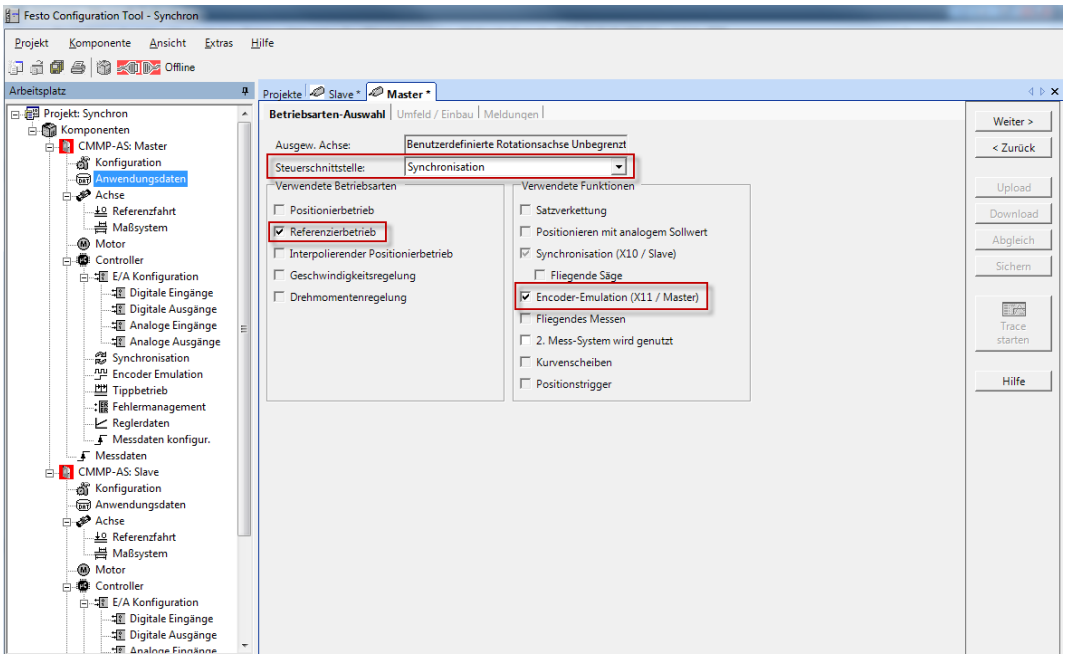
4.1.3 Supplement – synchronisation control interface

The master can be operated via the “Synchronisation” control interface. “Encoder Emulation (X11/master)” is selected to this end, as well as the “homing” mode if required.

For this mode it must be assured that the controller is always configured as a slave and that the “master” function has to be explicitly selected in addition.

The position data are thus provided by another master, from which they are forwarded to a further slave.

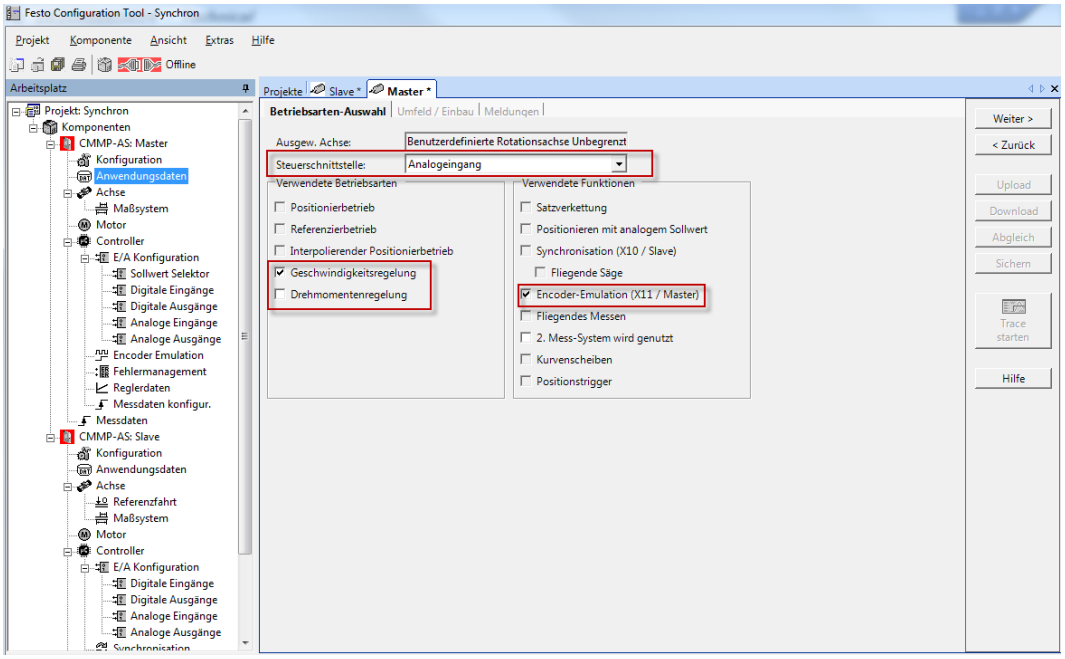
A controller of this sort cannot be operated, for example, by means of positioning records.



All other required settings are included as of section 4.1.

4.1.4 Supplement – analogue input as control interface

If the analogue input is selected as the control interface, the axis can also be configured as a master.
In this case, the master can be operated with either speed or torque control.
It must be assured that the slave cannot execute torque control, because the master only transmits the increments which are converted to position, speed and acceleration.



All other required settings are included as of section 4.1.

4.2 Slave configuration – synchronisation control interface

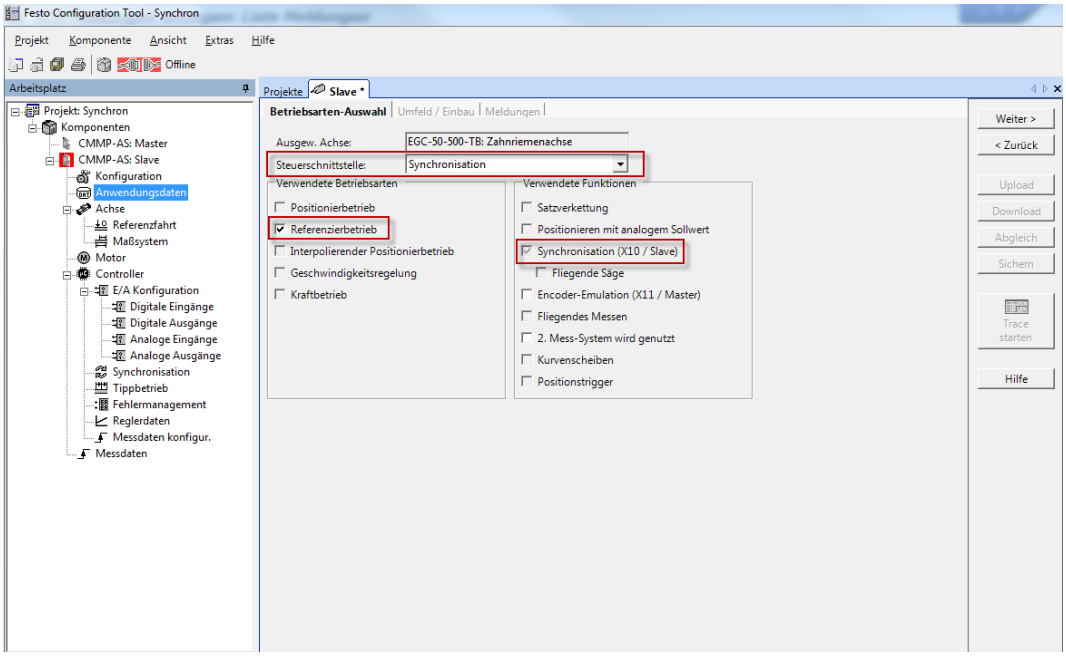
Basic configuration is described in section 4.1. The differences are shown in this section.

With this type of synchronisation is must be assured that the slave reacts to every change at the master. The possibilities for controlling this type of synchronisation are described in section 4.1.

In order to activate this operating mode, “Synchronisation” must be chosen as the control interface under operating mode selection in the application data menu.

“Synchronisation (X10/slave)” is activated automatically. Deselection is not possible.

If homing is required, the “homing” mode must be activated.



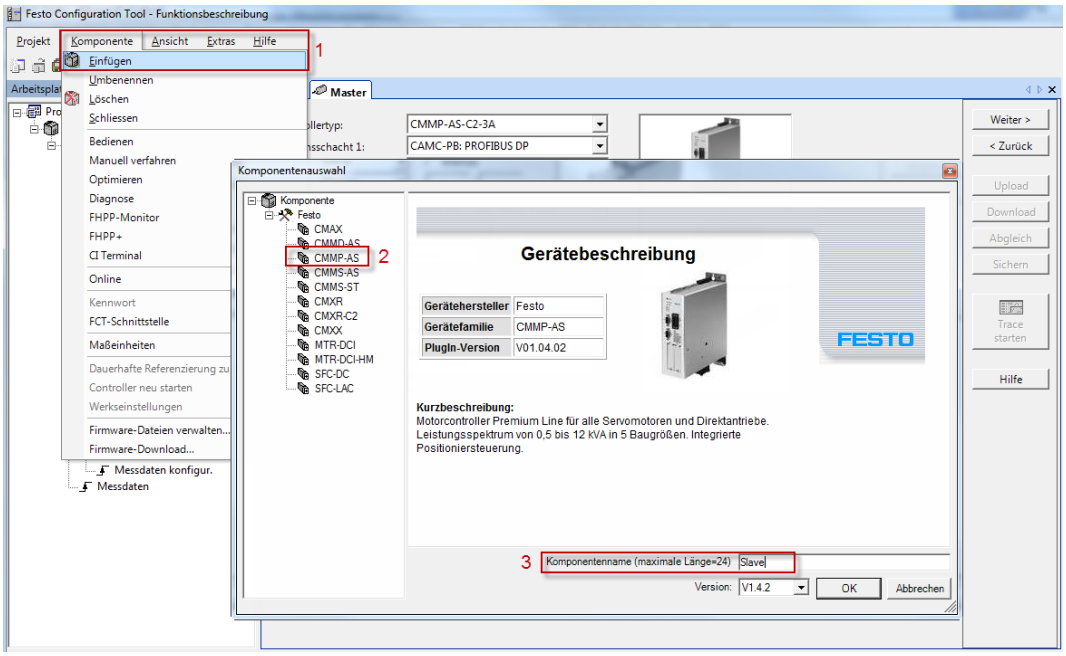
Execute homing if necessary.

As soon as the slave is active, it follows the master’s increments for all motion.

4.3 Slave configuration – I/O control interface

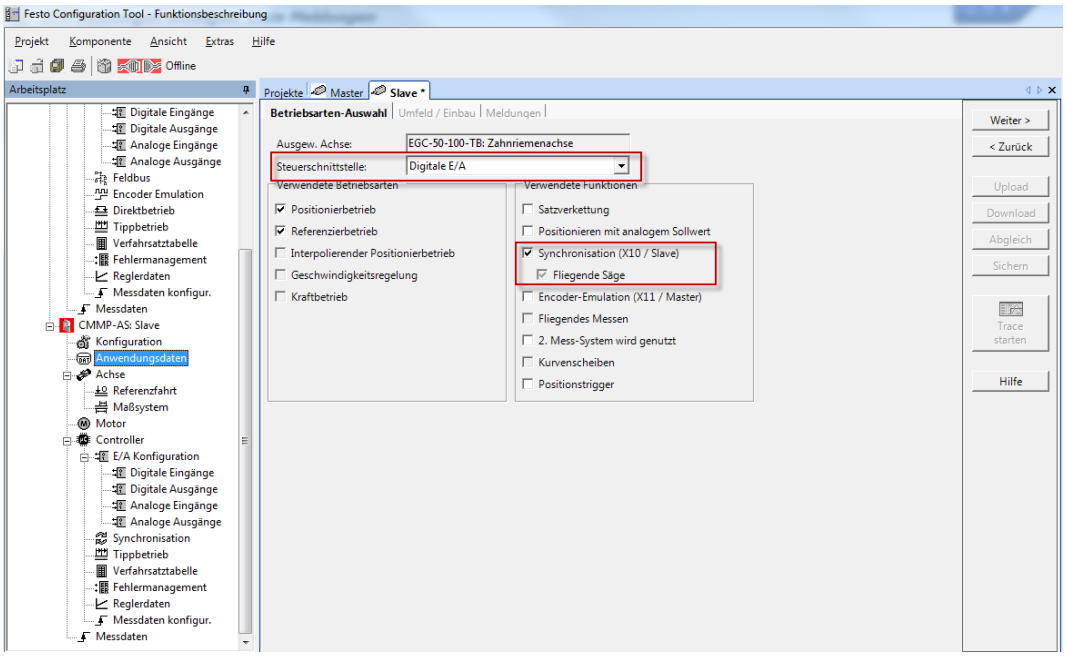
4.3.1 Parallel setup of two synchronous axes

Add a new CMMP controller using the “Add” function in the “Components” menu and assign a name to it. Acknowledge by clicking OK.



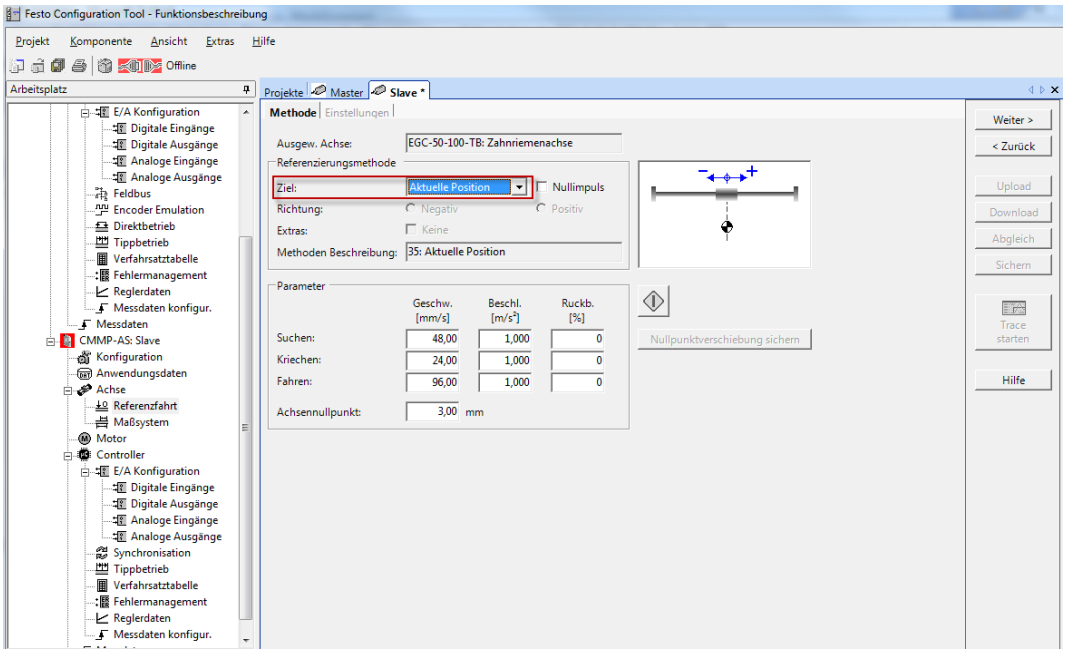
After configuration of the utilised components has been completed, synchronisation must be activated at the second controller.

Click “Application Data” and then “Operating Mode Selection” to this end.

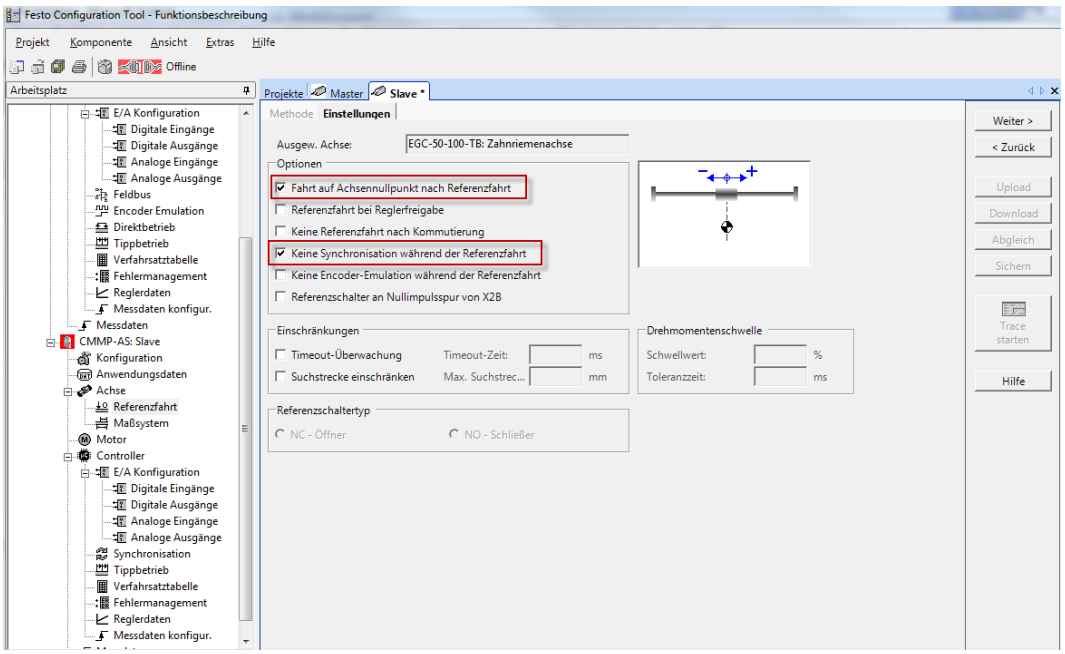


Configure homing for your slave.

In the case of synchronisation with a master axis for identical travel, it's advisable to set the homing method to "Current Position". This makes subsequent alignment of the two axes to each other possible (see detailed description in section 4.2.1).

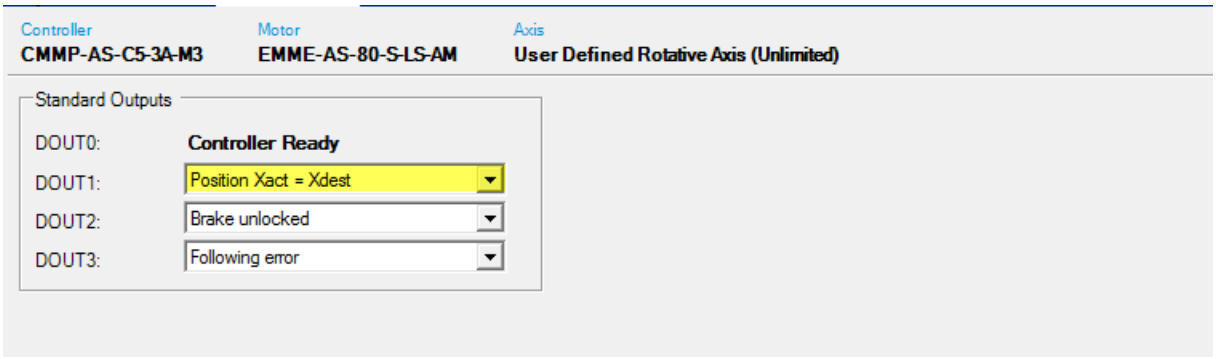


Homing options can be selected in "Settings" under "Homing". The corresponding option can be selected in order to be independent of the synchronisation signal during homing.



In the case of digital outputs, one is set to “Position Xact = Xtarget” so that a digital output can be queried for synchronous positioning of the slave. The output is active as soon as the actual position is the same as the target position within the scope of the configured window (MC).

This is the only way to query synchronous positioning of the slave. The output must be correspondingly connected to the PLC.



Set the synchronisation data in accordance with your master and the utilised mechanical system (e.g. a gearbox).

Our master reads out an A/B signal, so that this is also selected for the slave.

The line count also corresponds to that of the master.

Reversal of direction is used in order to change the slave’s direction of rotation. It serves as a substitute for reversal of direction in “Environment/Installation” under “Application Data”.

Certain transmission ratios for synchronization can be selected under “Gear Ratio”, so that the axes run synchronously as desired.

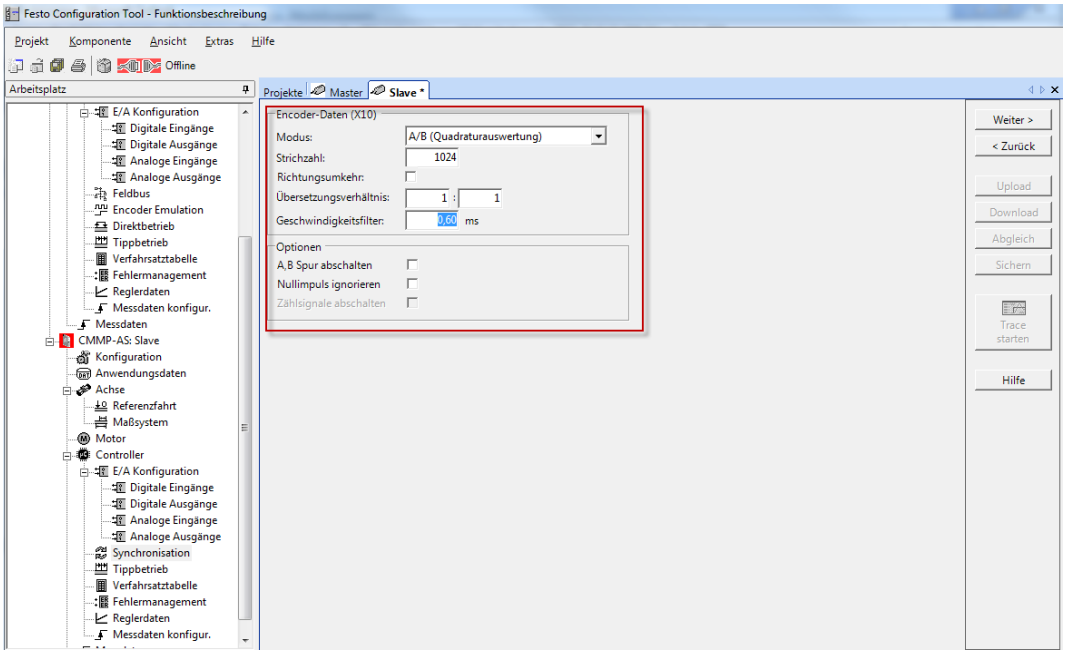
Since no gearbox is used in this example, the ratio is set to 1:1.

The speed filter shows the sampling rate at which the X10 interface is queried.

This setting remains at its default value of 0.6 ms.

The options provide you with the opportunity of deactivating or ignoring the A, B track or the zero pulse.

Neither is selected in this example.



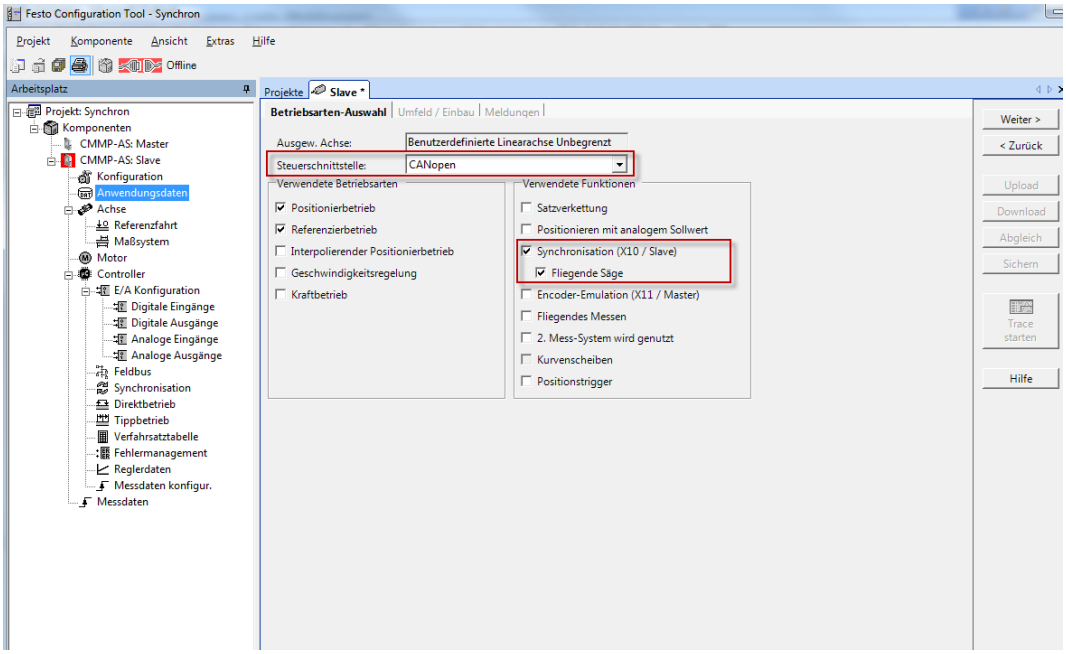
In order to control synchronisation, various positioning records must be created by means of which synchronisation is started or ended.

Different possibilities are available to this end, which are described as of section 5.2.

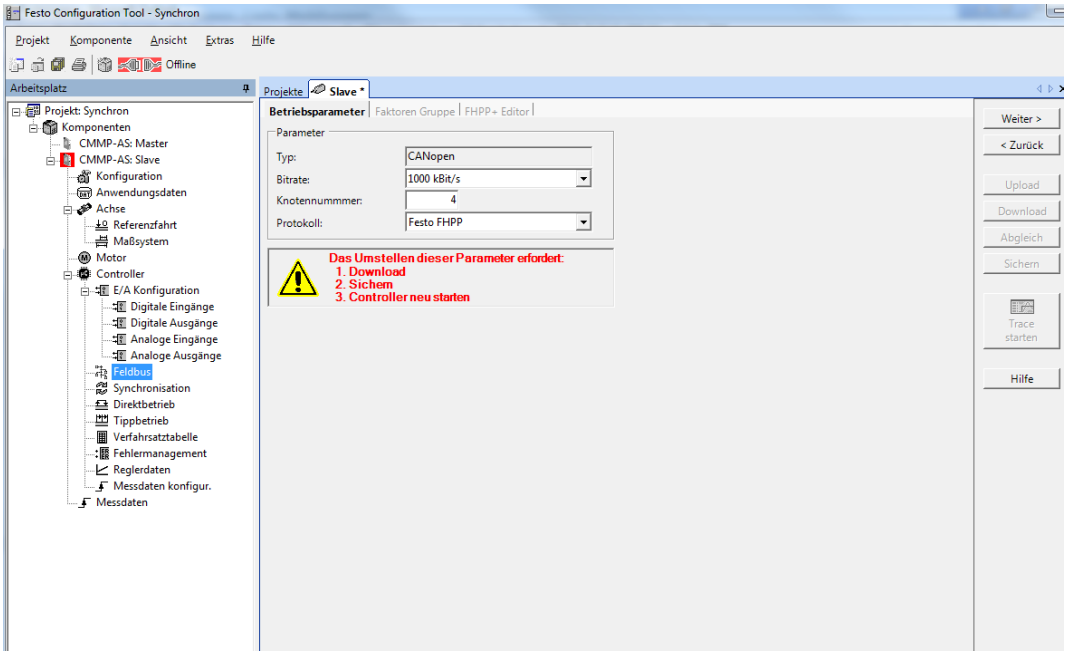
4.4 Slave configuration – CANopen control interface

Basic configuration of the slave is described as of section 4.2.
The differences are considered in this section – most of the settings for the CANopen interface are already described for the master axis (see section 4.1.1).

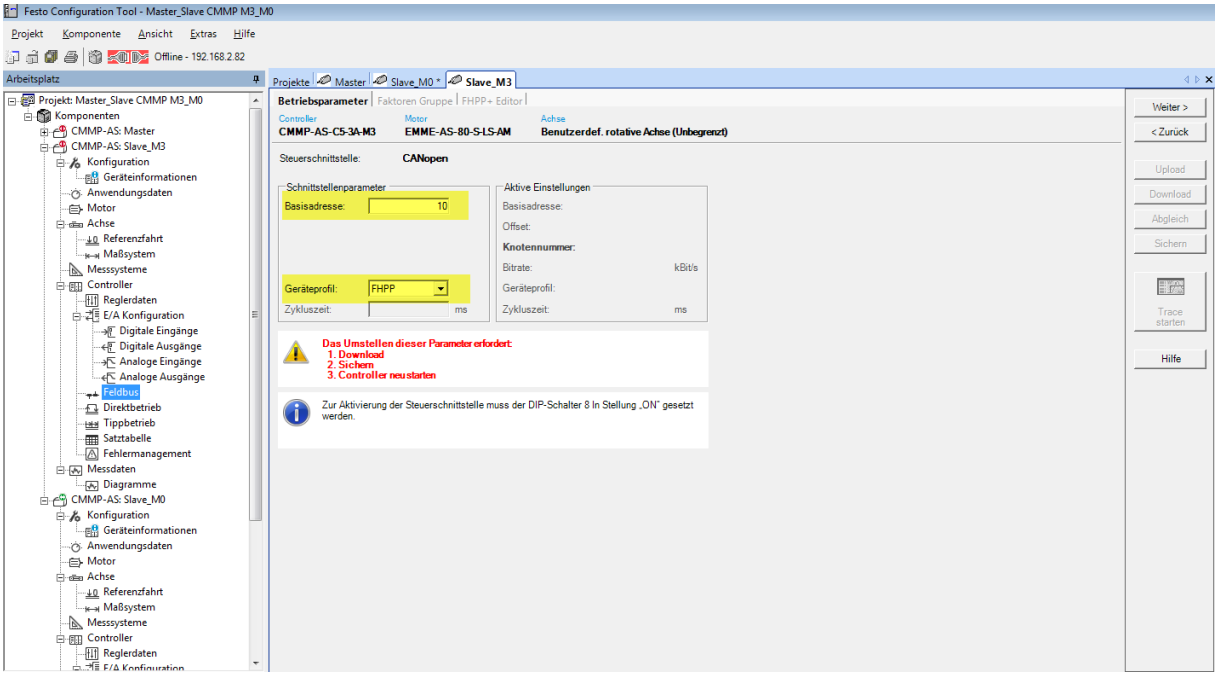
The control interface is set to CANopen under operating mode selection in the application data menu.
Then select synchronisation and the flying saw.



A new menu item then appears in the tree, namely “Fieldbus”.
The corresponding bus settings must be entered here, so that the controller can communicate with the control system.



In the case of the CMMP-...-M3, the tab in FCT appears as follows:



The basic address and the device profile (in this case FHPP) are selected here.

An offset for the CAN address, the transmission speed and activation of the fieldbus are set via the DIP switches on the plug-in module (Ext3).

DIL switch	CANopen/DriveBus	DeviceNet	PROFIBUS	PROFINET	Ethernet/IP	EtherCAT
	Onboard	CAMC-DN plugged in	CAMC-PB plugged in	CAMC-F-PN plugged in	CAMC-F-EP plugged in	CAMC-EC plugged in
1	NN bit 0	NN bit 0	NN bit 0	Not used		
2	NN bit 1	NN bit 1	NN bit 1	Not used		
3	NN bit 2	NN bit 2	NN bit 2	Not used		
4	NN bit 3	NN bit 3	NN bit 3	Not used		
5	NN bit 4	NN bit 4	NN bit 4	Not used		
6	Bit rate	Bit rate	NN bit 5	Not used		
7	Bit rate	Bit rate	NN bit 6	Not used		
8	Activation of fieldbus					
NN = node number						

Tab. 3.1 Setting of bit rate and node number

DIL switch	1 Mbit/s[1)]	500 kBit/s	250 kBit/s	125 kBit/s
6	On	Off	ON	Off
7	ON	ON	Off	Off

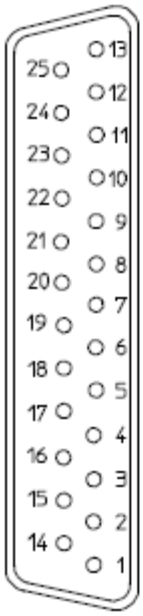
1) Only for CANopen/DriveBus; for DeviceNet, is limited to 500 kBit/s

Tab. 3.2 Setting of bit rate for CANopen and DeviceNet

DIL switch 8	Fieldbus
1	always activated
0	always off

Tab. 3.3 Activation of the fieldbus

As is also the case with the CMMP-...-M3, the CMMP-...-M0 is not equipped with any plug-in modules. The bus is configured directly via the I/O interface:

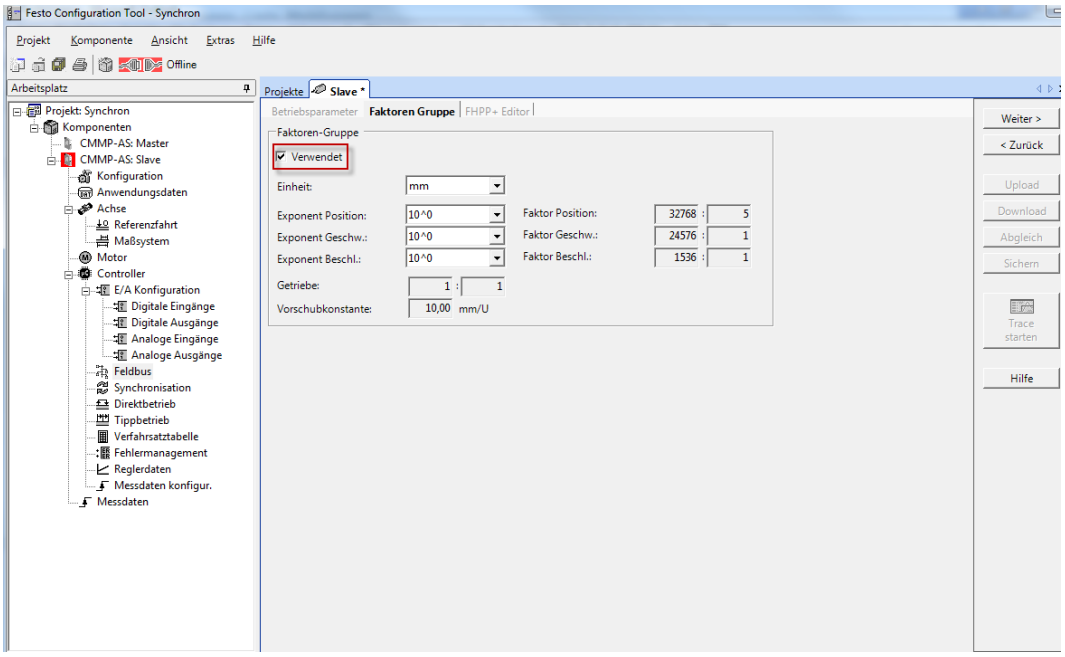
[X1]	Pin no.		Desig-nation	Specification
		13	DOUT3	Output freely parameterisable, optionally parameterisable as DIN11
	25		DOUT2	Output freely parameterisable, optionally parameterisable as DIN10
		12	DOUT1	Output freely parameterisable
	24		DOUT0	Controller ready, output permanently assigned
		11	DIN 9	Fieldbus data profile (CiA 402, FHPP), input freely parameterisable
	23		DIN 8	Fieldbus activation communication, input freely parameterisable
		10	DIN7	Limit switch 1 (blocks n < 0), input permanently assigned
	22		DIN6	Limit switch 0 (blocks n > 0), input permanently assigned
		9	DIN5	Controller enable, input permanently assigned
	21		DIN4	End stage enable, input permanently assigned
		8	DIN 3	Fieldbus offset node number bit 3, input freely parameterisable
	20		DIN 2	Fieldbus offset node number bit 2, input freely parameterisable
		7	DIN 1	Fieldbus offset node number bit 1, input freely parameterisable
	19		DIN 0	Fieldbus offset node number bit 0, input freely parameterisable
		6	GND24	Reference potential for digital I/Os
	18		+24 V	24 V output
		5	AOUT1	Analogue output freely parameterisable
	17		AOUT0	Analogue output freely parameterisable
		4	+VREF	Reference output for setpoint potentiometer
	16		DIN13	Fieldbus transmission rate bit 1, optionally parameterisable as AIN2
		3	DIN12	Fieldbus transmission rate bit 0, optionally parameterisable as AIN1
	15		#AIN0	Setpoint input 0, differential analogue input
		2	AIN0	
	14		AGND	Reference potential for analogue signals
		1	AGND	Screening for analogue signals, AGND

Tab. 4.3 Pin assignment: I/O communication [X1] (firmware factory setting)

As is also the case with the CMMP-...-M3, only the basic address is configured via FCT.

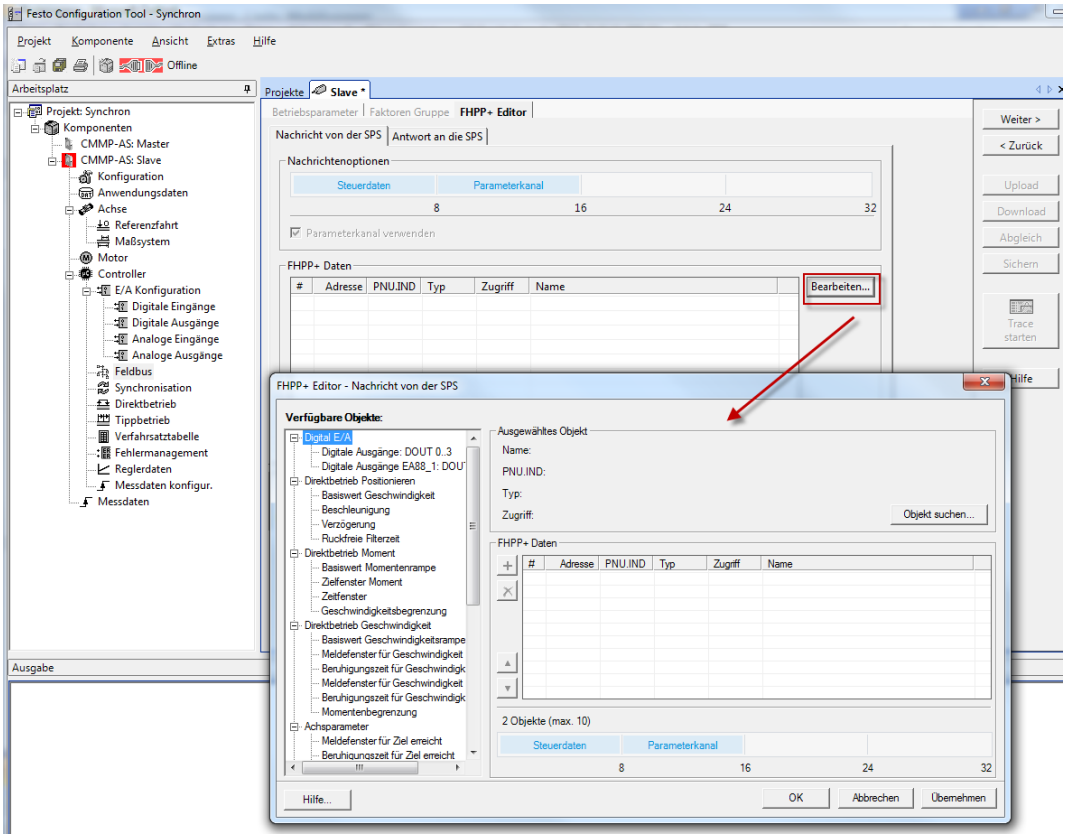
The protocol (FHPP, CiA 402), activation of the fieldbus, address offset and transmission speed are set via the I/O interface.

Use of the factors group can be activated in the “Factors Group” tab. This provides you with the option of, for example, displaying position at the module by means of a user-defined unit of measure instead of as an incremental value.



If you use the “Festo FHPP” protocol, cyclic values can be configured in the FHPP+ Editor which are also transmitted to or from the PLC for each cycle.

After clicking the “Edit ...” button, a new window appears at which the values can be selected.



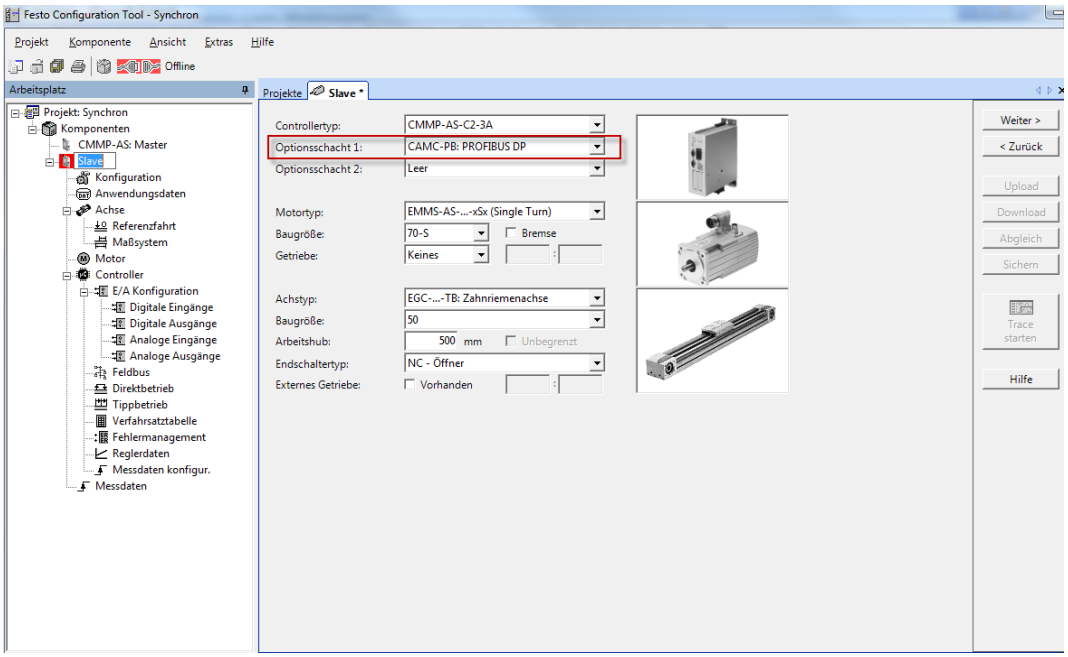
4.5 Slave configuration – Profibus control interface

Basic configuration of the slave is described as of section 4.2.

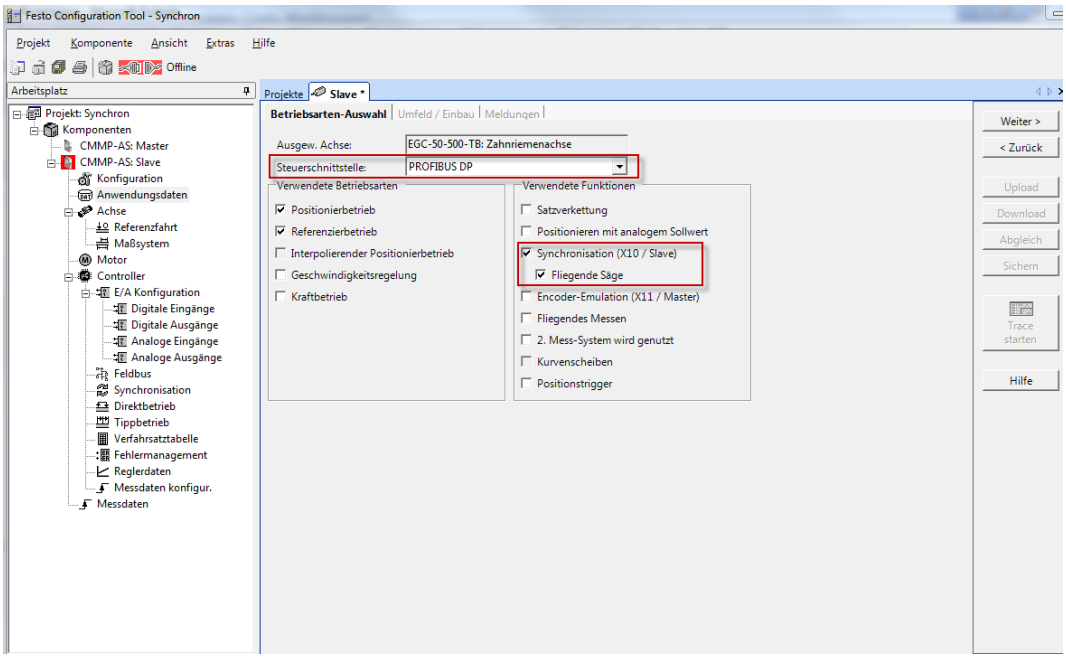
The differences are considered in this section – most of the settings for the Profibus interface are already described for the master axis (see section 3.1.2).

In order to be able to use the Profibus at the controller, you'll need an additional plug-in module for insertion into slot 1 or 2.

This must be correspondingly configured in FCT.

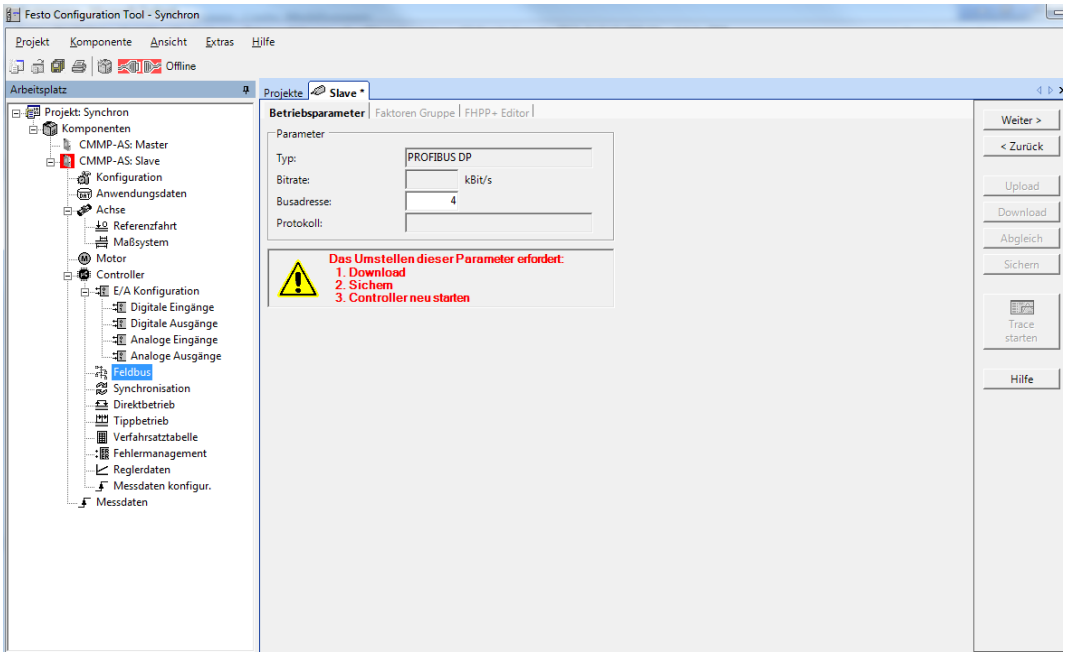


The control interface must be changed to Profibus DP in the application data menu and the synchronisation and flying saw functions have to be activated.

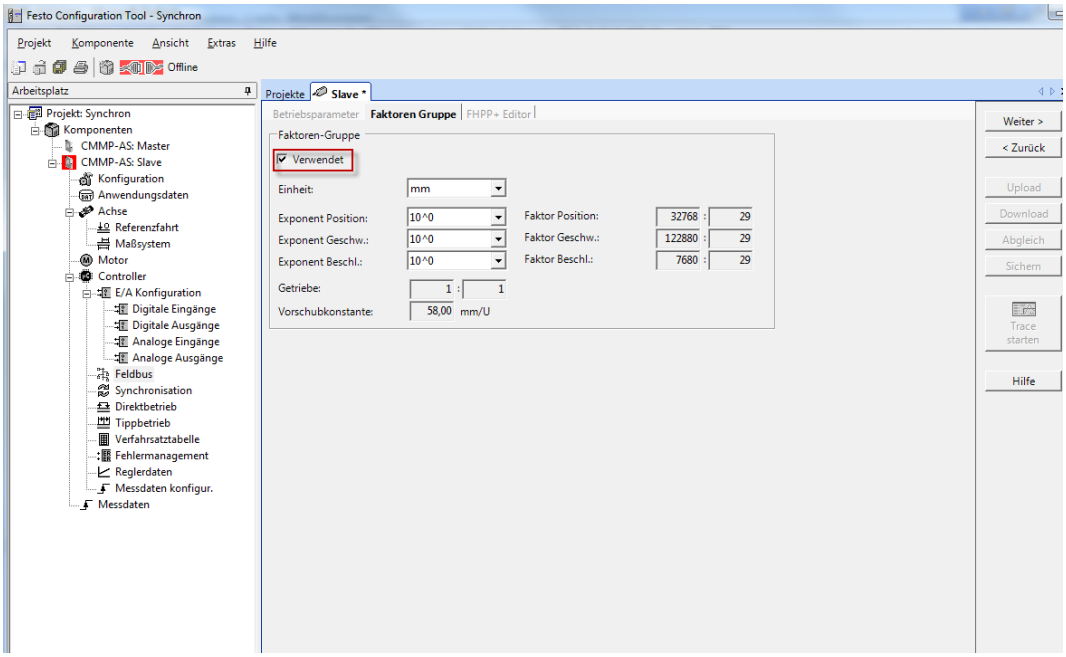


A new menu item then appears in the tree, namely “Fieldbus”.

The corresponding bus settings must be entered here, so that the controller can communicate with the control system.

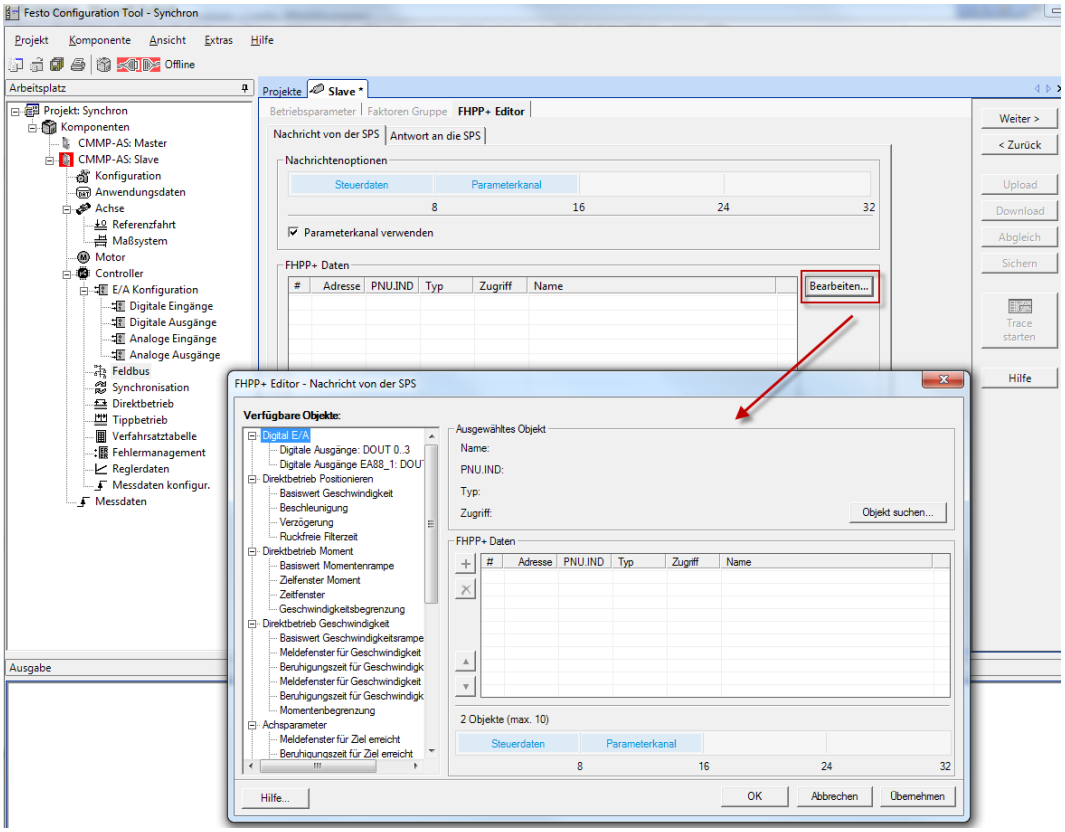


Use of the factors group can be activated in the “Factors Group” tab. This provides you with the option of, for example, displaying position at the module by means of a user-defined unit of measure instead of as an incremental value.



If you use the “Festo FHPP” protocol, cyclic values can be configured in the FHPP+ Editor which are also transmitted to or from the PLC for each cycle.

After clicking the “Edit ...” button, a new window appears at which the values can be selected.



5 Starting synchronization

5.1 Synchronisation via the “synchronisation” control interface at the CMMP

Synchronisation in this operating mode is active immediately after the controller is restarted.

In order to deactivate synchronisation once again, a digital input can be assigned (different than the start input, e.g. Din 9). Synchronisation is stopped when a trailing edge occurs at this input.

Synchronisation can be restarted by permanently setting the corresponding digital input to “1” (in this case Din 9).

And understanding of the input’s function is important when using a digital input for controlling synchronisation of the slave.

After restarting the slave controller, the input is edge-sensitive, i.e. synchronisation is not deactivated until a trailing edge occurs at this input. Whether the input is high or low is unimportant at first. Nor does a rising edge have any effect.

After the first trailing edge at the input, synchronisation is evaluated as level-sensitive.

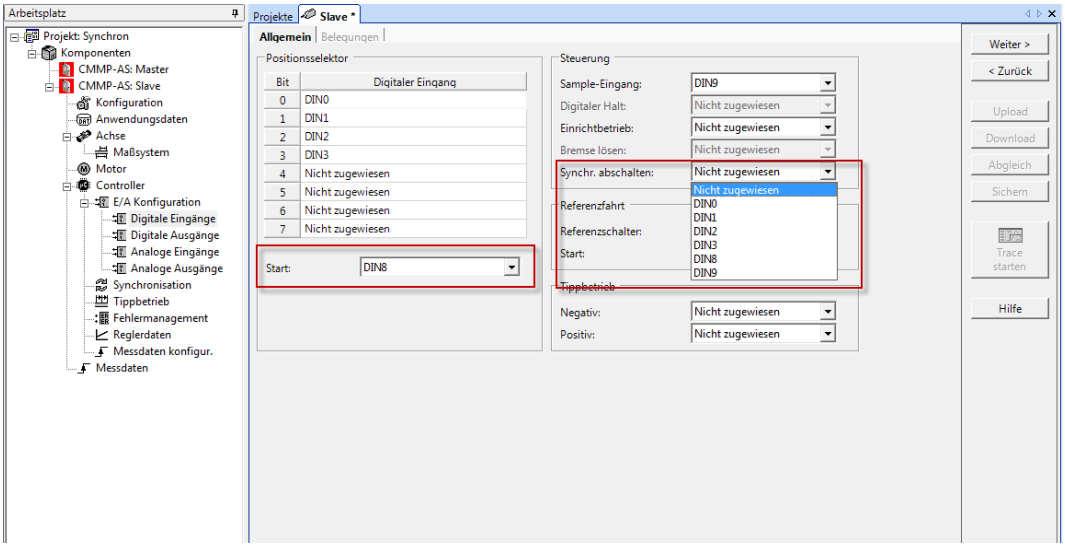
In order to reactivate synchronisation, the input must be continuously high. The function is deactivated with a continuously low signal.

Sample application: infinite master (e.g. belt drive) and slave axis (e.g. EGC-80-200-...):

The slave is stopped by means of an emergency off (amongst other factors controller enable: low). Synchronisation can now be safely switched off with the corresponding input. By placing the drive back under closed-loop control, undesired start-up of the axis can be prevented regardless of the active positioning record.

The slave can also be synchronised to a moving master.

However, the slave is only synchronised to the corresponding speed. Synchronisation to a moving axis, including synchronisation to the position of the master (see section 4.2.2), is not possible here.



5.2 Synchronisation via the “I/O” control interface at the CMMP

5.2.1 Synchronisation of two parallel axes

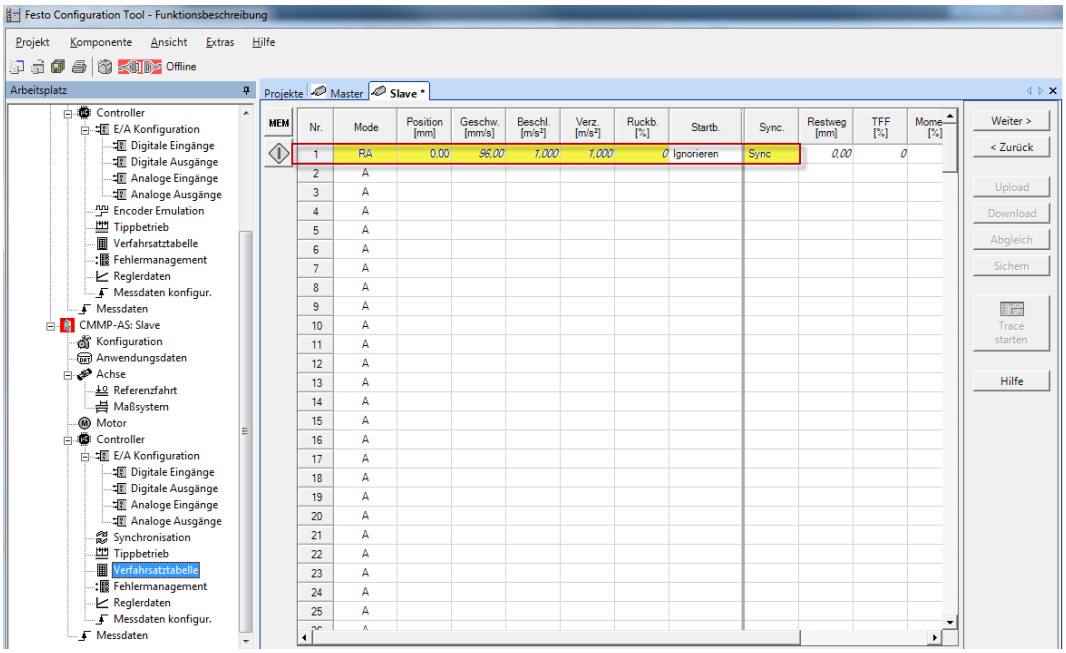
Synchronisation is started and stopped again via the slave’s positioning records table.

In order to have the slave axis run synchronous to the master axis, it’s operated relative to the position of the master (mode: RA).

The position corresponds to the same master position (position: 0.00).

Speeds (additional delta speed relative to the master), acceleration rates and jerk limitation are the values by means of which the slave attempts to synchronise itself to the master.

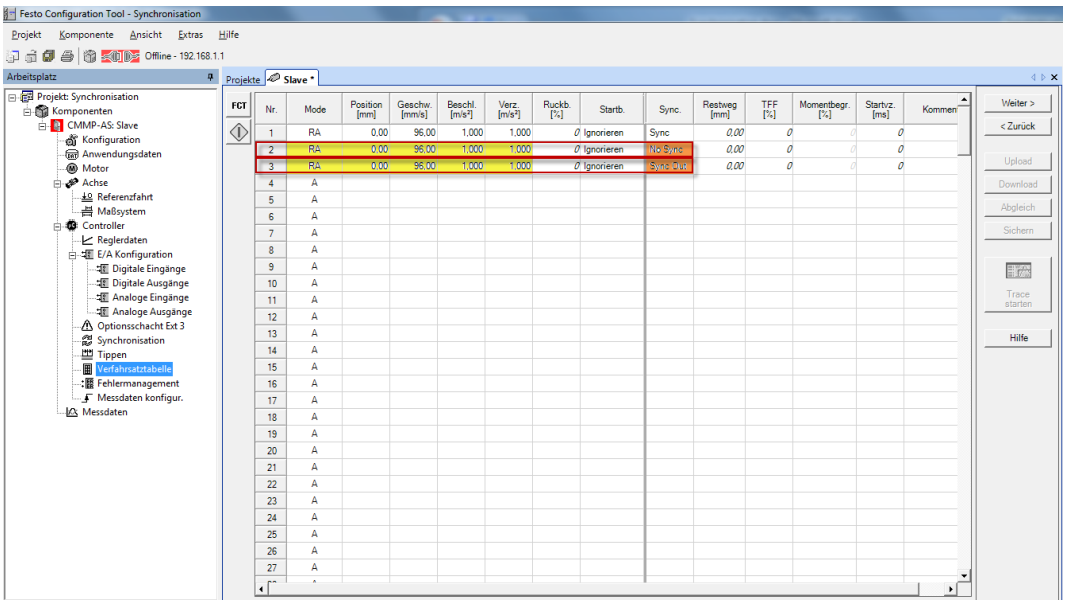
Synchronisation is activated with the first (RA) positioning record (synchronisation: sync).



There are two ways to switch synchronisation back off again.

A new positioning record has to be created in both cases (speeds, acceleration rates etc. can be adjusted).

The difference involves the selection in the sync column. This option is set to “No Sync” or “Sync out”. With this type of synchronisation (synchronisation of two parallel axes), the selection does not result in any difference with regard to desynchronisation because, as a rule, the axes are mechanically linked and deactivation of synchronisation only makes sense at a standstill.



In the case of master-slave operation of two parallel axes (e.g. basic axes of a Cartesian gantry), the following procedure is recommended:

The cross member must be mechanically aligned such that the motors draw nearly identical current at standstill (check via the controller's trace function: actual current).
The positions of the two axes are set separately.

When the zero pulse is activated at both controllers (master and slave), the motors must first of all be aligned to the zero pulse while disengaged from the mechanical system.
The motors are operated in the synchronous mode to this end until the slave has synchronised itself to the master's zero pulse after one revolution. The motors can then be remounted to the axis with this setting.
When the zero pulse occurs, the slave is always synchronised to the master's zero pulse, and thus if there's any offset from the position on the part of the slave, it's corrected and synchronised to the master's position after one revolution (status in Sept. 2015: zero pulse doesn't work).

After aligning the cross member, synchronisation of the slave is activated before initiating travel at the master, in this case by starting positioning record 1.
Synchronous positioning can be queried via the correspondingly configured output (Xactual = Xtarget).
The slave waits for corresponding signals from the master. As soon as the master is set into motion, the slave follows its position.
Homing can now be started via the master. As soon as the master has been homed, the slave can also be homed to the "current position". The value should be saved to the encoder in both cases.
The system of measures at the slave must be selected such that it cannot be advanced into the software end positions when the master is set into motion. This may lead to problems if the value saved to the encoder lies outside of the limits or close to them!

The actual positions of both axes can be monitored by means of a PLC in order to detect offsets as early as possible and correct the slave's positon.
In particular after an emergency off, it's advisable to check the positions in order to avoid damage to the mechanical system.
In order to once again deactivate synchronisation, for example in order to align the slave, positioning record 2 or 3 must be invoked. The axis is no longer synchronised. If the two axes (master and slave) are mechanically coupled, the master axis must not be operated any more.

General comments:

If synchronisation is not explicitly deactivated (digital input, positioning record, cam disc function), it remains active even when control of the slave is toggled!

The function of the digital input for controlling synchronisation is described in detail in section 4.1.

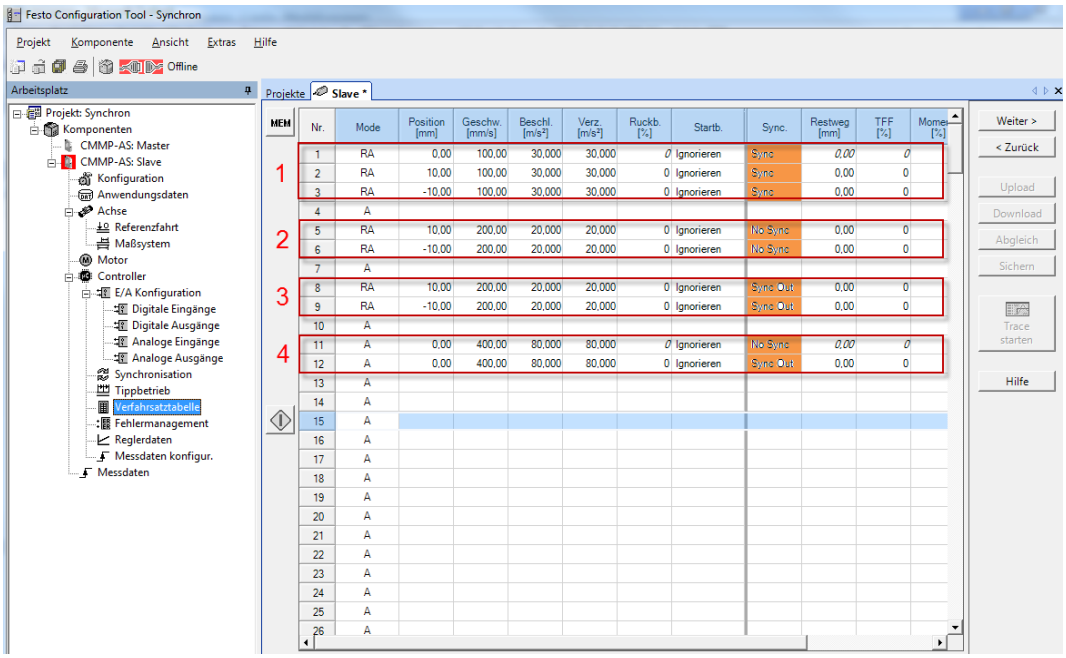
5.2.2 Synchronisation to an axis / a belt

In the case of this application, the slave axis synchronises itself to an axis or a belt which is already in motion. And thus the slave must catch up to the master's position and speed after synchronisation is started.

The slave's basic settings are described in section 4.3.1.

The characteristics of the individual positioning records are explained in the following, as well as how the slave axis responds to them.

As an example, it will be assumed that the master rotates infinitely at a constant speed.



(1) Synchronisation is started with one of the three positioning records by means of a corresponding trigger.
Selection of the position is relative to the master at the point in time of the trigger signal. Positioning records 2 and 3 provide you with the opportunity of shifting the position, in this case by ±10 mm relative to the master.
The speed set in the positioning records is a delta value. In this case, the slave travels up to 100 mm per second faster than the master with the configured acceleration and deceleration, in order to catch up with its position.
The acceleration and deceleration values are absolute values independent of the master.

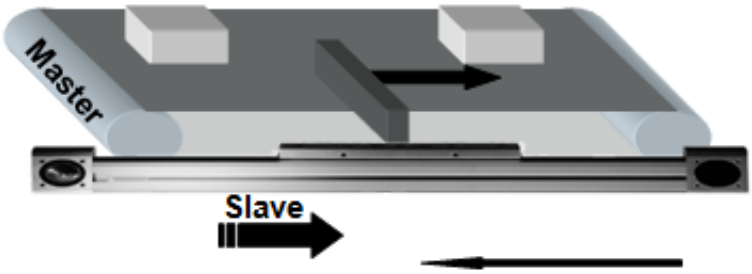
(2) This is an option for stopping synchronisation (No Sync). In this case, upon selecting the positioning record, the slave continues to travel 10 mm relative at the configured speed and acceleration rate. The respective positioning record must be selected depending on the direction of travel.

(3) This type of desynchronisation (Sync Out) is similar to the type described in point 2. The difference is that the configured speed is not used, but rather the speed of the master at the point in time of desynchronisation.

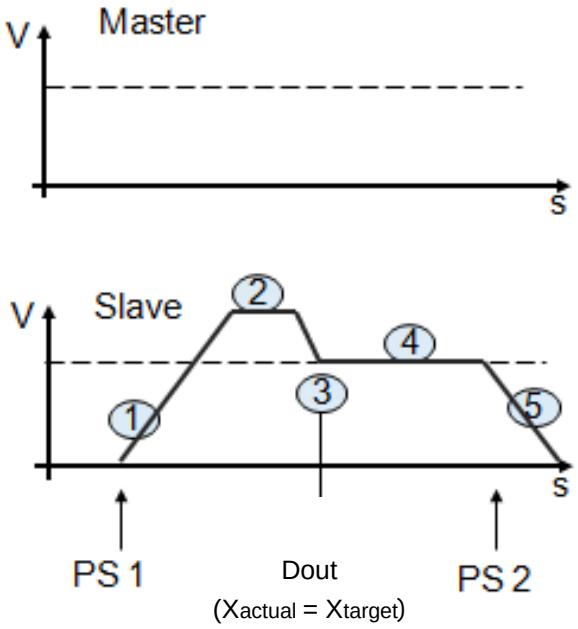
(4) These positioning records provide you with the opportunity of traveling to an absolute position after desynchronisation. As already described in points (2) and (3), travel is possible at the configured speed (No Sync) or the master's speed upon desynchronisation (Sync Out).

5.2.3 Sample application – synchronisation

The slave needs to be synchronised to a moving master:



The master travels at a constant speed.
When a certain trigger signal occurs (1), the slave synchronises itself to the master's position at a corresponding delta speed (2).
As soon as the slave is synchronised to the master, a digital output is set (if configured, Xactual = Xtarget) (3).
Synchronous positioning is maintained (4) until the desynchronisation signal occurs (5).



5.3 Synchronisation via Profibus at the CMMP – S7 example

Configure the controller in accordance with the instructions and add it to your control system.

There are basically two different ways to control synchronisation via the Profibus.

In this case, for the purpose of explanation, a CMMP controller and the function block with revision level 08/2011 (available from the Support Portal) are used.

In the case of parallel axes, a motor with multi-turn encoder should be used if possible.

After one-time only mechanical alignment of the axes, the current positions of the master and the slave can be monitored via the bus.

If a difference is detected, synchronisation can be stopped and the position of the slave can be corrected. The slave is regulated to the master's actual position.

In general, after switching the master and the slave on, the slave should be regulated to the master's position. This ensures that both axes are parallel and that stressing due to misalignment of the transverse axis is avoided.

5.3.1 Positioning record selection

Synchronisation can be controlled via the Profibus with the help of configured positioning records.

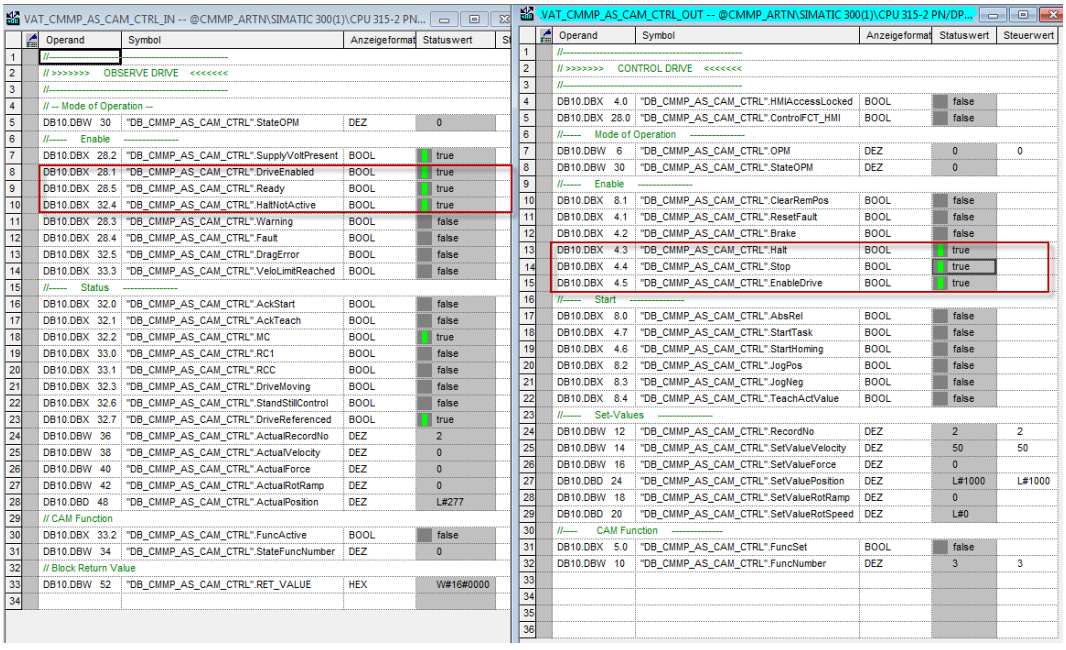
Positioning records are created in accordance with the type of synchronisation to this end, as described in sections 5.2.1 and 5.2.2.

The positioning records are started via the bus, in order to achieve the desired closed-loop control.

Fundamental control of slave synchronisation is explained on the basis of the variables table in the sample project for the function block.

In order to set the controller to closed-loop control, the following, corresponding hardware signals must be available: output stage enable (Din4), controller enable (Din5), safe stop (terminal X3, pin 2) and if applicable the proximity switch (Din 6 and Din 7) if NC limit switches are used.

Set "Halt", "EnableDrive" and "Stop", one after the other after the respective acknowledgement, until the controller is set to closed-loop control ("Ready" == 1).



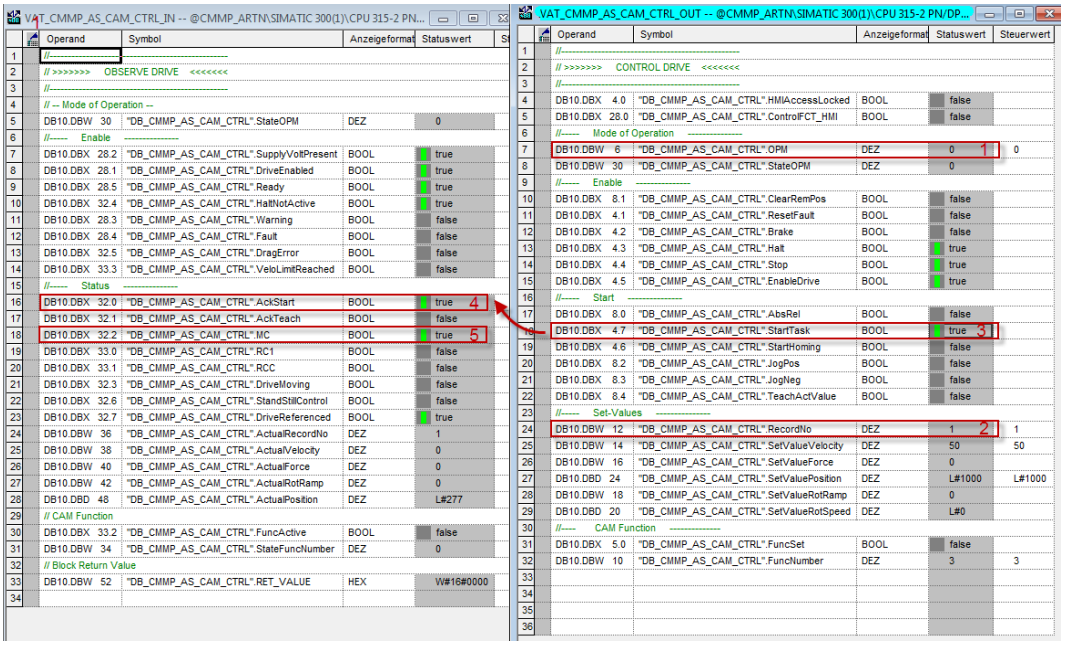
In order to be able to activate the positioning records, the controller must be in operation mode (OPM) 0 – corresponds to positioning record selection.

The desired positioning record for starting synchronisation can be selected via “RecordNo” (in this example positioning record number 1), and then started by applying a rising edge to “StartTask”.

As soon as the “AckStart” bit arrives, “StartTask” can be reset to 0.

“MotionComplete” (MC) goes to 0. But there’s no signal that can be used for synchronous positioning. If a new positioning record is started during motion, it remains at 0 although the slave might no longer be synchronous.

In this case, synchronisation must be checked via a digital output at the slave controller (see section 3.3.1).



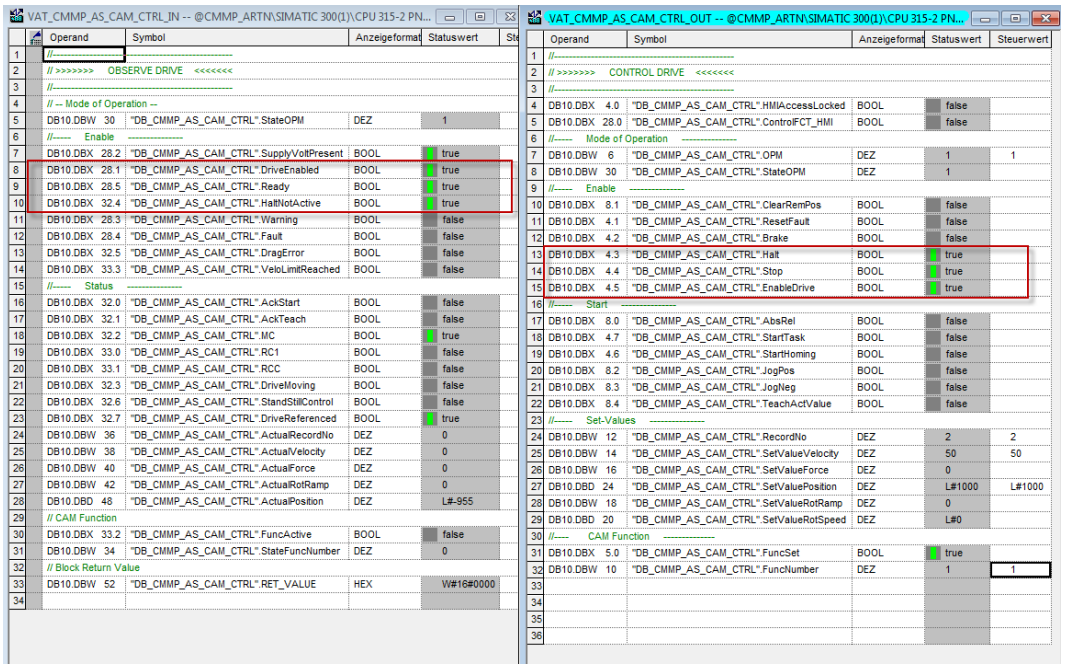
In order to once again stop synchronisation of the slave, another appropriate positioning record must be selected from the table of configured positioning records and executed.

5.3.2 Direct drive – use of the cam disc function

With the help of this function, the slave can be synchronised without positioning records via direct drive. However, this is only possible with the CMMP premium controller.

The controller must first of all be set to closed-loop control. The following, corresponding hardware signals must be available to this end: output stage enable (Din4), controller enable (Din5), safe stop (terminal X3, pin 2) and if applicable the proximity switch (Din 6 and Din 7) if NC limit switches are used.

Set “Halt”, “EnableDrive” and “Stop”, one after the other after the respective acknowledgement, until the controller is set to closed-loop control (“Ready” == 1).



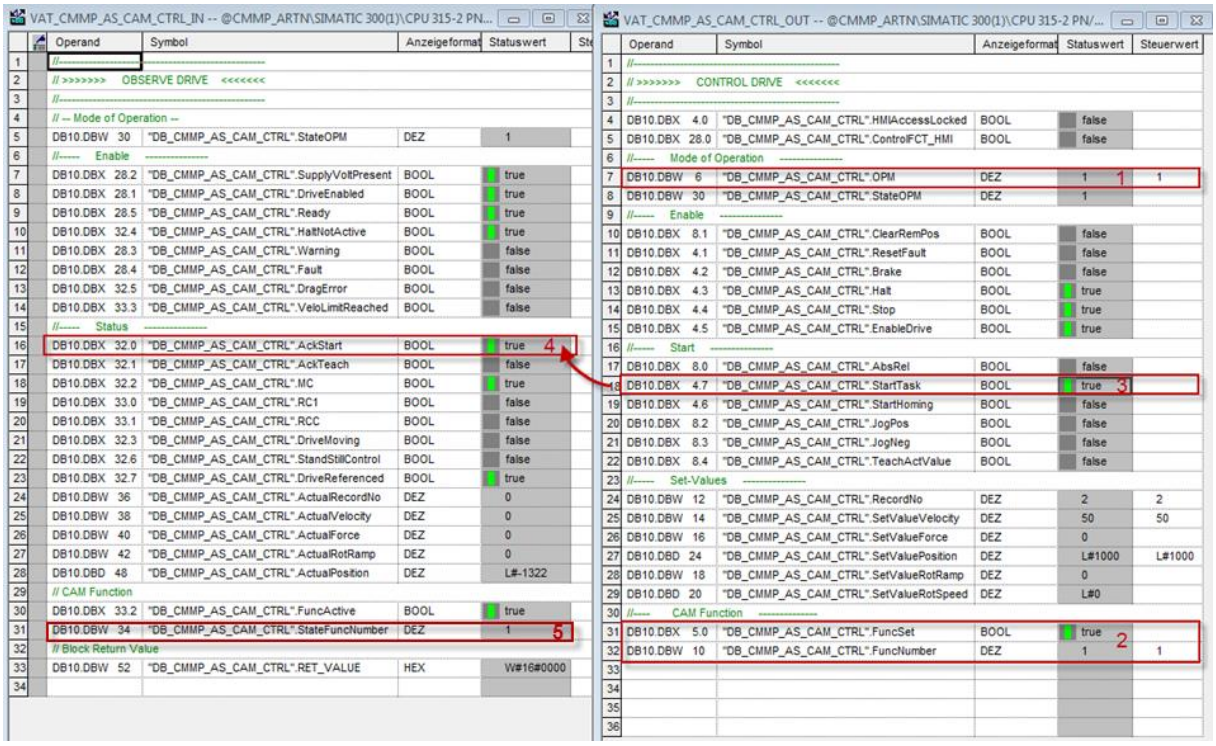
In order to operate the controller in the direct drive mode, “OPM” must be set to 1.

The slave can be synchronised with the help of the cam disc functions.

“FuncSet” is activated to this end, and the “FuncNumber” is set to 1. Synchronisation to an external input is active.

Synchronisation is activated by applying a rising edge to “StartTask”.

As soon as the “AckStart” bit arrives, “StartTask” can be reset to 0.



VAT_CMMP_AS_CAM_CTRL_IN -- @CMMP_ARTN\SIMATIC 300(1)\CPU 315-2 PN...

	Operand	Symbol	Anzeigeformat	Statuswert
1	//			
2	// >>>>>>> OBSERVE DRIVE <<<<<<<			
3	//			
4	// -- Mode of Operation --			
5	DB10.DBW 30	"DB_CMMP_AS_CAM_CTRL".StateOPM	DEZ	1
6	//------ Enable ----------			
7	DB10.DBX 28.2	"DB_CMMP_AS_CAM_CTRL".SupplyVoltPresent	BOOL	true
8	DB10.DBX 28.1	"DB_CMMP_AS_CAM_CTRL".DriveEnabled	BOOL	true
9	DB10.DBX 28.5	"DB_CMMP_AS_CAM_CTRL".Ready	BOOL	true
10	DB10.DBX 32.4	"DB_CMMP_AS_CAM_CTRL".HaltNotActive	BOOL	true
11	DB10.DBX 28.3	"DB_CMMP_AS_CAM_CTRL".Warning	BOOL	false
12	DB10.DBX 28.4	"DB_CMMP_AS_CAM_CTRL".Fault	BOOL	false
13	DB10.DBX 32.5	"DB_CMMP_AS_CAM_CTRL".DragError	BOOL	false
14	DB10.DBX 33.3	"DB_CMMP_AS_CAM_CTRL".VeloLimitReached	BOOL	false
15	//------ Status ----------			
16	DB10.DBX 32.0	"DB_CMMP_AS_CAM_CTRL".AckStart	BOOL	true 4
17	DB10.DBX 32.1	"DB_CMMP_AS_CAM_CTRL".AckTeach	BOOL	false
18	DB10.DBX 32.2	"DB_CMMP_AS_CAM_CTRL".MC	BOOL	true
19	DB10.DBX 33.0	"DB_CMMP_AS_CAM_CTRL".RC1	BOOL	false
20	DB10.DBX 33.1	"DB_CMMP_AS_CAM_CTRL".RCC	BOOL	false
21	DB10.DBX 32.3	"DB_CMMP_AS_CAM_CTRL".DriveMoving	BOOL	false
22	DB10.DBX 32.6	"DB_CMMP_AS_CAM_CTRL".StandStillControl	BOOL	false
23	DB10.DBX 32.7	"DB_CMMP_AS_CAM_CTRL".DriveReferenced	BOOL	true
24	DB10.DBW 36	"DB_CMMP_AS_CAM_CTRL".ActualRecordNo	DEZ	0
25	DB10.DBW 38	"DB_CMMP_AS_CAM_CTRL".ActualVelocity	DEZ	0
26	DB10.DBW 40	"DB_CMMP_AS_CAM_CTRL".ActualForce	DEZ	0
27	DB10.DBW 42	"DB_CMMP_AS_CAM_CTRL".ActualRotRamp	DEZ	0
28	DB10.DBD 48	"DB_CMMP_AS_CAM_CTRL".ActualPosition	DEZ	L#-1322
29	// CAM Function			
30	DB10.DBX 33.2	"DB_CMMP_AS_CAM_CTRL".FuncActive	BOOL	true
31	DB10.DBW 34	"DB_CMMP_AS_CAM_CTRL".StateFuncNumber	DEZ	1 5
32	// Block Return Value			
33	DB10.DBW 52	"DB_CMMP_AS_CAM_CTRL".RET_VALUE	HEX	W#16#0000
34				

VAT_CMMP_AS_CAM_CTRL_OUT -- @CMMP_ARTN\SIMATIC 300(1)\CPU 315-2 PN/...

	Operand	Symbol	Anzeigeformat	Statuswert	Steuerwert
1	//				
2	// >>>>>>> CONTROL DRIVE <<<<<<<				
3	//				
4	DB10.DBX 4.0	"DB_CMMP_AS_CAM_CTRL".HMIAccessLocked	BOOL	false	
5	DB10.DBX 28.0	"DB_CMMP_AS_CAM_CTRL".ControlFCT_HMI	BOOL	false	
6	//------ Mode of Operation ----------				
7	DB10.DBW 6	"DB_CMMP_AS_CAM_CTRL".OPM	DEZ	1 1	1
8	DB10.DBW 30	"DB_CMMP_AS_CAM_CTRL".StateOPM	DEZ	1	
9	//------ Enable ----------				
10	DB10.DBX 8.1	"DB_CMMP_AS_CAM_CTRL".ClearRemPos	BOOL	false	
11	DB10.DBX 4.1	"DB_CMMP_AS_CAM_CTRL".ResetFault	BOOL	false	
12	DB10.DBX 4.2	"DB_CMMP_AS_CAM_CTRL".Brake	BOOL	false	
13	DB10.DBX 4.3	"DB_CMMP_AS_CAM_CTRL".Halt	BOOL	true	
14	DB10.DBX 4.4	"DB_CMMP_AS_CAM_CTRL".Stop	BOOL	true	
15	DB10.DBX 4.5	"DB_CMMP_AS_CAM_CTRL".EnableDrive	BOOL	true	
16	//------ Start ----------				
17	DB10.DBX 8.0	"DB_CMMP_AS_CAM_CTRL".AbsRel	BOOL	false	
18	DB10.DBX 4.7	"DB_CMMP_AS_CAM_CTRL".StartTask	BOOL	true 3	
19	DB10.DBX 4.6	"DB_CMMP_AS_CAM_CTRL".StartHoming	BOOL	false	
20	DB10.DBX 8.2	"DB_CMMP_AS_CAM_CTRL".JogPos	BOOL	false	
21	DB10.DBX 8.3	"DB_CMMP_AS_CAM_CTRL".JogNeg	BOOL	false	
22	DB10.DBX 8.4	"DB_CMMP_AS_CAM_CTRL".TeachActValue	BOOL	false	
23	//------ Set-Values ----------				
24	DB10.DBW 12	"DB_CMMP_AS_CAM_CTRL".RecordNo	DEZ	2	2
25	DB10.DBW 14	"DB_CMMP_AS_CAM_CTRL".SetValueVelocity	DEZ	50	50
26	DB10.DBW 16	"DB_CMMP_AS_CAM_CTRL".SetValueForce	DEZ	0	
27	DB10.DBD 24	"DB_CMMP_AS_CAM_CTRL".SetValuePosition	DEZ	L#1000	L#1000
28	DB10.DBW 18	"DB_CMMP_AS_CAM_CTRL".SetValueRotRamp	DEZ	0	
29	DB10.DBD 20	"DB_CMMP_AS_CAM_CTRL".SetValueRotSpeed	DEZ	L#0	
30	//------ CAM Function ----------				
31	DB10.DBX 5.0	"DB_CMMP_AS_CAM_CTRL".FuncSet	BOOL	true 2	
32	DB10.DBW 10	"DB_CMMP_AS_CAM_CTRL".FuncNumber	DEZ	1	1
33					
34					
35					
36					

Switching to synchronisation with the cam disc function can be monitored with the help of “StateFuncNumber”.

In contrast to starting synchronisation via positioning records, for example, it’s not possible to query synchronous positioning via a digital output.

5.4 Synchronisation via CAN at the CMMP – Codesys V3.5 example

Configure the controller in accordance with the instructions and add it to your control system.

There are basically two different ways to control synchronisation via the CAN bus.

In this case, for the purpose of explanation, a CMMP controller and the function block with revision level 10/2015 (available from the Support Portal) are used.

In the case of parallel axes, a motor with multi-turn encoder should be used if possible.

After one-time only mechanical alignment of the axes, the current positions of the master and the slave can be monitored via the bus.

If a difference is detected, synchronisation can be stopped and the position of the slave can be corrected. The slave is regulated to the master's actual position.

In general, after switching the master and the slave on, the slave should be regulated to the master's position (optimised mechanical alignment in advance is a prerequisite). This ensures that both axes are parallel and that stressing due to misalignment of the transverse axis is avoided.

The following two sections describe the two synchronisation options.

5.4.1 Positioning record selection

Synchronisation can be controlled via the CAN bus with the help of configured positioning records.

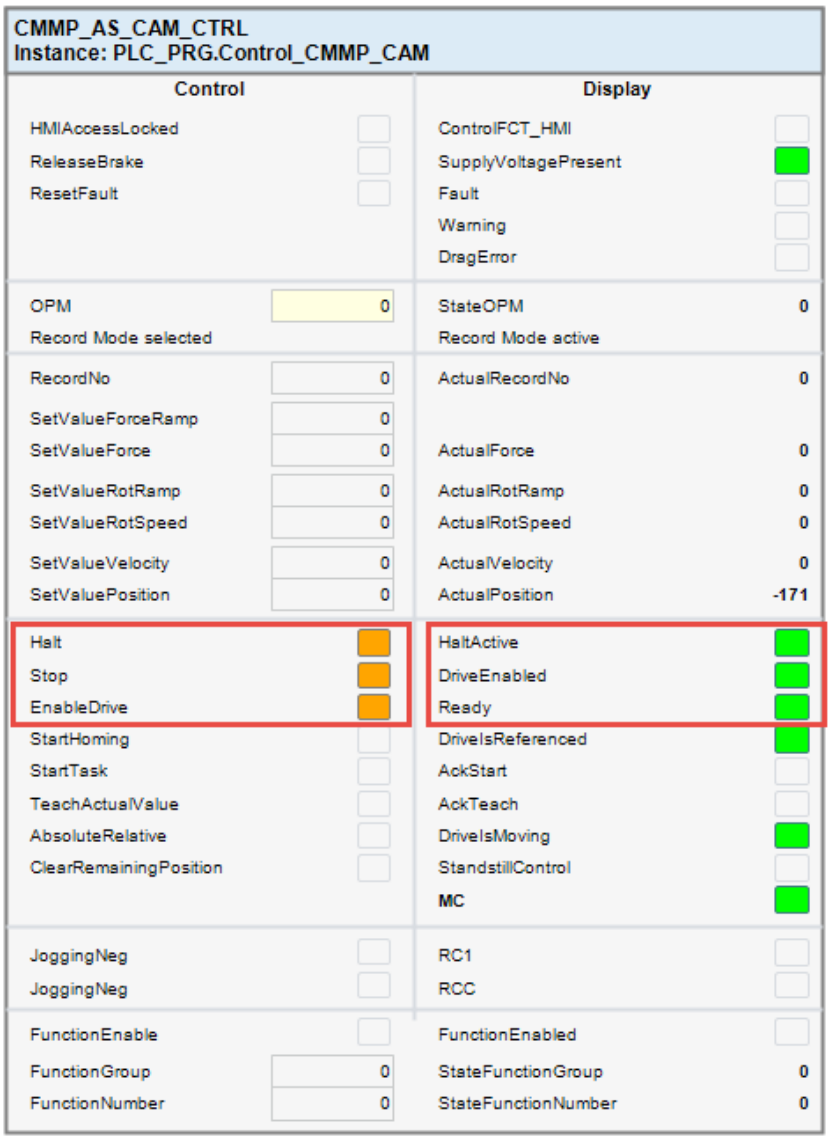
Positioning records are created in accordance with the type of synchronisation to this end, as described in sections 5.2.1 and 5.2.2.

The positioning records are started via the bus, in order to achieve the desired closed-loop control.

Fundamental control of slave synchronisation is explained on the basis of the variables table in the sample project for the function block.

In order to set the controller to closed-loop control, the following, corresponding hardware signals must be available: output stage enable (Din4), controller enable (Din5), safe stop (terminal X3, pin 2) and if applicable the proximity switch (Din 6 and Din 7) if NC limit switches are used.

Set “Halt”, “EnableDrive” and “Stop”, one after the other after the respective acknowledgement, until the controller is set to closed-loop control (“Ready” == 1).

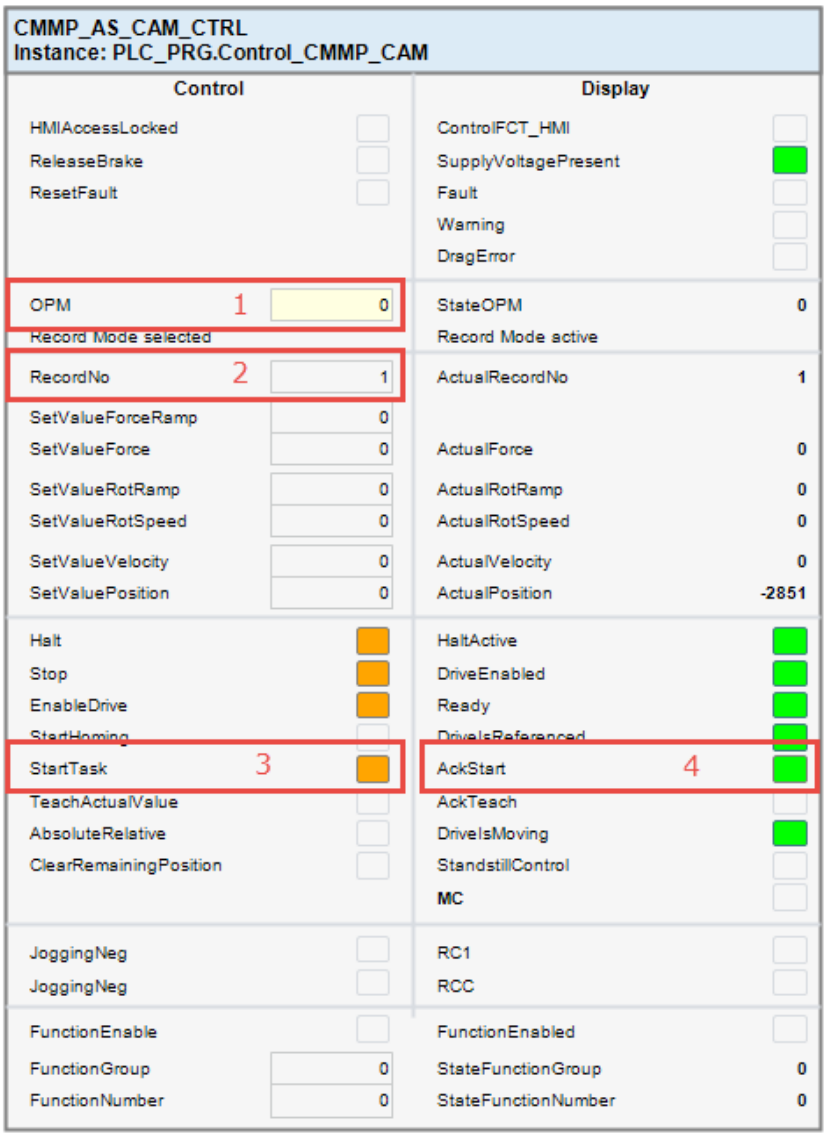


In order to be able to activate the positioning records, the controller must be in operation mode (OPM) 0 – corresponds to positioning record selection.

The desired positioning record for starting synchronisation can be selected via “RecordNo” (in this example positioning record number 1), and then started by applying a rising edge to “StartTask”.

As soon as the “AckStart” bit arrives, “StartTask” can be reset to 0.

However, there’s no feedback concerning the slave’s synchronous position. This must be conducted by means of a configured digital output at the slave controller (see section 3.3.1).



In order to once again stop synchronisation of the slave, another appropriate positioning record must be selected from the table of configured positioning records and executed.

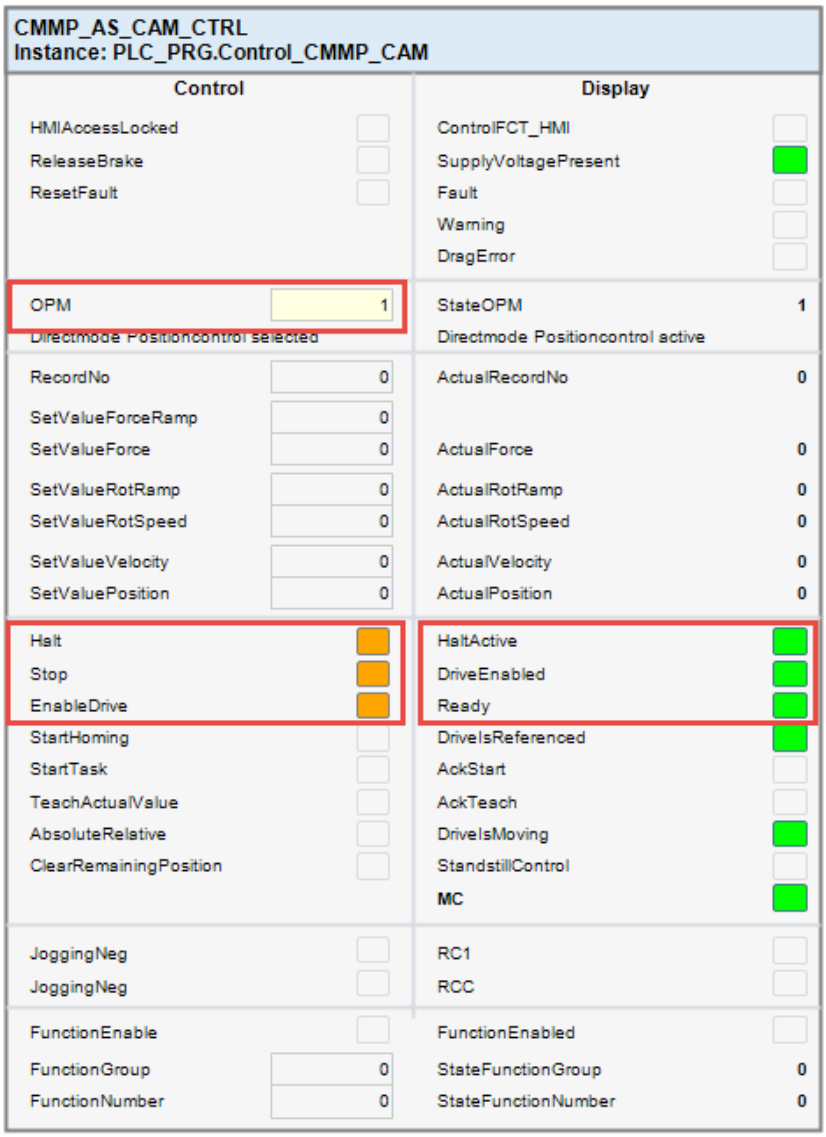
5.4.2 Direct drive – use of the cam disc function

With the help of this function, the slave can be synchronised without positioning records via direct drive. However, this is only possible with the CMMP premium controller.

The controller must first of all be set to closed-loop control. The following, corresponding hardware signals must be available to this end: output stage enable (Din4), controller enable (Din5), safe stop (terminal X3, pin 2) and if applicable the proximity switch (Din 6 and Din 7) if NC limit switches are used.

Set “Halt”, “EnableDrive” and “Stop”, one after the other after the respective acknowledgement, until the controller is set to closed-loop control (“Ready” == 1).

In order to operate the controller in the direct drive mode, “OPM” must be set to 1.

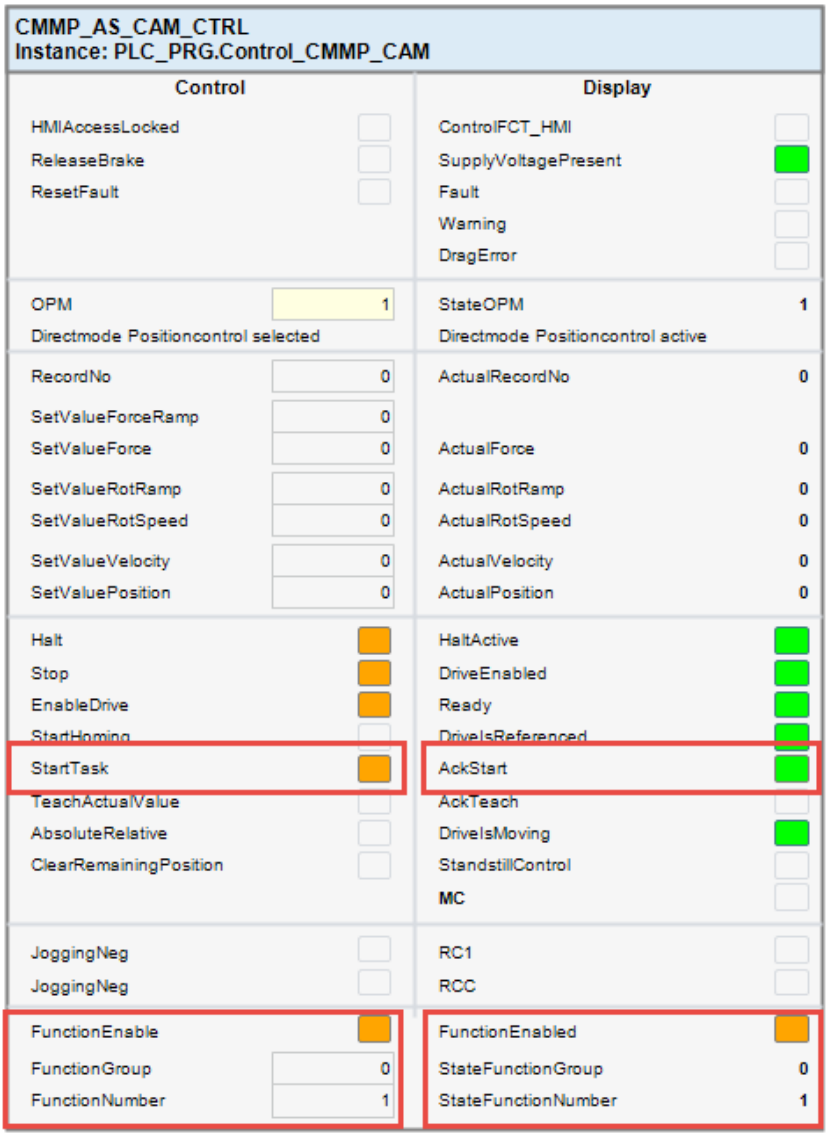


The slave can be synchronised with the help of the cam disc function.

“FunctionEnable” is activated to this end, “FunctionNumber” is set to 1 and “FunctionGroup” is set to 0. Synchronisation to an external input is active.

Synchronisation is activated in the cam disc function by applying a rising edge to “StartTask”.

As soon as the “AckStart” bit arrives, “StartTask” can be reset to 0.



The active “Synchronisation” function in the cam disc function can be monitored with the help of “FunctionEnabled”. In contrast to starting synchronisation via positioning records, for example, it’s not possible to query synchronous positioning via a digital output.

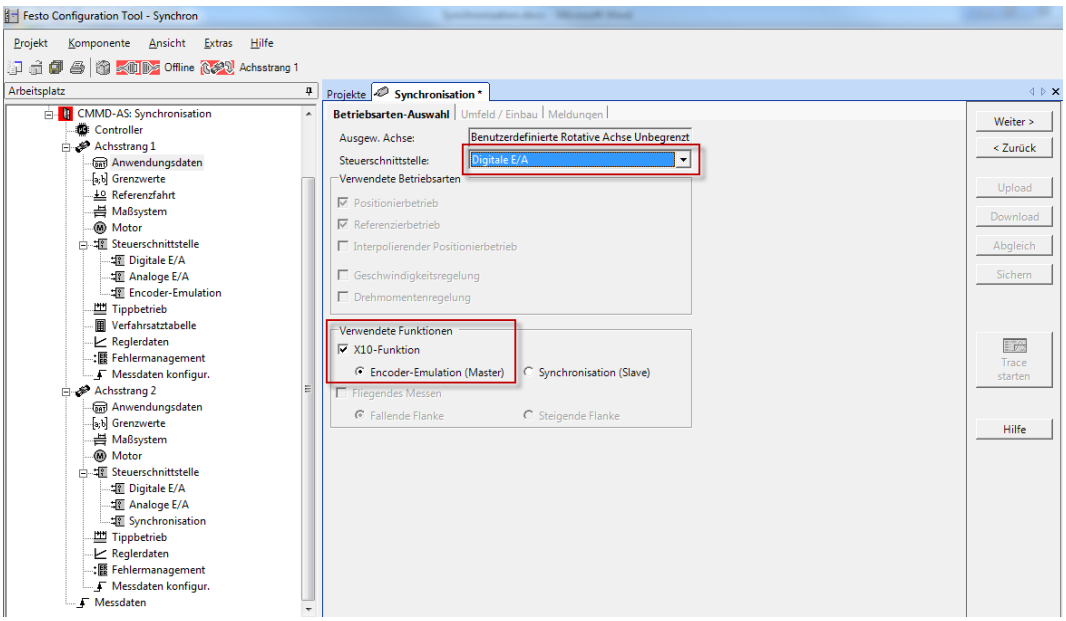
6 CMMS-x/CMMD-AS configuration

The fundamental procedure for configuring a controller is described in section 3.1 using the CMMP-AS as an example, and thus only those settings required for synchronisation are presented below.

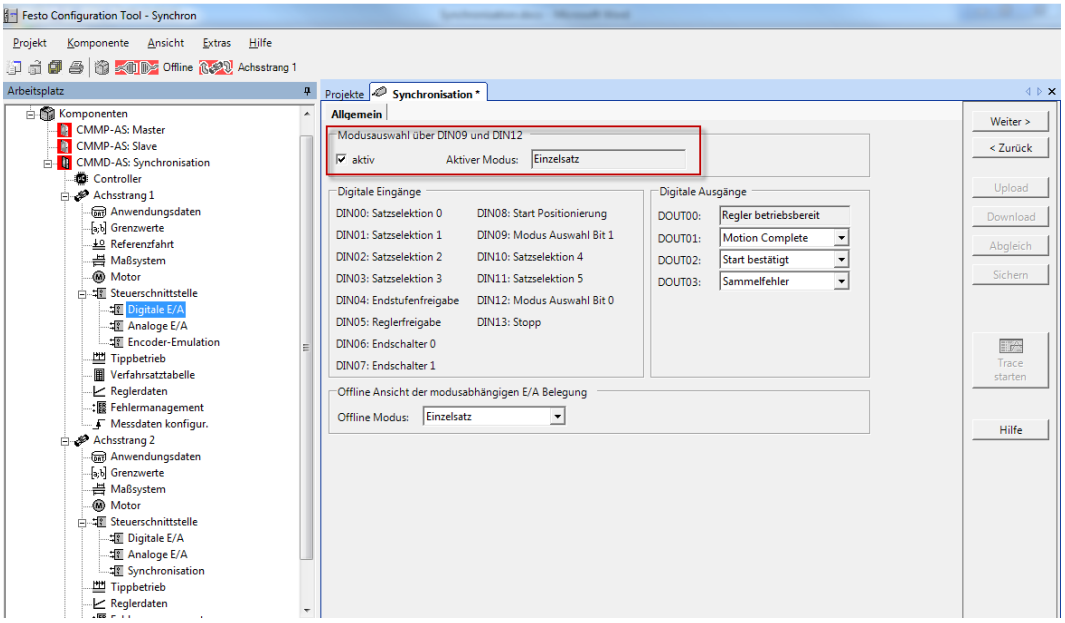
6.1 Configuring the master

In the example, the drive is controlled via the controller's digital inputs and outputs.

The "Digital I/O" control interface must be selected to this end, as well as the following additional function: X10 function -> Encoder emulation (master).



If further functions are required at the master in addition to the normal I/O mode (single record), for example jogging operation or linked positioning records, mode selection must be activated via DIN9 and DIN12.



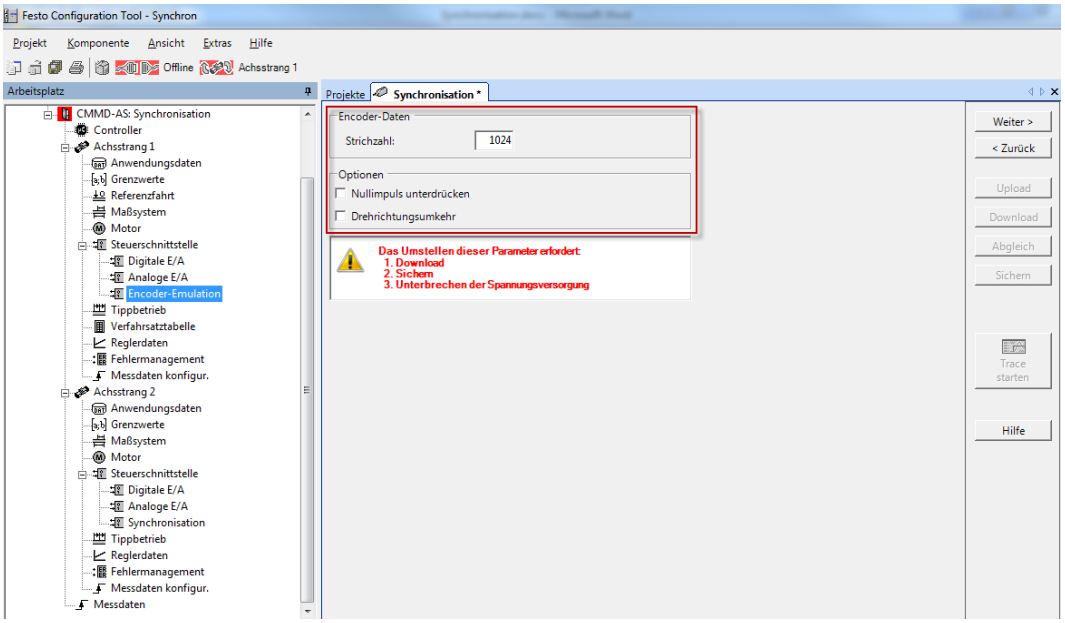
Settings for the synchronisation signals are entered via the "Encoder emulation" menu item.

In the case of the CMMS/CMMD controller, the synchronisation signal can only be read out by the master in the form of A/B signals via the X10 output.

The line count can be a value within a range of 32 to 2048, and is thus limited in comparison with the CMMP.

The zero pulse (see details in section 4.2.1) can also be suppressed and the direction of rotation can be reversed.

After changing the data, it must be assured that they are uploaded and saved to the controller, after which the controller has to be restarted before the changes become effective.

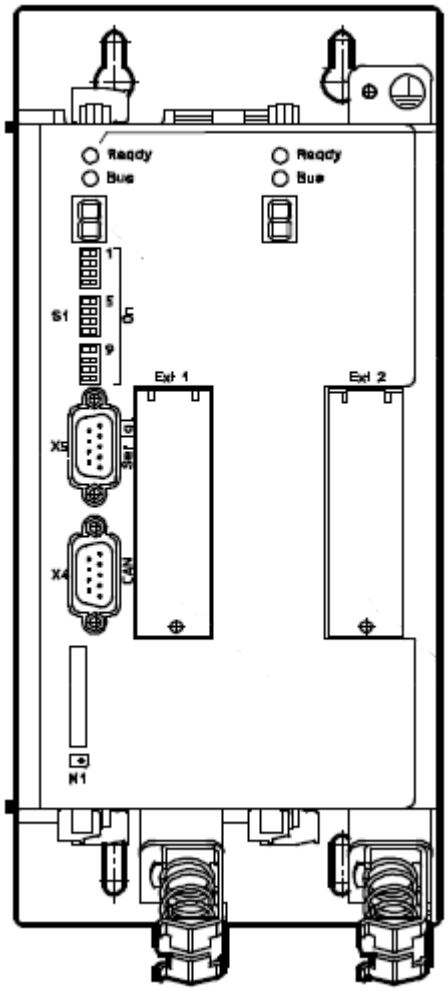


The line count is set to 1024 in our example and no options are selected.

6.1.1 Supplement – CAN bus / Profibus control interface

Peculiarities regarding controller configuration via CAN bus or Profibus are included in sections 3.1.1 and 3.1.2.
Differences between the CMMS/CMMD and the CMMP addressed here.

After one of the control interfaces has been selected, fieldbus settings can be entered.
The controller's address and the transmission rate are not set via FCT, but rather directly at the controller with the DIP switches.
In the case of the CAN bus, the baud rate must also be set and the CANopen interface must be activated.



DIL switch	Function
S1.1 ... 7	Bus address or MAC-ID on first axis ➔ Example Tab. 2.5 The second axis gets the address of the first axis +1 NodeNumberSlave = NodeNumberMaster +1
S1.8	Automatic loading of a new firmware file from the memory card by the start programme (bootloader): 1) – ON: Download from the SD memory card to the motor controller. – OFF: No download.
S1.9 ... 10	Setting the bus transmission rate
S1.11	Activation of the CAN-bus interface
S1.12	Terminating resistor for CAN-bus

Additional CAN

1) Additional information can be found in the firmware download ➔ Description of functions and commissioning, GDCP-CMMS/D-FW-....

Tab. 2.4 Function of the DIL switches

S1.1 ... 7	ON/OFF (example)	Significance 1)
1	ON 1	DIL switch S1.1 is the low-order bit.
2	ON 1	Example: address = **1011011** = 91
3	OFF 0	
4	ON 1	
5	ON 1	
6	OFF 0	
7	ON 1	

1) Additional information ➔ Description of functions and commissioning, GDCP-CMMS/D-FW-....

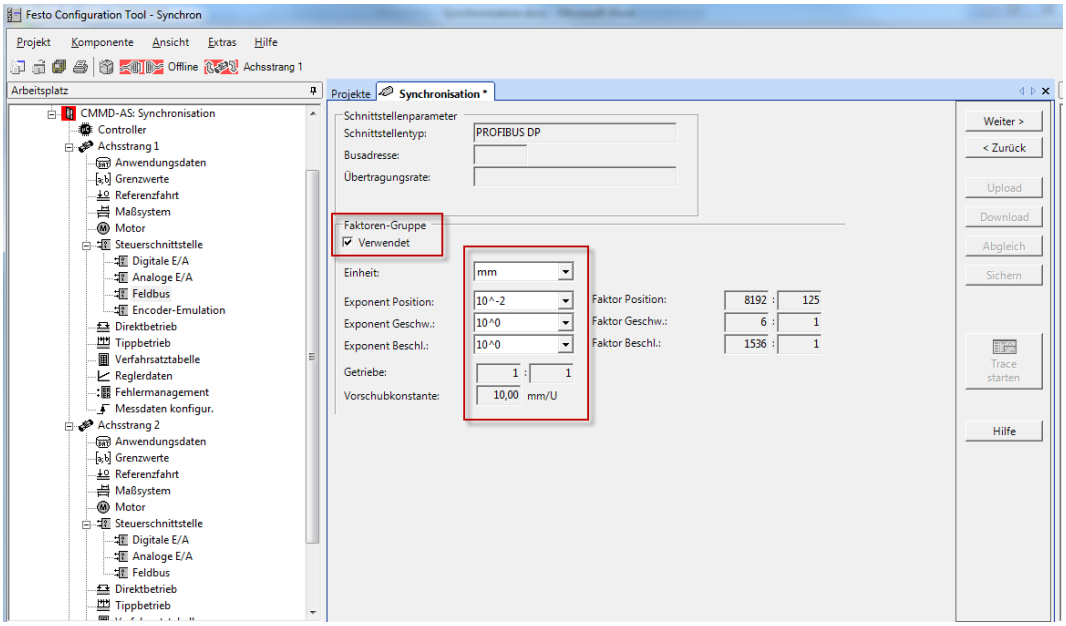
Tab. 2.5 CAN bus address or MAC-ID

S1.9 ... 10	ON/OFF (example)	Significance 1)
9	ON 1	DIL switch S1.9 is the low-order bit. 00: 125 kBit/s **01: 250 kBit/s (example)** 10: 500 kBit/s 11: 1000 kBit/s
10	OFF 0	

1) Additional information ➔ Description of functions and commissioning, GDCP-CMMS/D-FW-....

Tab. 2.6 CAN bus transmission rate

The factors group can be additionally activated in the “Fieldbus” window so that positions, speeds and acceleration rates can be displayed in user-defined units of measure.



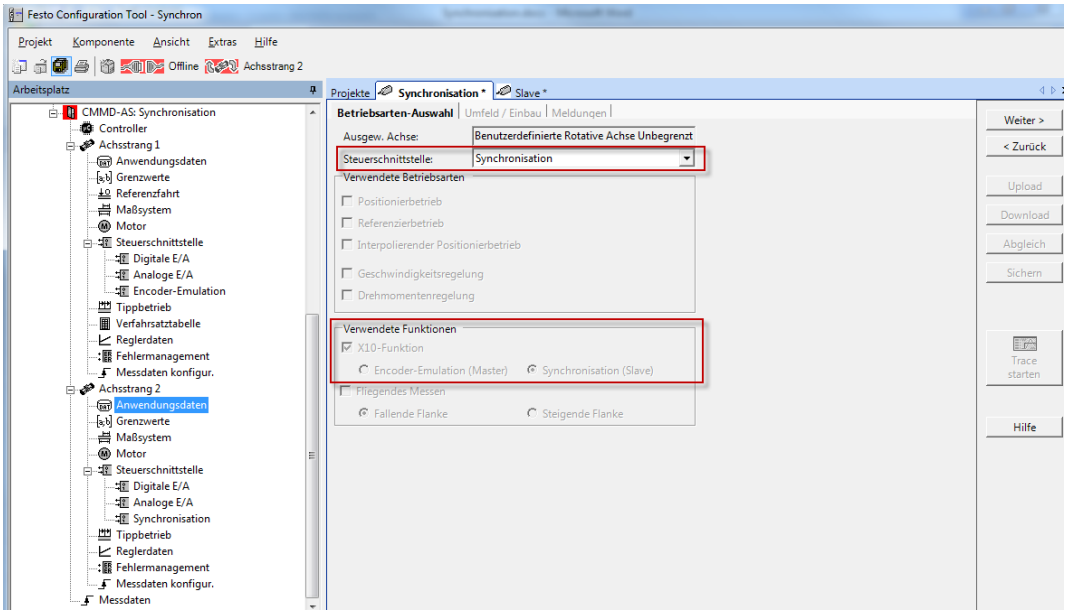
In contrast to the CMMP controller, the option for querying additional information cyclically via FHPP+ is not available in this case.

6.1.2 Supplement – analogue input as control interface

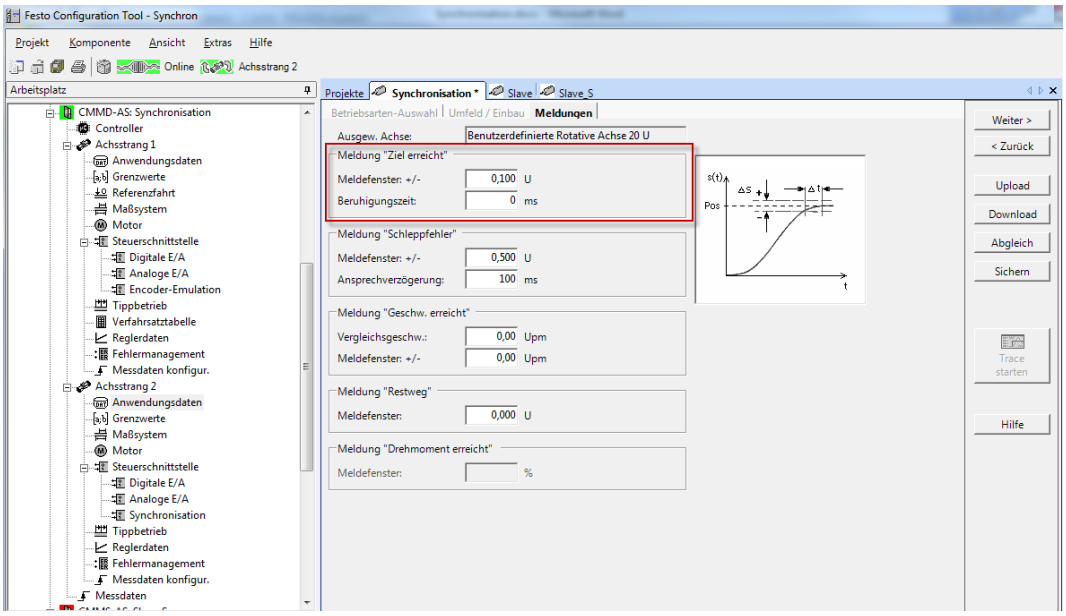
See section 3.1.4

6.2 Slave configuration – synchronisation control interface

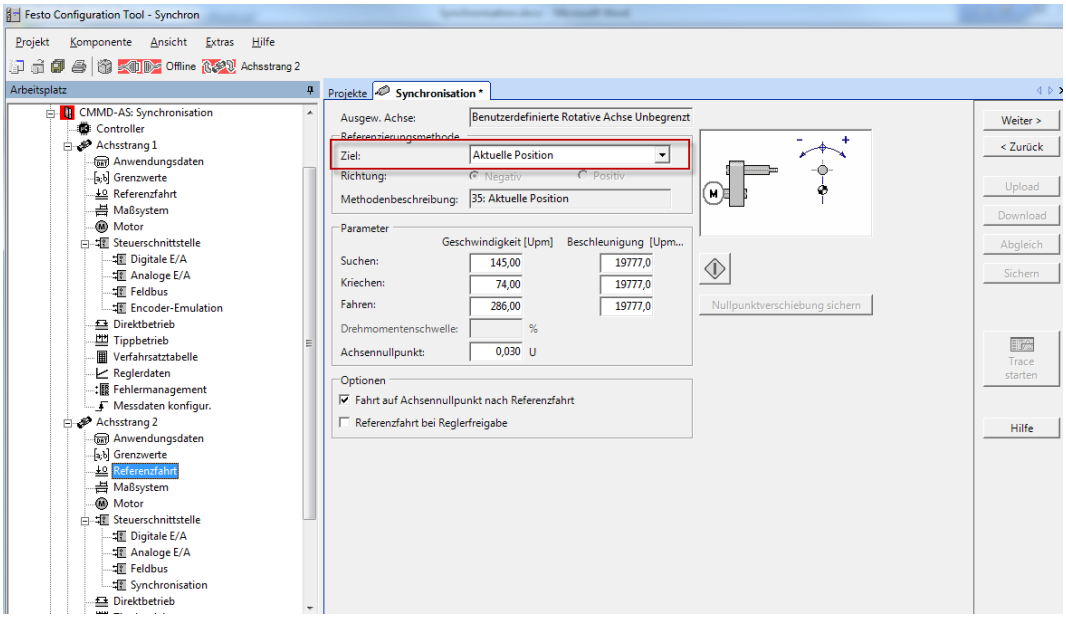
The “Synchronisation” control interface can be selected under application data.
“X10” and “Synchronisation (slave)” are selected automatically in this case.



Messages for certain events can be set up in the “Messages” tab.
The “Target reached” message is of particular interest, by means of which the window for the slave is set up, via which synchronous positioning relative to the master is queried.



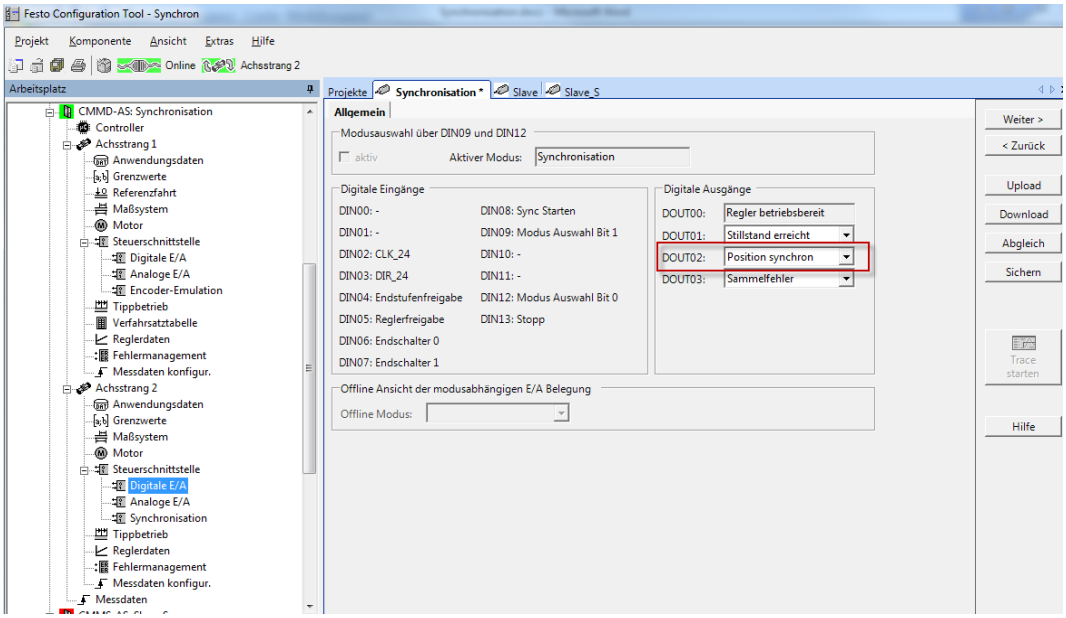
In the case of synchronisation with a master axis for identical travel, it’s advisable to set the homing method to “Current Position”. This makes subsequent alignment of the two axes to each other possible.



The corresponding outputs can be assigned in the digital I/Os window.

In this respect it makes good sense to assign an output to “Synchronous position”, in order to be able to determine whether or not the slave is synchronous to the master.

This output becomes active and is set to DOUT2 as a default value, as soon as the “Synchronisation” mode is selected.



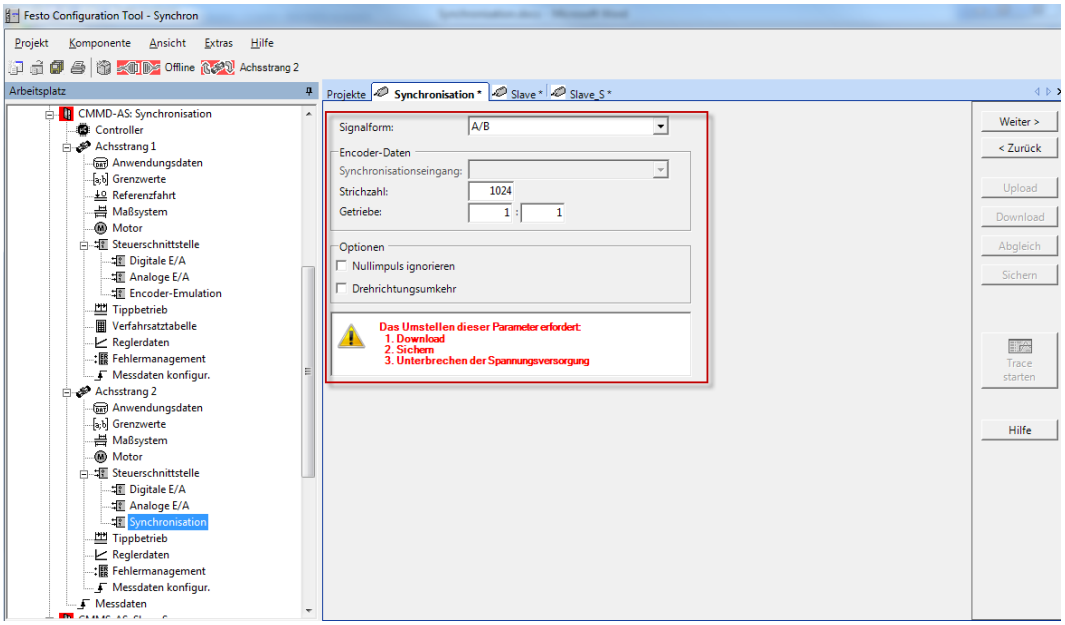
Settings for slave synchronisation can be entered under the “Synchronisation” menu item.

Selection can be made from a total of three signal types, and if type “CLK/DIR” or “CW/CCW” is selected, the synchronisation input must also be specified.

Furthermore, a line count and a gear ratio must also be entered, as well as zero pulse suppression (refer to section 4.2.1 for details) and rotary direction reversal as options.

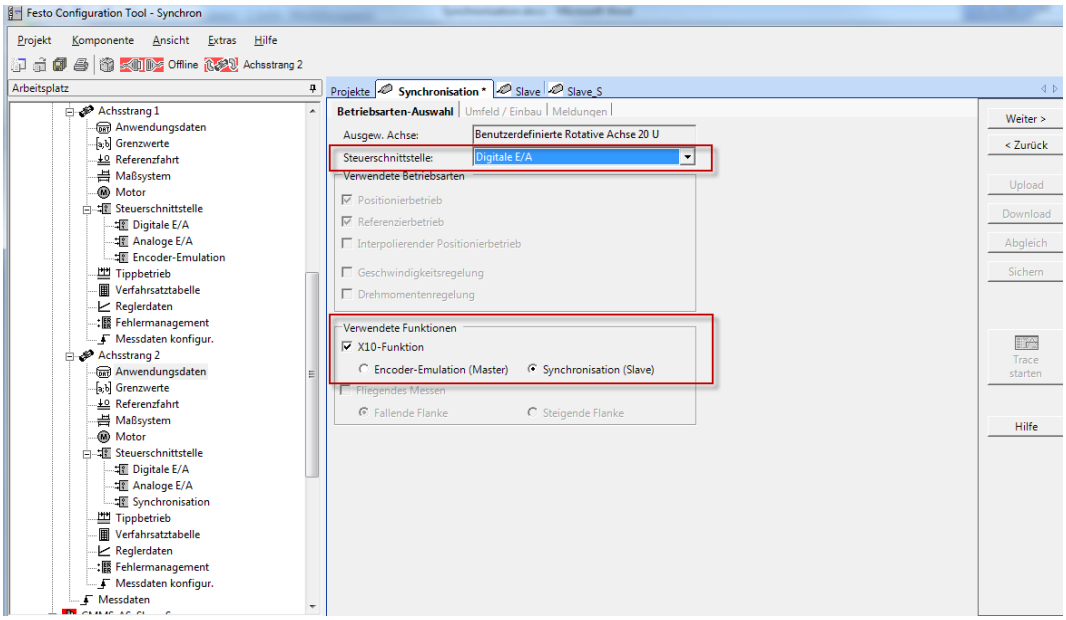
Due to the fact that a CMMD controller is used in the example (axis string 1 = master, axis string 2 = slave), and because two axes need to be run parallel to each other, the A/B track with a line count of 1024 and a gear ratio of 1:1 is used.

After changing the data they have to uploaded and saved to the controller, after which the controller has to be restarted.



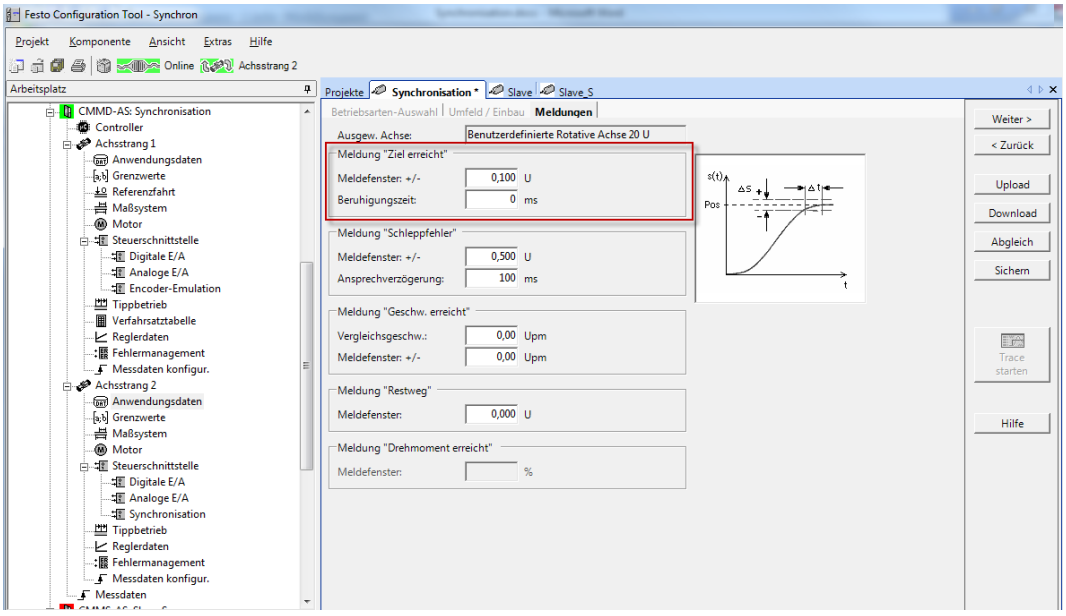
6.3 Slave configuration – I/O control interface

The control interface must be set to “Digital I/O” and the X10 function must be selected along with “Synchronisation (slave)”.

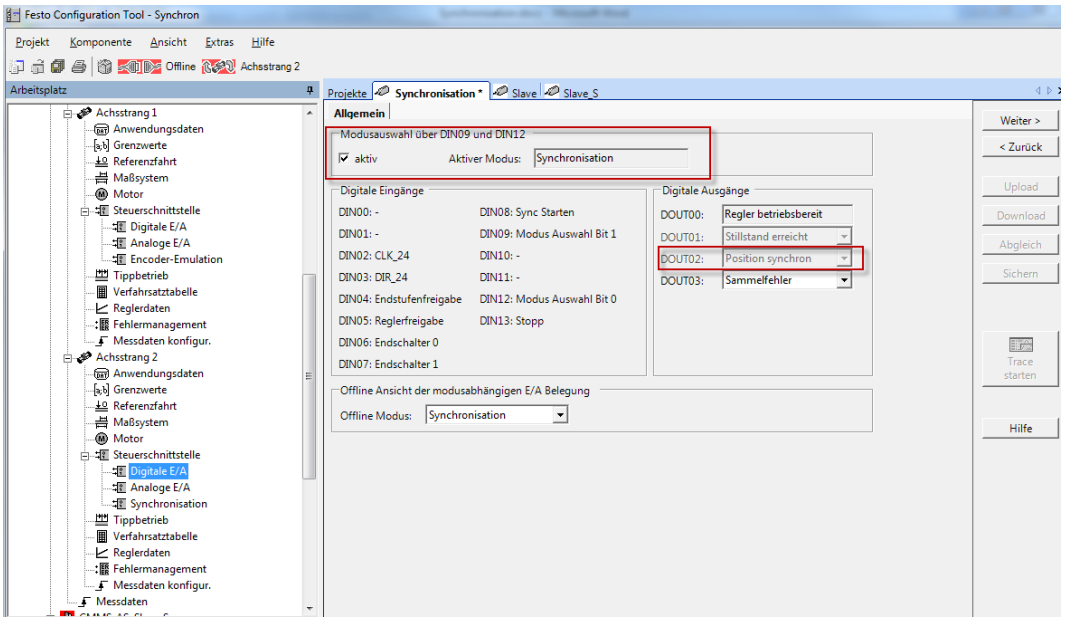


Messages for certain events can be set up in the “Messages” tab.

The “Target reached” message is of particular interest, by means of which the window for the slave is set up, via which synchronous positioning relative to the master is queried.

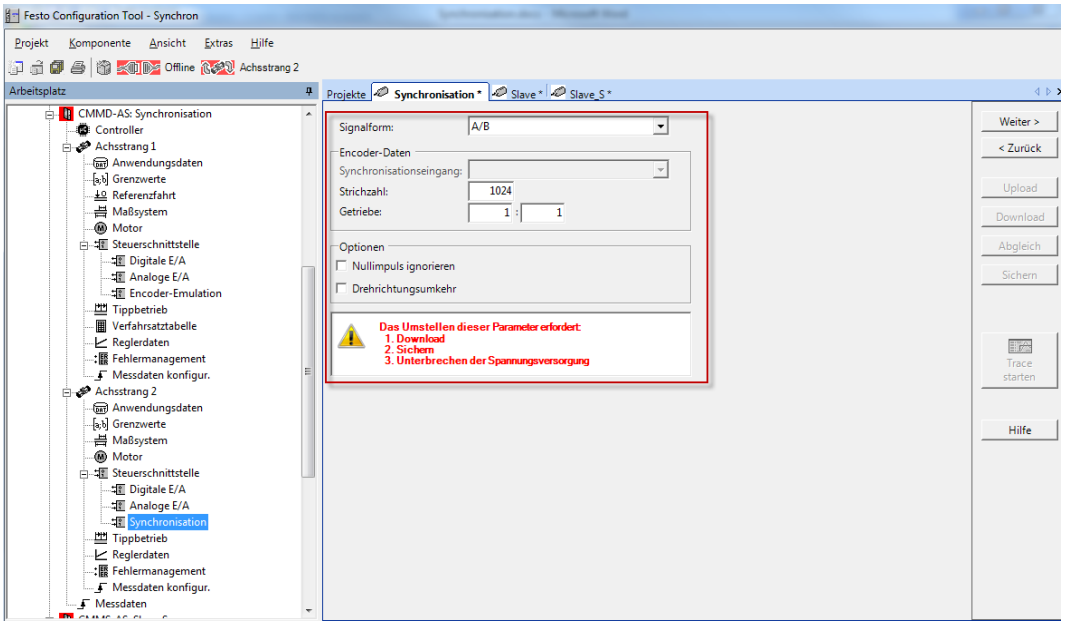


In order to be able to activate synchronisation via I/O, mode switching must be activated.
In the synchronisation mode, the signal for synchronous position is read out to DOUT2.
This cannot be changed.



Settings for slave synchronisation can be entered under the “Synchronisation” menu item.
Selection can be made from a total of three signal types, and if type “CLK/DIR” or “CW/CCW” is selected, the synchronisation input must also be specified.
Furthermore, a line count and a gear ratio must also be entered, as well as zero pulse suppression and rotary direction reversal as options.
Due to the fact that a CMMD controller is used in the example (axis string 1 = master, axis string 2 = slave), and because two axes need to be run parallel to each other, the A/B track with a line count of 1024 and a gear ratio of 1:1 is used.

After changing the data they have to uploaded and saved to the controller. The controller is then restarted.



Control of synchronisation via fixed positioning records, and thus the possibility of accurately synchronising a slave to a moving axis, is not available with the CMMS/CMMD.

6.4 Slave configuration – Profibus/CANopen control interface#

See section 5.1.1

7 Starting synchronization

7.1 Synchronisation via the “synchronisation” control interface at the CMMS/CMMD

Synchronisation of the slave is active immediately after the controller is restarted, i.e. the slave responds to each signal transmitted via the X10 interface.

Synchronisation can only be deactivated via the controller enable. In contrast to the CMMP, a digital input cannot be assigned.

No stop input is necessary, so that this switch-off path is eliminated as well.

Separate activation of synchronisation via mode switching (DIN 9 + Din 12) is unnecessary.

7.2 Synchronisation via the “I/O” control interface at the CMMS/CMMD

In order to activate synchronisation, mode switching must also take place via the digital inputs in addition to the usual enables (output stage enable, controller enable, stop input).

DIN 9 and DIN 12 are set to this end.

As soon as the start input (DIN 8) has also been set, the slave travels synchronous to the master.

In order to stop synchronisation, the start input must once again be deactivated.

Pin	Designation	Value	Mode = 3 – Synchronisation
1	AGND	0 V	Screen for analogue signals
2	DIN12		Mode switch slave synchronisation "1" = synchronisation
3	DIN10		
4	+VREF	+10 V ±4 %	Reference output for setpoint potentiometer
5	Free		
6	GND24		Reference potential for digital inputs and outputs
7	DIN1		Record selection 1 (high active)
8	DIN3	24 V	Direction_24 /CCW
9	DIN5		Controller release (high active)
10	DIN7		Limit switch 1
11	DIN9		Mode switch slave synchronisation "1" = synchronisation
12	DOUT1	24 V100 mA	Output freely programmable – default motion complete (high active)
13	DOUT3	24 V100 mA	Output freely programmable – default error (low active)
14	AGND	0 V	Reference potential for the analogue signals
15	DIN13		Stop input (low active)
16	DIN11		
17	AMON0	0 ... 10 V	Analogue monitor output 0
18	+ 24 V	24 V100 mA	24 V feed-in feed-out
19	DIN0		Record selection 0 (high active)
20	DIN2	24 V	Pulse_24 / CW
21	DIN4		Output stage enable (high active)
22	DIN6		Limit switch 0
23	DIN8		Start synchronization
24	DOUT0	24 V100 mA	Ready for operation output (high active)
25	DOUT2	24 V100 mA	Setpoint output reached (high active)

Table 6.5 Pin allocation: I/O interface [X1] mode 3

7.3 Synchronisation via bus at the CMMS/CMMD

The required enables must be set via the respective bus, in order to set the master and the slave to closed-loop control.

However, actual synchronisation cannot be started via the bus. Analogous to control via the I/O interface, it's started via the digital inputs.

Synchronous positioning of the slave can also only be queried via digital output DOUT2. The MC at the module does not provide any feedback in this regard. This is changed to logic 1 as soon as the slave is in position and synchronisation is switched back off.

The advantage of bus operation is monitoring of the respective positions in order to detect offset as early as possible and avoid damage to the mechanical system