Application on Drive Technology

applications
& TOOLS

SIEMENS

SIMATIC Easy Motion Control

Applikationsbeschreibung

Closed-Loop Controlled Positioning of an Axis with SIMATIC
CPU 314C-2 DP, MICROMASTER and SIMATIC Easy Motion Control

Warranty, Liability and Support

We accept no liability for information contained in this document.

We do not accept liability, whatever the legal basis, for any damages arising from the use of examples, notes, programs, configuration and performance data etc. described in this document, except where we are obliged to by the German Product Liability Act or in cases of willful damage or gross negligence, injury to life, body or health, breach of guarantee for the condition of products or items assumed by us, fraudulent concealment of a defect or breach of a substantial contractual obligation. However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above conditions are not meant to change the burden of proof to the detriment of the user.

The application examples do not purport to cover all details or variations in equipment, nor do they provide for every possible contingency. They are not customer-specific solutions. They are only intended to provide support for typical tasks. You are responsible for ensuring that the described products are used correctly. These application examples do not relieve you of the responsibility to use sound practices in application, installation, operation and maintenance. By using this application example you accept that Siemens is not liable for any damages except for those specified in the above liability clause. We reserve the right to make changes in these application examples at any time without prior notice. If there are any deviations between the recommendations provided in this application example and other Siemens publications – e.g. Catalogs – the contents of the other documents have priority.

For questions about this document please use the following e-mail address:

mailto:csweb@ad.siemens.de

Foreword

Structure of the document

The documentation for this application is divided into the following main parts.

Part	Description	Note
A1	Part A1 tells you everything needed to create an overview. You will find out about the components used (standard hardware and software components, as well as the additional software developed). The key function data shown demonstrates the performance capability of this application.	
A2	Part A2 looks at the detailed functional sequences of the hardware and software components involved. You only require this part if you wish to find out about the detailed sequence and the interaction between the solution components.	You can skip this part if you wish to test the application first of all using the step-by-step instructions
B	Part B is a step-by-step guide to setting up and commissioning the application.	This is the quick way to set the application into operation with the aid of the program files.
C	Part C is of interest to anyone wishing to extend/adapt their system on the basis of the existing software.	
D	Part D contains definitions of some basic terminology, as well as literature and Internet links.	

Note

You can find further, more detailed information and step-by-step configuration guides to

- assigning parameters to the MICROMASTER
- assigning parameters to the CPU 314C–2 DP
- assigning parameters to Easy Motion Control
- installing and assigning parameters to the SIMATIC NET OPC server

in the document entitled **Parameterizations**, to supplement Part B of this document.

Contents

Part A1: Application Description

Content of Part A1

Part A1 tells you everything needed to create an overview. You will find out about the components used (standard hardware and software components, as well as the additional software developed).

The key function data shown demonstrates the performance capability of this application.

Objective of Part A1

Part A1 of this document is meant to

- explain the automation task to the reader
- show him or her a possible solution
- demonstrate the performance capability of the overall application.

Topics covered:

1 Basic Information

1.1 Basic Information about the Application

Positioning application with an axis

The enclosed sample application is a closed-loop controlled single-axis positioning application. It focuses around the movement of **one** linear axis from position A to position B. The application is based on the SIMATIC CPU 314C-2 DP, Easy Motion Control V2.0 and a MICROMASTER 440 with an asynchronous motor.

It does not entail setpoint coupling and interpolation.

As the application also includes an HMI interface, it can also be used as a demonstration system. A storage lift is used as an example to demonstrate closed-loop controlled positioning with SIMATIC Easy Motion Control.

Prior knowledge

You need to have a basic understanding of motion control in order to follow the sample application.

For example, you should be familiar with terms like torque, inertia torque and open or closed-loop controlled positioning.

Note

You can find out basic information, about the design of the drive for example, in the **e-Infoshop** on "Simple Positioning".

Structure of the application

The application comprises an **application description** with an explanation of the automation task and of the solution, including hardware and software components from the SIMATIC and drive range.

The **wiring plans** and the **list of components** enable the underlying test setup to be recreated. Loading the **application software** forms the basis for moving the axis (see Figure 1-1).

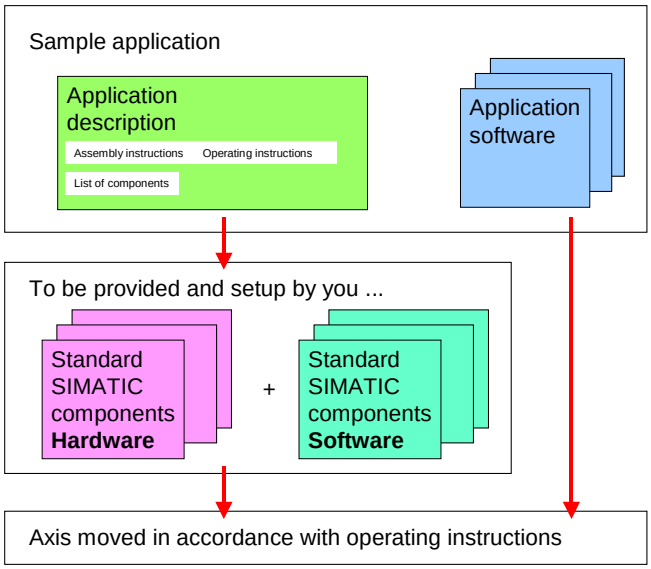


Figure 1-1-1

Objective of the sample application

This application is intended to demonstrate the potential of cost-efficient single-axis positioning with the aid of the SIMATIC Easy Motion Control software package.

A storage lift with an HMI interface is automated in order to demonstrate this as simply and practically as possible. This means that the application is also used as a demonstration model.

The application should serve to illustrate:

- how the components that are used interact
- how they are programmed or parameters are assigned to them
- how measured values are transmitted with the aid of the OPC server, imported into EXCEL, displayed graphically and analyzed
- which alternative configurations are possible and what the advantages and disadvantages of these are compared to the selected configuration
- what degrees of positioning accuracy and consistency can be achieved under real conditions. For this purpose, the power data has been determined by means of a load torque simulation.

The chosen automation task entailed charging and discharging boxes of materials using a storage lift comprising 25 storage shelves.

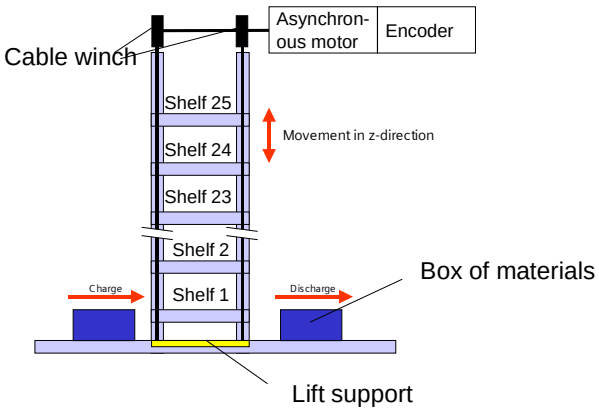


Figure 1-2 Storage lift

The solution for the automation task has been created with the following components and has been described in detail:

- CPU 314C-2DP
- SIMATIC Easy Motion Control software package
- MICROMASTER 440
- Standard asynchronous motor
- Incremental encoder
- User interface in ProTool/Pro RT

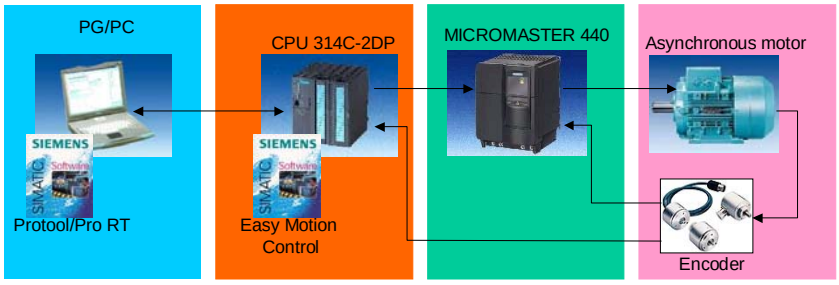


Figure 1-3 Components used

It should give an insight into the positioning accuracy and consistency that can be achieved under real conditions. For this purpose, the power data has been determined by means of a load torque simulation.

1.2 Safety Instructions

Risk of injury

Warning

The hardware components used as components in plant and systems require compliance with special rules and regulations, depending on the area of application.

Please comply with prevailing safety and accident-prevention regulations, e.g. IEC 204 (emergency STOP devices).

Failure to comply with these regulations may lead to serious injury and to damage to machinery and equipment.

Danger

Risk of injury caused by moving parts.

Danger

You may come into contact with live cables. Therefore, you MUST isolate the power before wiring the application setup.

2 Automation Task

Positioning with SIMATIC Easy Motion Control will be demonstrated using a storage lift as an example.

From a technological perspective, this entails closed-loop single-axis positioning control with 25 target positions on a linear axis.

A lift support is moved up and down with the aid of cable winches. The cable winches are located on an axis that is connected to the drive.

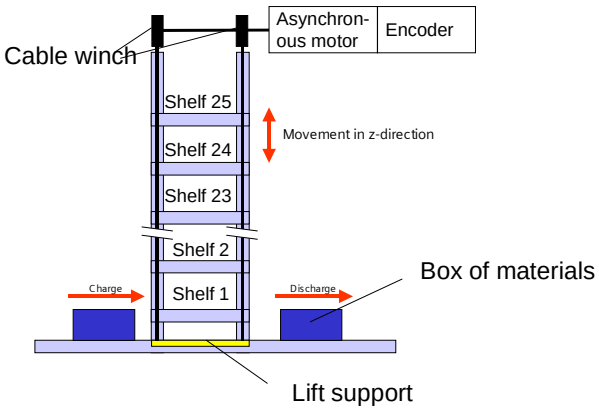


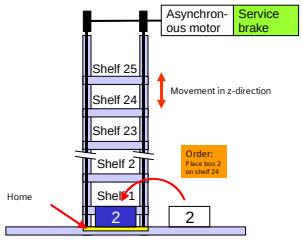
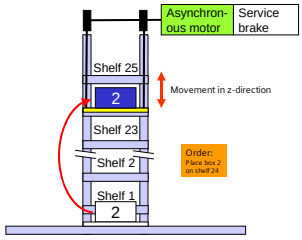
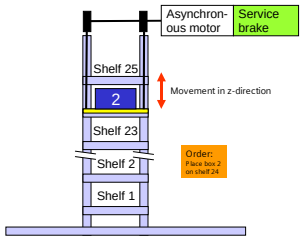
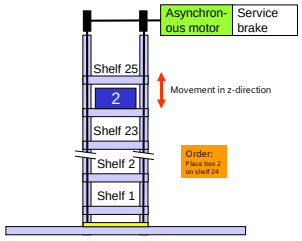
Figure 2-1 Diagram of the storage lift

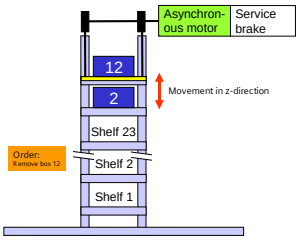
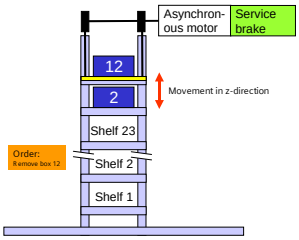
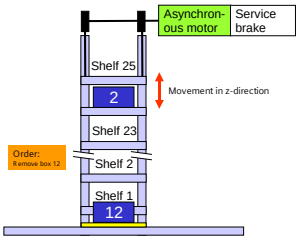
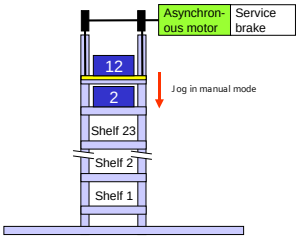
Features to be implemented:

- The complete controls of the store will be performed via the HMI application ProTool/Pro.
- The storage lift can be moved in two operating modes:
  - **Manual operation** with the following functions:
    - Jogging, i.e. moving the axis using a jogging button
    - Search for reference, i.e. synchronizing the axis with the aid of the reference point switch
    - Positioning to a manually pre-defined position.
  - **Automatic operation** with the following functions:
    - Automatic positioning to the selected shelf
    - Managing a store DB which reflects the store and also contains the 25 shelf positions.
- If no charge or discharge job appears for a certain length of time and the lift is not in the basement position, it is automatically moved to idle position.
- While the lift is on its way to its idle position, its journey may be interrupted if a charge or discharge job is entered, causing the lift to move to the shelf where it is needed without any undue time loss. This is done with the support of SIMATIC Easy Motion Control's technological dynamic positioning order replacement function, i.e. a positioning job is interrupted and replaced by another job without causing the drive to come to a standstill.
- The placing and removal of boxes from the lift, as well as the transfer between the lift and the shelves should only be simulated with the aid of the HMI.
- The actual speed and the setpoint speed of the drive can be measured and can be imported into EXCEL to be displayed graphically and evaluated.

Technical sequence: Charge and discharge of a box of materials

Table 2-1 Storage lift

Level	Function	Comment
Charging a box of materials		
1		Once the lift support is free and in its home position, the box of materials is placed on the lift support. To enable the materials to be tracked, the box is assigned a unique number (e.g. 2), and the target shelf (e.g. 24) is selected.
2		If shelf 24 is free, the service brake is released and the box of materials is transported up to shelf 24 with the aid of the lift support.
3		Once the lift support reaches the target position, the service brake is activated. Once the brake is activated, the box of materials is pushed onto the shelf.
4		Once the box of materials has been deposited onto its target shelf, the service brake is released, and the lift support moves to its home position. The service brake is activated again once the lift support has reached the home position. If a discharge order is entered during the journey to the home position, the positioning job is interrupted and replaced by a new job.

Level	Function	Comment
Discharging a box of materials		
5		If no charge or discharge order is currently being processed, the lift support moves to the requisite shelf.
6		Once the positioning job has been completed, the service brake is activated, and the box of materials is conveyed onto the lift support.
7		Once the box of materials is on the lift support, the service brake is released, and the lift support moves to its home position A new order cannot be processed until the box of materials is removed.
Malfunction		
8		In the event of a malfunction, the pending positioning order is cancelled. The box of materials is then moved to the home position, along with the lift plate, manually.

Technical dimensions of the high-bay store

The physical requirements, such as height, weight of the boxes of materials and positioning speed have been adapted to the components used in the sample application.

Table 2-2 Technical data for the high-bay store

Components	Dimensions
Dimensions of the box	Height: 30 cm Width: 50 cm Depth: 40 cm
Weight	Lift support excl. box: 9 kg Box: 4 kg
Cable winch incl. motor:	Diameter: 15 cm Inertia torque: 0.0076 kgm^2 (Relative to the motor spindle) Max. acceleration: 0.45 m/s^2 Transmission ratio: 1:10
Positions of the shelves	Position of shelf 1: 50 cm Position of shelf 2: 100 cm Position of shelf 23: 1150 cm Position of shelf 24: 1200 cm Position of shelf 25: 1250 cm

3 Automation Solution

Taking the automation solution as a basis, there are several configurations which can be used to solve the task.
The solution which is applied and two alternative solutions are described and compared below.

SIMATIC Easy Motion Control represents the closed-loop position control, as illustrated in the following diagram, in all three solutions.

Automation setup

The control structure, upon which the automation setup is based, comprising closed-loop positioning control with underlying closed-loop speed control, is identical in all the solution concepts.

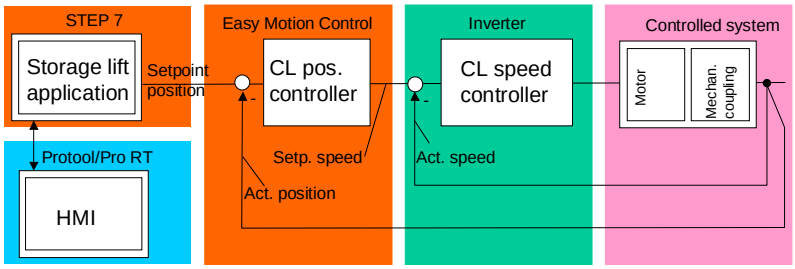


Figure 3-1 Diagram of the automation setup

Overview of the solutions

Table 3-1 Overview of the solutions

Comparison criteria	**Applied solution** **Analog signal + MICROMASTER**	**Alternative solution 1** **PROFIBUS + MICROMASTER**	**Alternative solution 2** **MASTERDRIVE + servomotor**
Controller	CPU 314C-2DP	CPU315-2DP	CPU314C-2DP
Positioning takes place in	Digital / analog	PROFIBUS	Digital / analog
Inverter	MICROMASTER 440	MICROMASTER 440	MASTERDRIVE MC
Motor	Standard asynchronous motor	Standard asynchronous motor	Synchronous servomotor
Encoder	Incremental encoder	Incremental encoder	-

Table 3-2

Comparison criteria	**Applied solution** **Analog signal + MICROMASTER**	**Alternative solution 1** **PROFIBUS + MICROMASTER**	**Alternative solution 2** **MASTERDRIVE + servomotor**
Precision	Increment-precise	Increment-precise	Increment-precise
Dynamics	Suitable for positioning jobs over a long distance	Suitable for positioning jobs over a long distance	Suitable for positioning jobs over a long distance and for frequent short positioning jobs
Hardware costs	Drive costs are low	Drive costs are low	Drive costs are higher
Hardware cost comparison	100%	110%	150%
Engineering workload	Basic knowledge	Basic knowledge	Specialist knowledge
Control during standstill	Requires a holding brake or separately driven fan	Requires a holding brake or separately driven fan	No restriction
Power range	120W – 75 kW	120W – 75 kW	550W – 2300kW
Immunity to interference	Analog signal may be falsified or may drift	High immunity to interference due to transmission with PROFIBUS	Analog signal may be falsified or may drift
Setup	Centralized	Decentralized	Centralized

3.1 Applied Solution: Central Controller with a CPU314C + MICROMASTER 440 + an Asynchronous Motor

Overview

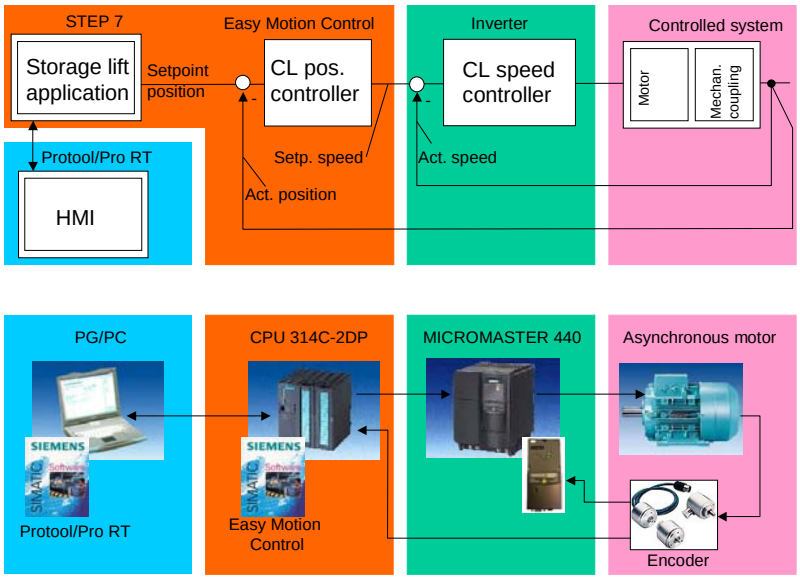


Figure 3-2 Overview

Functional description

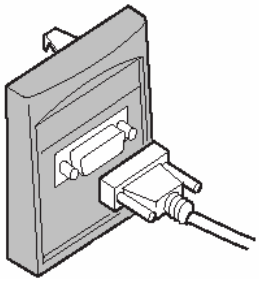
The charge or discharge tasks entered via the HMI are processed by the storage lift program which has been created for this application and are passed on to the position controller as positioning setpoint.
The closed-loop positioning control takes place in the CPU314C by means of the Easy Motion function blocks. An analog output with ± 10V is used for communication with the MICROMASTER, which performs the speed control and activates the motor. The speed and the position are identified by the MICROMASTER and by the CPU314C.

Hardware components

The following hardware components are required to create the application. (June 2003 prices in Euros)

Table 3-3 Hardware Components

Component	Picture	MLFB / Order number Function	L-price
Controller			
DIN rail		6ES7390-1AE80-0AA0 The DIN rail is the mechanical mounting rack for an S7-300 and is required to set up the controller.	25.70
PS307 2A power supply unit		6ES7-307-1BA00-0AA0 The power supply unit provides the requisite 24V DC supply.	89.00
CPU 314C-2DP		6ES7314-6CF00-0AB0 The CPU 314C-2 DP calculates the user program. The encoder is read in and the drive activated via the integrated I/Os.	1360.00
392 front-panel connector with screw contacts 40-pin		6ES7392-1AM00-1AA0 The front-panel connector provides a simple and user-friendly facility for connecting the sensors and actuators to signal modules. It is plugged into the module and concealed by the front door.	2 @ 31.60
Micro memory card 64 KB		6ES7953-8FL00-0AA0 The S7-program is stored on the MMC.	35.00

Component	Picture	MLFB / Order number Function	L-price
Drive			
MICROMASTER MM 440		6SE6440-2UC12-5AA1 The MICROMASTER supplies the current for the motor and works in speed control.	237.00
Pulse encoder module		6SE6400-0EN00-0AA0 The pulse encoder module evaluates the encoder and supplies the actual speed values for speed control.	150.00
PC-inverter connection kit		6SE6400-1PC00-0AA0 The PC-inverter connection kit supports the simple assignment of parameters for the MICROMASTER via an RS232 interface with the **Starter** software.	30.00
BOP (optional)		6SE6400-0AP00-0AA0 The MICROMASTER's Basic Operator Panel is used for the purposes of operation display and for parameter assignment.	35.00
AOP (optional)		6SE6400-0AP00-0AA1 The MICROMASTER's Advanced Operator Panel with its plain text display is used for the purposes of operation display and for parameter assignment.	135.00

Component	Picture	MLFB / Order number Function	L-price
Brake resistor		6SE6400-4BC05-0AA0 The brake resistor enables the MICROMASTER to be actively braked.	122.00
Network filter		6SE6400-2FL01-0AB0 The network filter attenuates the system in-rush from the MICROMASTER.	120.00
PROFIBUS interface (optional)		6SE6400-1PB00-0AA0 The PROFIBUS interface supports communication with the MICROMASTER via the PROFIBUS. Parameters can also be assigned with the Starter software from the DriveES package.	150.00
Motor			
Standard 180W asynchronous motor		1LA7063-4AB10-Z Options: A23, G26 The motor drives the axis.	169.00 385.80
Incremental encoder		6FX2001-4SB00 The encoder records the speed and position of the drive.	156.00
Coupling for attaching the encoder		e.g. 6FX2001-7KS10 The coupling connects the encoder to the axis and/or the motor.	29.00
HMI			
PG/PC with an MPI interface		- The PG/PC is used to process the HMI operator interface and to display the measured values in EXCEL	-

Component	Picture	MLFB / Order number Function	L-price
Costs			
Hardware costs (excl. options)			2,970.90

Software components

The following software components are required to create the application. (June 2003 prices in EUROS)

Component	MLFB / Order number Function	L-price
STEP7 V5.2	6ES7810-4CC06-0YX0 STEP 7 is the basic package for the other software packages and is used for programming the S7	1,631.00
SIMATIC Easy Motion Control V2.0	6ES7864-0AC01-0YX0 Easy Motion Control	400.00
Starter (stand-alone) V3.x (optional)	On the Internet (see annex) Starter is a parameterization tool for the MICROMASTER. It uses the serial interface. It cannot be installed together with DriveES.	freeware
DriveES Basic V5.2 (optional)	6SW1700-5JA00-2AA0 DriveES is a parameterization tool for Siemens drives. It integrates them into STEP 7 and contains the "Starter". It uses the serial interface and/or PROFIBUS. It cannot be installed together with the Starter (stand-alone).	332.34
ProTool/Pro 6.0 + SP2 CS (optional)	6AV6582-2BX06-0CX0 ProTool/Pro is used for programming the HMI user interface. You cannot make any changes to the HMI user interface without this software.	1,675.00
ProTool/Pro 6.0 + SP2 RT128	6AV6584-1AB06-0CX0 ProTool/Pro RT enables a PG/PC to be used as an operator panel.	705.00
SIMATIC NET V6.x (optional)	6GK1704-5CW60-3AA0 SIMATIC NET contains an OPC server which supports access to the controller for Windows programs. The measured values cannot be imported into EXCEL without this software.	450.00

Component	MLFB / Order number Function	L-price
Microsoft EXCEL 2000 (optional)	- EXCEL is used for the graphical display and analysis of the measured values.	-
Software costs (excl. options)		2,736.00

Application software

Table 3-4 Application software

User program	Function	Source
User program or application program	Storage management and positioning of the lift axis on the basis of Easy Motion Control	MC_EMC.zip
ProTool/Pro	HMI user interface	MC_EMC.zip
MICROMASTER configuration data (for DriveES)	MICROMASTER configuration data in Vector Control mode	MC_EMC.zip
MICROMASTER configuration data (for Starter)	MICROMASTER configuration data in Vector Control mode	MM440_Starter.zip
EXCEL table	To record the current position and the actuating signal for the actual position	MeasureData.xls

Also refer to 6.3.1 Dearchive the STEP 7 Project.

Savings potential resulting from the use of SLVC

Speed control in the inverter is necessary for positioning. The MM 440 is the only model in the MICROMASTER series to offer this feature.

The MICROMASTER 440 can also be operated in SLVC (Sensor Less Vector Control) mode without an external encoder, saving the encoder modules.
The current actual speed of the motor is calculated with the aid of the motor data that is input, the values measured during the motor identification process and the present current and voltage values. This calculated value is then used for speed control.

However, as this calculation is imprecise at low speeds (under about 1 Hz), control in this range is generally worse than with an encoder.

As far as positioning is concerned, this means that, as the system moves towards the target position, deviations may occur which also have to be

compensated for by the position controller. This causes the accuracy of the positioning to fall.

As the costs for the MICROMASTER encoder module only amount to approx. 5% of the total hardware costs, we would advise you to use the module.

Performance at higher motor power ratings

The motor power capacity is initially of no significance to closed-loop positioning control. The size of the drive has no significant bearing on positioning performance while the drive follows the setpoint speed.

However, as the inertia torque rises more quickly than the maximum motor torque, the drive performs more slowly at higher power ratings, and positioning is less accurate and takes longer.

This effect depends on the ratio between the inertia of the motor and the load .

Key performance data, see Section 4 Key Performance Data, has been ascertained using a motor with a power rating of 120W. Consequently, we would advise anyone using higher motor power ratings to discuss the application and the dimensions with a drive specialist.

3.2 Alternative Solution 1: Decentral Controller with a CPU 315-2DP + MICROMASTER 440 + an Asynchronous Motor

Overview

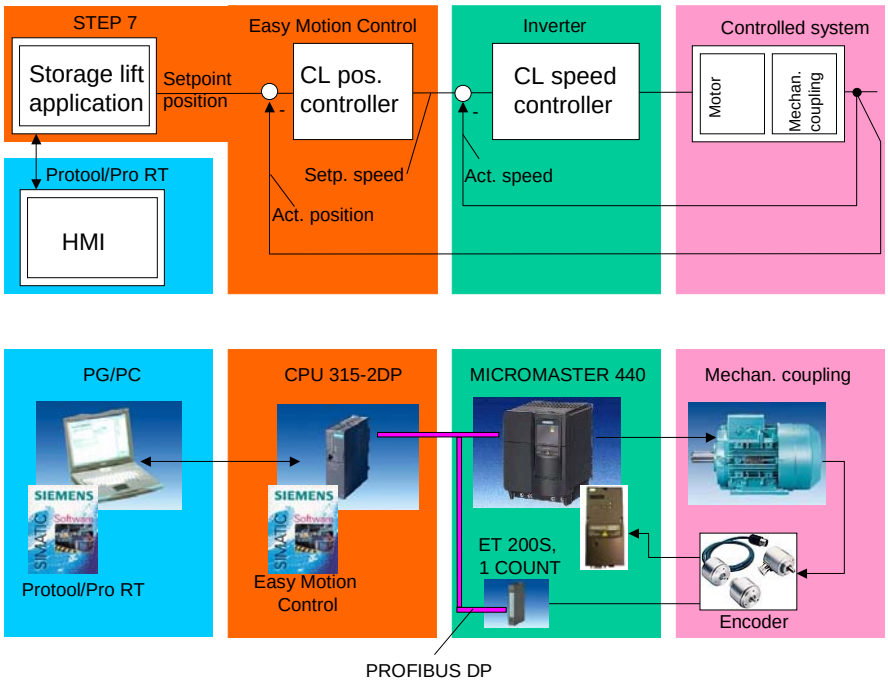


Figure 3-3 Overview

Functional description

The charge or discharge tasks entered via the HMI are processed by the storage lift program which has been created for this application and are passed on to the position controller as positioning setpoint.
The closed-loop positioning control takes place in the CPU315 by means of the Easy Motion function blocks. The PROFIBUS is used for communication with the MICROMASTER. It performs the speed control and activates the motor. The speed and the position are identified by the encoder and are identified by the MICROMASTER and by an ET200 module for the S7-CPU.

Compared to the applied solution, this alternative solution is decentrally structured. This allows the MICROMASTER and the counter module to be mounted in situ on the motor, instead of in a central electronics box. The transmission of the setpoint via the PROFIBUS reduces the risk of parasitic interference.

Hardware components

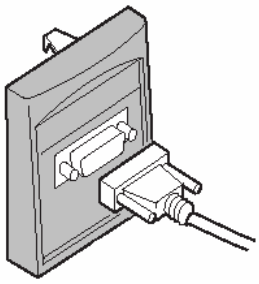
The following hardware components are required to create the application. (June 2003 prices in EUROS)

Table 3-5 Hardware Components

Component	Picture	MLFB / Order number Function	L-price
Controller			
DIN rail		6ES7390-1AE80-0AA0 The DIN rail is the mechanical mounting rack for an S7-300 and is required to set up the controller.	25.70
PS307 2A power supply unit		6ES7-307-1BA00-0AA0 The power supply unit provides the requisite 24V DC supply.	89.00
CPU 315-2DP		6ES7315-2AG10-0AB0 The CPU 315-2 DP calculates the storage lift automation incl. the Easy Motion Control blocks.	1250.00
Micro Memory Card 64KB		6ES7953-8FL00-0AA0 The S7-program is stored on the MMC.	35.00
IM151-1 standard interface module		6ES7151-1AA02-0AB0 The IM 151 interface module enables the ET 200S to be connected to the PROFIBUS DP and independently processes communication between the modules and the higher-level DP master.	220.00

Component	Picture	MLFB / Order number Function	L-price
TM-P15S22-01 terminal module		6ES7193-4CE00-0AA0 Terminal modules are purely mechanical components for setting up the ET 200S. They hold the electronic modules.	6.10
TM-E15S24-A1 terminal module		6ES7193-4CA20-0AA0 Terminal modules are purely mechanical components for setting up the ET 200S. They hold the electronic modules.	32.90 (5 pcs)
PM-E power module		6ES7138-4CA00-0AA0 PM-E power modules are used to monitor load and encoder supply voltages that are fed via the TM-P terminal modules to the self-generating potential rails.	11.60
ET 200S module 1 COUNT 24 V/100 kHz		6ES7138-4DA03-0AB0 The ET 200S module 1 COUNT 24 V/100 kHz is used for speed and position identification.	175.00

Component	Picture	MLFB / Order number Function	L-price
Drive			
Micromaster MM 440		6SE6440-2UC12-5AA1 The MICROMASTER supplies the current for the motor and works in speed control.	237.00
PROFIBUS interface		6SE6400-1PB00-0AA0 The PROFIBUS interface supports communication with the MICROMASTER via the PROFIBUS. Parameters can also be assigned with the Starter software from the DriveES package.	150.00

Component	Picture	MLFB / Order number Function	L-price
PC-inverter connection kit (optional)		6SE6400-1PC00-0AA0 The PC-inverter connection kit supports the simple assignment of parameters for the MICROMASTER via an RS232 interface with the **Starter** software.	30.00
BOP (optional)		6SE6400-0AP00-0AA0 The MICROMASTER's Basic Operator Panel is used for the purposes of operation display and for parameter assignment.	35.00
AOP (optional)		6SE6400-0AP00-0AA1 The MICROMASTER's Advanced Operator Panel with its plain text display is used for the purposes of operation display and for parameter assignment.	135.00
Brake resistor		6SE6400-4BC05-0AA0 The brake resistor enables the MICROMASTER to be actively braked.	122.00
Network filter		6SE6400-2FL01-0AB0 The network filter attenuates the system in-rush from the MICROMASTER.	120.00
Motor			
Standard 180W asynchronous motor		1LA7063-4AB10-Z Options: A23, G26 The motor drives the axis.	169.00 385.80
Incremental encoder		6FX2001-4SB00 The encoder records the speed and position of the drive.	156.00

Component	Picture	MLFB / Order number Function	L-price
Coupling for attaching the encoder		e.g. 6FX2001-7KS10 The coupling connects the encoder to the axis and/or the motor.	29.00
HMI			
PG/PC with an MPI interface		- The PG/PC is used to process the HMI operator interface and to display the measured values in EXCEL	-
Costs			
Hardware costs (excl. options)			3213.30

Software components

The following software components are required to create the application. (June 2003 prices in EUROS)

Component	MLFB / Order number Function	L-price
STEP7 V5.2	6ES7810-4CC06-0YX0 STEP 7 is the basic package for the other software packages and is used for programming the S7	1,631.00
SIMATIC Easy Motion Control V2.0	6ES7864-0AC01-0YX0 Easy Motion Control	400.00
Starter (stand-alone) V3.x (optional)	On the Internet (see annex) Starter is a parameterization tool for the MICROMASTER. It uses the serial interface. It cannot be installed together with DriveES.	freeware
DriveES Basic V5.2 (optional)	6SW1700-5JA00-2AA0 DriveES is a parameterization tool for Siemens drives. It integrates them into STEP 7 and contains the "Starter". It uses the serial interface and/or PROFIBUS. It cannot be installed together with the Starter (stand-alone).	332.34

Component	MLFB / Order number Function	L-price
ProTool/Pro 6.0 + SP2 CS (optional)	6AV6582-2BX06-0CX0 ProTool/Pro is used for programming the HMI user interface. You cannot make any changes to the HMI user interface without this software.	1,675.00
ProTool/Pro 6.0 + SP2 RT128	6AV6584-1AB06-0CX0 ProTool/Pro RT enables a PG/PC to be used as an operator panel.	705.00
SIMATIC NET V6.x (optional)	6GK1704-5CW60-3AA0 SIMATIC NET contains an OPC server which supports access to the controller for Windows programs. The measured values cannot be imported into EXCEL without this software.	450.00
Microsoft EXCEL 2000 (optional)	- EXCEL is used for the graphical display and analysis of the measured values.	-
Software costs (excl. options)		2736.00

3.3 Alternative Solution 2: Central Controller with a CPU 314C + MASTERDRIVE + a Synchronous Servomotor

Overview

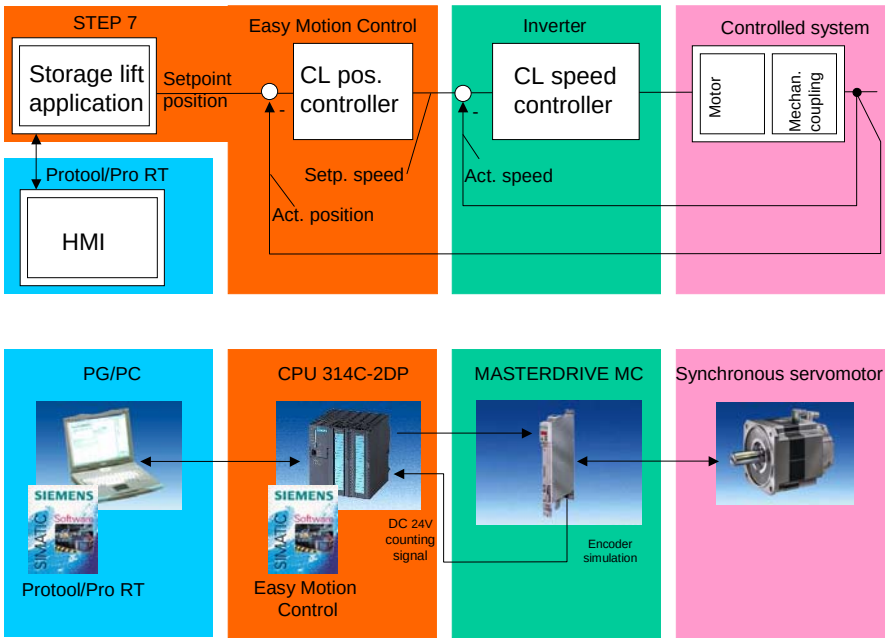


Figure 3-4 Overview

Functional description

The actual position or closed-loop position control functions are performed in SIMATIC by the Easy Motion function blocks. An analog output with ± 10V is used for communication with the MICROMASTER, which performs the speed control and activates the motor. The speed and the position are identified in the MASTERDRIVE and are output as a encoder simulation for the controller.

In contrast to the applied solution, this alternative solution uses a servomotor. This dispenses with the need for a holding brake because a servomotor can also hold a rated load continuously without overheating or without the need for a separately driven fan.

In another variant, the setpoint could be transmitted to the MASTERDRIVE via PROFIBUS, reducing the risk of parasitic interference to the setpoint.

Hardware components

The following hardware components are required to create the application. (June 2003 prices in EUROS)

Table 3-6 Hardware Components

Component	Picture	MLFB / Order number Function	L-price
Controller			
DIN rail		6ES7390-1AE80-0AA0 The DIN rail is the mechanical mounting rack for an S7-300 and is required to set up the controller.	25.70
PS307 2A power supply unit		6ES7-307-1BA00-0AA0 The power supply unit provides the requisite 24V DC supply.	89.00
CPU 314C-2DP (from FW 2.0!)		6ES7314-6CF01-0AB0 The CPU 314C-2 DP calculates the storage lift automation incl. the Easy Motion Control blocks. The encoder is read in via the integrated I/Os, and the setpoint speed is output.	1360.00
392 front-panel connector with screw contacts 40-pin		6ES7392-1AM00-1AA0 The front-panel connector provides a simple and user-friendly facility for connecting the sensors and actuators to signal modules. It is plugged into the module and concealed by the front door.	2 @ 31.60
Micro Memory Card 64 KB		6ES7953-8FL00-0AA0 The S7-program is stored on the MMC.	35.00

Component	Picture	MLFB / Order number Function	L-price
Drive			
SIMOVERT MASTERDRIVES MOTION CONTROL Compact Plus device		6SE7011-5EP50 The MASTERDRIVE supplies the current for the motor and works in closed-loop speed control mode.	1,119.73
Measured value encoder module for an SBM2 Multiturn encoder		6SX7010-0FE00 The SBM2 evaluates the signal from the motor encoder.	292.97
PROFIBUS CBP2 communication module		6SX7010-0FF05 The CBP2 is used to connect the MASTERDRIVE to the PROFIBUS.	234.17
SIMOVERT MASTERDRIVES 2 KW brake resistor for the Compact Plus design $P_{continuous}$ = 300 W		6SE7016-3ES87-2DC0	238.00
Motor			
Synchronous servomotor		1FK7033-7AK71-1EA3 The motor drives the axis.	848.30
Connecting cable: MOTION CONNECT leads		6FX5002-2XQ10-1AB0 6FX5002-5CA01-1AB0	35.82 55.04

Component	Picture	MLFB / Order number Function	L-price
HMI			
PG/PC with an MPI interface		- The PG/PC is used to process the HMI operator interface and to display the measured values in EXCEL	-
Costs			
Hardware costs (excl. options)			3276.50

Software components

The following software components are required to create the application. (June 2003 prices in EUROS)

Component	MLFB / Order number Function	L-price
STEP7 V5.2	6ES7810-4CC06-0YX0 STEP 7 is the basic package for the other software packages and is used for programming the S7	1,631.00
SIMATIC Easy Motion Control V2.0	6ES7864-0AC01-0YX0 Easy Motion Control	400.00
Starter (stand-alone) V3.x (optional)	On the Internet (see annex) Starter is a parameterization tool for the MICROMASTER. It uses the serial interface. It cannot be installed together with DriveES.	freeware
DriveES Basic V5.2 (optional)	6SW1700-5JA00-2AA0 DriveES is a parameterization tool for Siemens drives. It integrates them into STEP 7 and contains the "Starter". It uses the serial interface and/or PROFIBUS. It cannot be installed together with the Starter (stand-alone).	332.34
ProTool/Pro 6.0 + SP2 CS (optional)	6AV6582-2BX06-0CX0 ProTool/Pro is used for programming the HMI user interface. You cannot make any changes to the HMI user interface without this software.	1,675.00

Component	**MLFB / Order number** **Function**	**L-price**
ProTool/Pro 6.0 + SP2 RT128	6AV6584-1AB06-0CX0 ProTool/Pro RT enables a PG/PC to be used as an operator panel.	705.00
SIMATIC NET V6.x (optional)	6GK1704-5CW60-3AA0 SIMATIC NET contains an OPC server which supports access to the controller for Windows programs. The measured values cannot be imported into EXCEL without this software.	450.00
Microsoft EXCEL 2000 (optional)	- EXCEL is used for the graphical display and analysis of the measured values.	-
Software costs (excl. options)		2,736.00

4 Key Performance Data

The measurement has been performed with a load simulation.

To make it easier to follow, the simulation value settings for the **inertia torque, friction** and **gravity** correspond to the technical data for a storage lift, as shown in the table below.

Table 4-1

Parameter	Value
Storage lift	
Positioning distance from the basement to shelf 10	10 x 50cm = 5m
Weight of the lift support (basic load)	9.38 kg
Weight of the full box	4.11 kg
Drive	
Power of the standard asynchronous motor	180 W
Transmission ratio	10:1
Power of the MICROMASTER MM440	250 W
Technical parameters	
Max. speed of the lift support	1.06 m/s
Max. acceleration of the lift support	0.45 m/s^2

The MICROMASTER MM 440 can be operated in the following modes: V/f characteristic, SLVC and VC. Additional encoder signal acquisition is required for VC mode.

A multi-dimensional model is calculated in SLVC mode in the rectifier with the aid of the input motor data that is measured during motor identification. Using this vector model, actual speeds can be calculated on the basis of the size of the measured output currents, the output frequency and the motor voltage. This is used instead of an actual speed encoder.
Due to the principle, the model becomes increasingly imprecise as the speed approaches a standstill. Consequently, the standstill detection function proves to be problematic in principle, making this type of control unsuitable for the storage lift application. No measurements have been recorded with SLVC.

In order to assess the influence of the operating mode on the positioning result, the 6 measurements were repeated 10 times for each operating mode and are saved in the tables below.

The minimal, average and maximum deviations from the target position are shown.

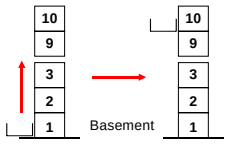
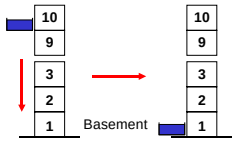
The measured deviation in mm represents the performance capability of the controller and drive combination.
In a real application, the tolerances of the mechanical parts, such as the extension of the cable, are increased.

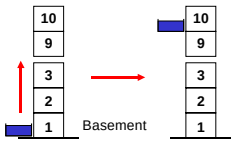
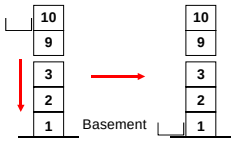
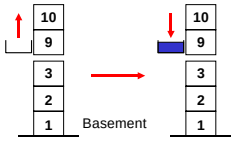
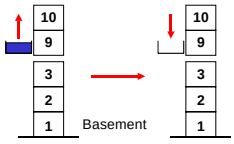
4.1 Measurements

A total of 6 measurements were performed. They reflect the charge and discharge of a box onto/from a shelf (shelf 10 has been selected).

Consequently, all measurements are recorded as the lift moves between the 0 mm (basement, charge/discharge station) and 5000 mm (shelf 10) positions.

Table 4-2

Measurement	Positioning precision		Why has this been measured?
	VC	V/f	
Measurement 1:  Motion without a box from the basement to shelf 10	min: 2.7 mm ∅: 2.8 mm max: 2.9 mm	min: -0.4 mm ∅: -0.5 mm max: -0.6 mm	Applied in conjunction with measurement 3, this is intended to examine the influence that load (moving against gravity) has on positioning accuracy. The result is also valid if no holding brake is used.
Measurement 2:  Motion with a box from shelf 10 to the basement	min: 2.3 mm ∅: 2.4 mm max: 2.4 mm	min: -55.9 mm ∅: -61.9 mm max: -75.0 mm	Applied in conjunction with measurement 4, this is intended to examine the influence that load (moving with gravity) has on positioning accuracy.

Measurement	Positioning precision		Why has this been measured?
	VC	V/f	
Measurement 3:  Motion with a box from the basement to shelf 10	min: 3.0 mm ∅: 3.0 mm max: 2.9 mm	min: -1.5 mm ∅: -1.7 mm max: -2.0mm	Applied in conjunction with measurement 1, this is intended to examine the influence that load (moving against gravity) has on positioning accuracy. Relative to the application, the motor has to apply the maximum torque in this case.
Measurement 4:  Motion without a box from shelf 10 to the basement	min: 2.4 mm ∅: 2.4 mm max: .,4mm	min: -15.1 mm ∅: -20.8 mm max: -24.9 mm	Applied in conjunction with measurement 2, this is intended to examine the influence that load (moving with gravity) has on positioning accuracy.
Measurement 5:  Motion to shelf 10, loading and motion towards the basement	Upward motion: min: 2.7 mm ∅: 1.3 mm max: 3.0 mm Downward motion: min: 2.3 mm ∅: 2.3 mm max: 2.3 mm	Upward motion: min: -0.8 mm ∅: -1.0 mm max: -1.8 mm Downward motion: min: -59.5 mm ∅: -78.7 mm max: -91.9 mm	The load varies before and after loading. During standstills, the lift support is fixed in place by the holding brake. This measurement is intended to show how jerky the downward positioning pattern when the lift starts to move.
Measurement 6:  Motion to shelf 10 with a box, unloading and motion towards the basement	Upward motion: min: 2.7 mm ∅: 2.8 mm max: 2.9 mm Downward motion: min: 2.4 mm ∅: 2.3 mm max: 2.3 mm	Upward motion: min: -1.7 mm ∅: -1.8 mm max: -2.0 mm Downward motion: min: -24.0 mm ∅: -31.0 mm max: -34.0 mm	The load varies before and after unloading. During standstills, the lift support is fixed in place by the holding brake. This measurement is intended to show how jerky the downward positioning pattern is when the lift starts to move.

4.2 Measurement Diagrams

One of the 10 recordings for each measurement is shown below:

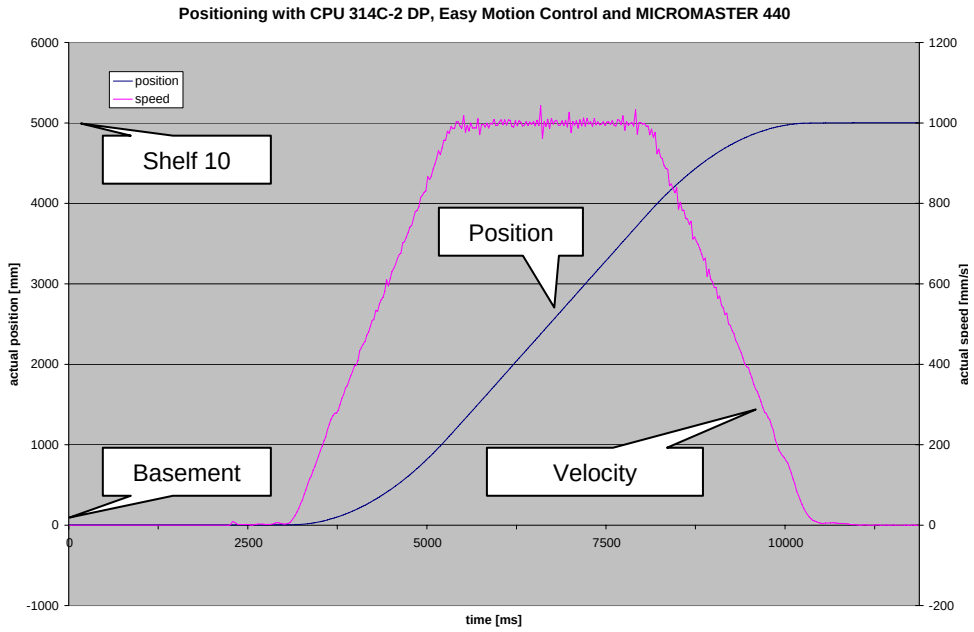


Figure 4-1 Measurement 1, VC

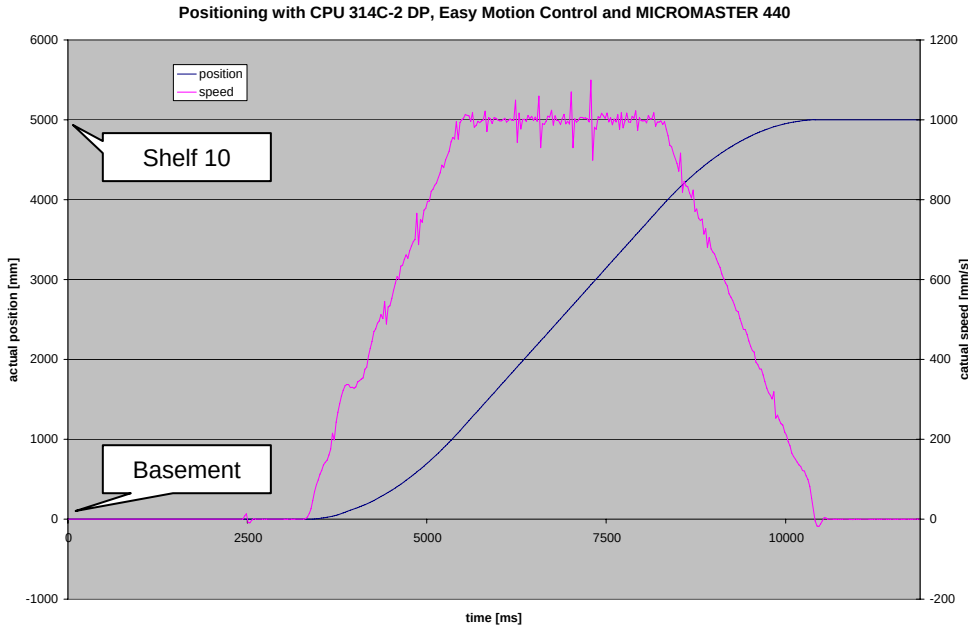


Figure 4-2 Measurement 1, V/f

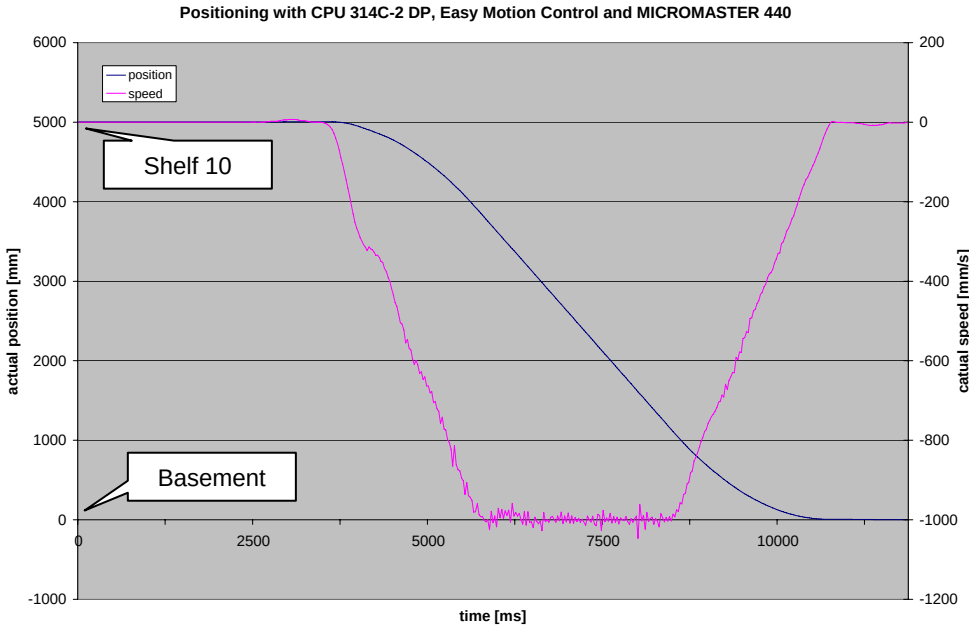


Figure 4-3 Measurement 2 VC

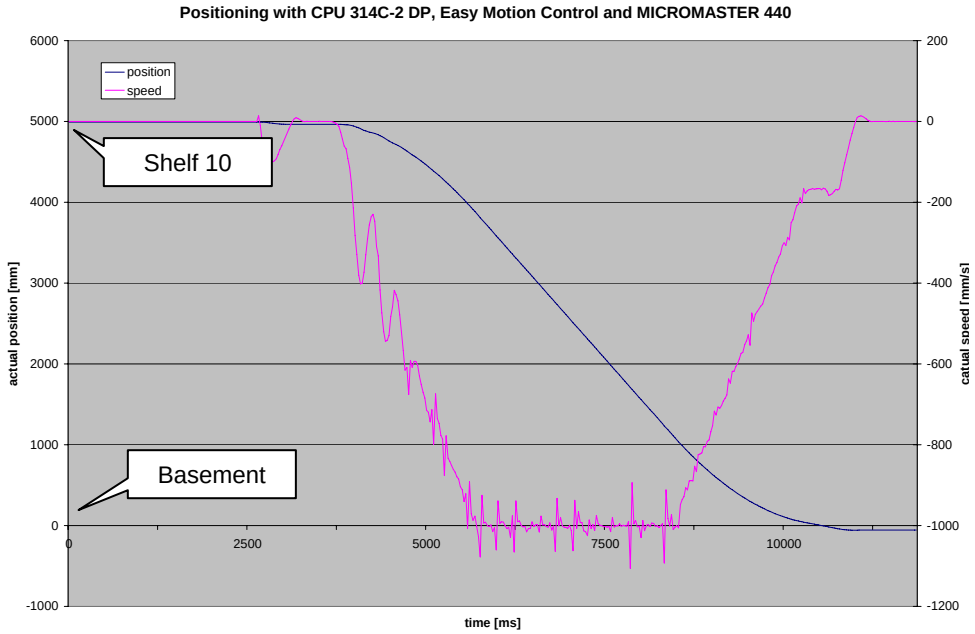


Figure 4-4 Measurement 2 V/f

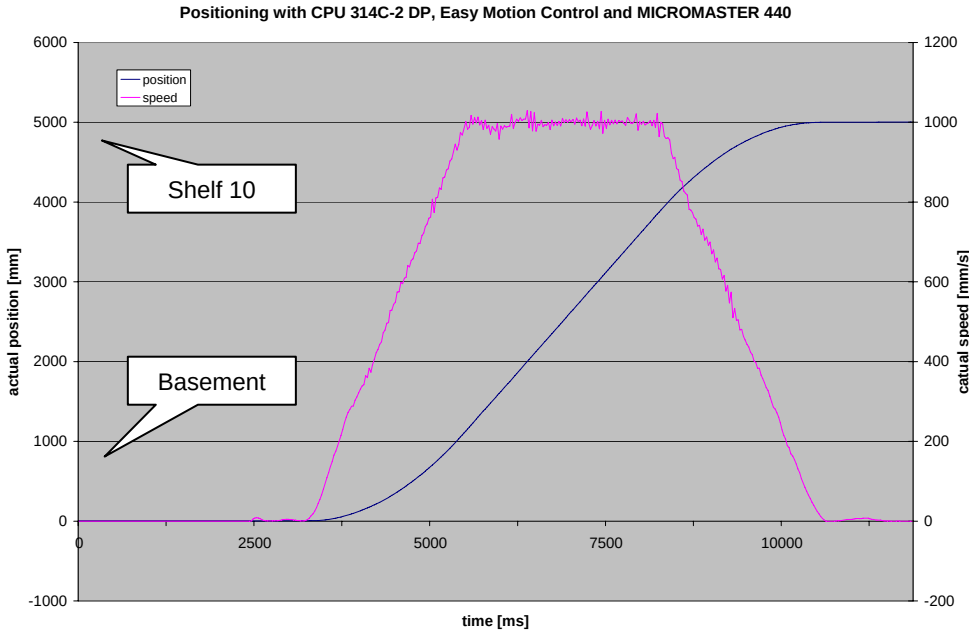


Figure 4-5 Measurement 3 VC

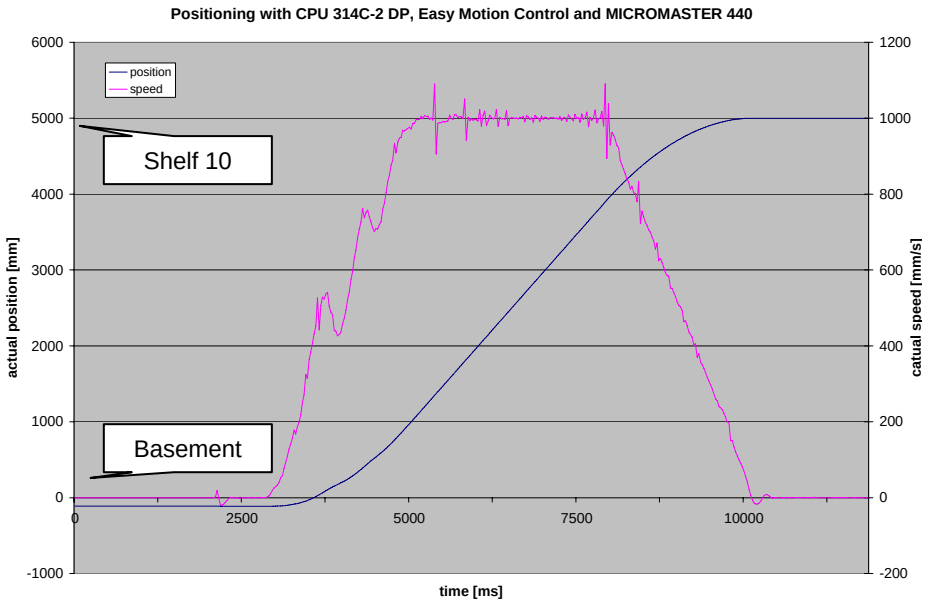


Figure 4-6 Measurement 3 V/f

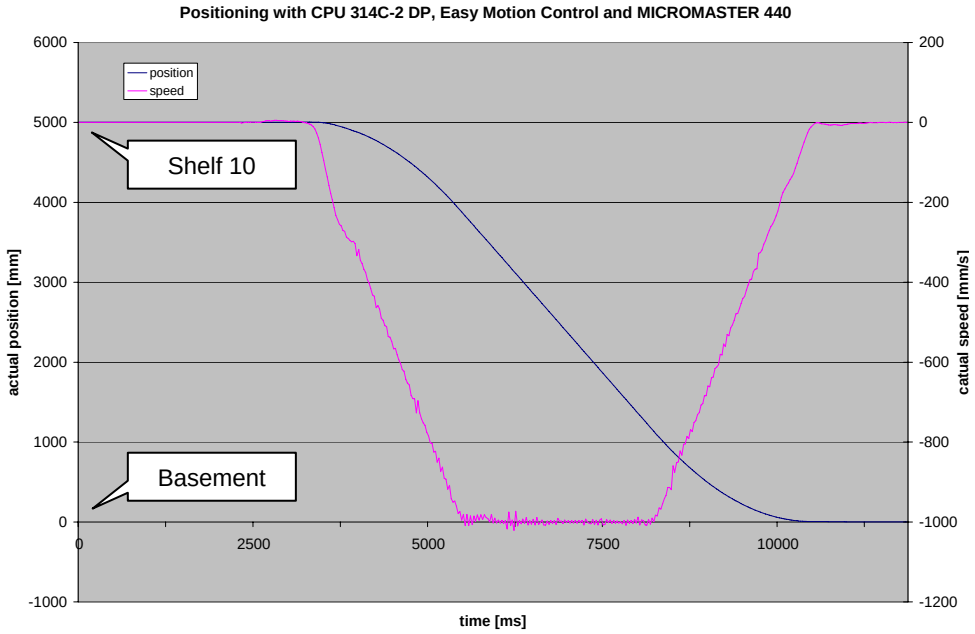


Figure 4-7 Measurement 4 VC

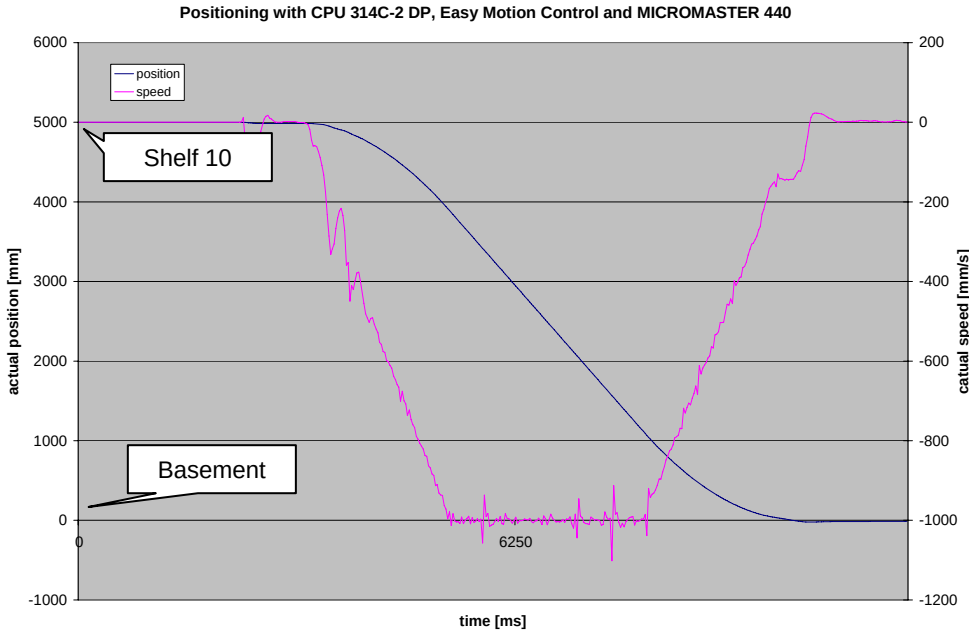


Figure 4-8 Measurement 4 V/f

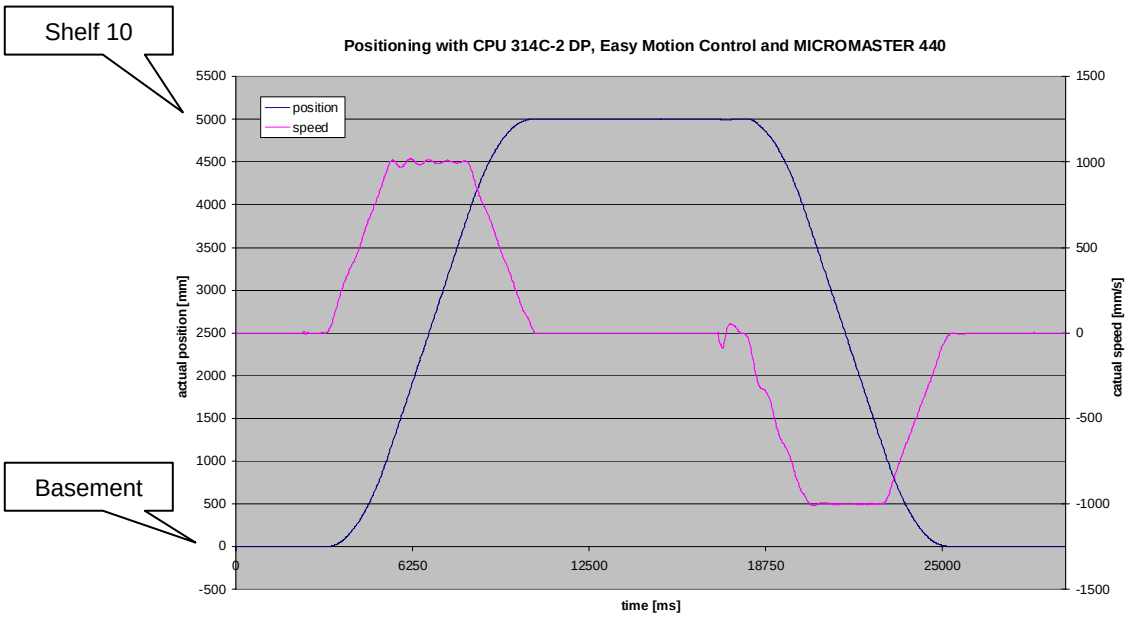


Figure 4-9 Measurement 5 VC

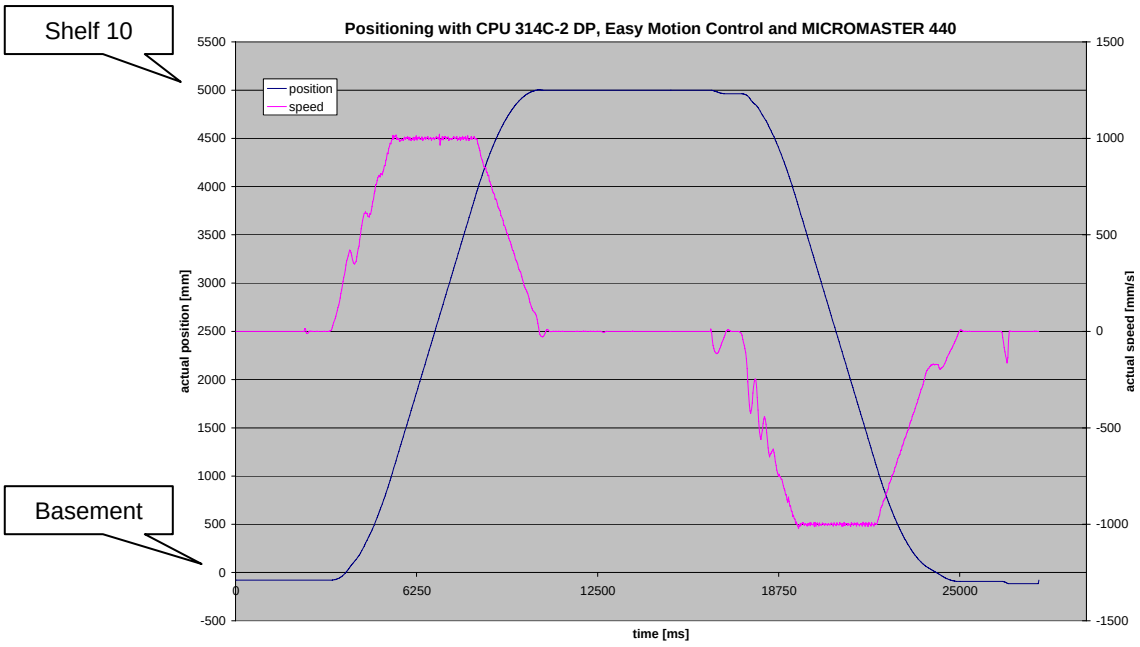


Figure 4-10 Measurement 5 V/f

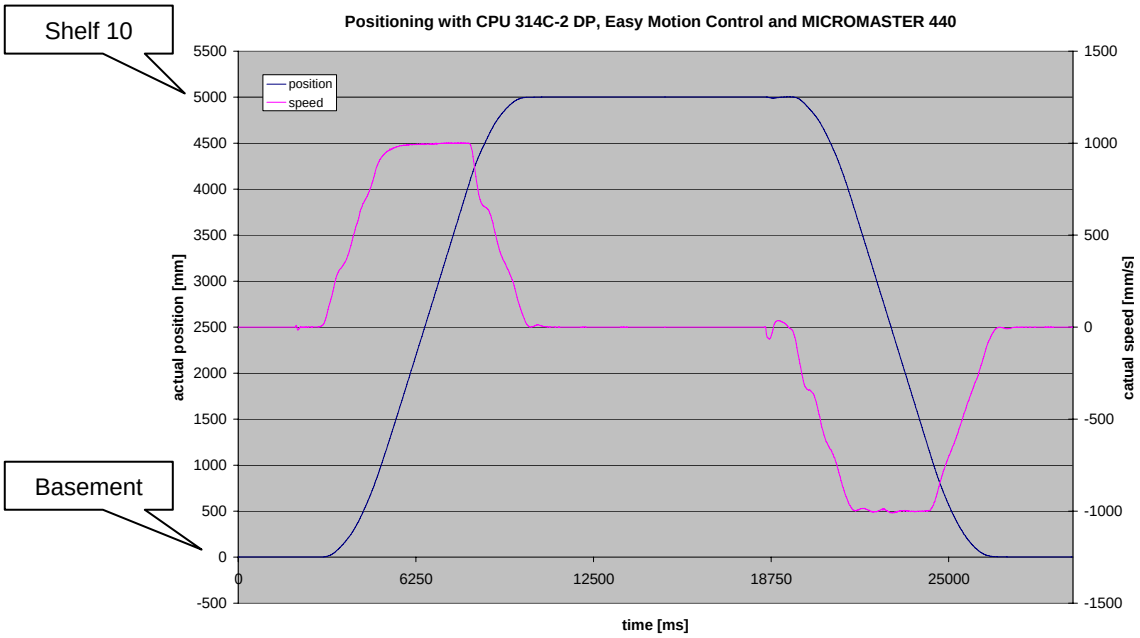


Figure 4-11 Measurement 6 VC

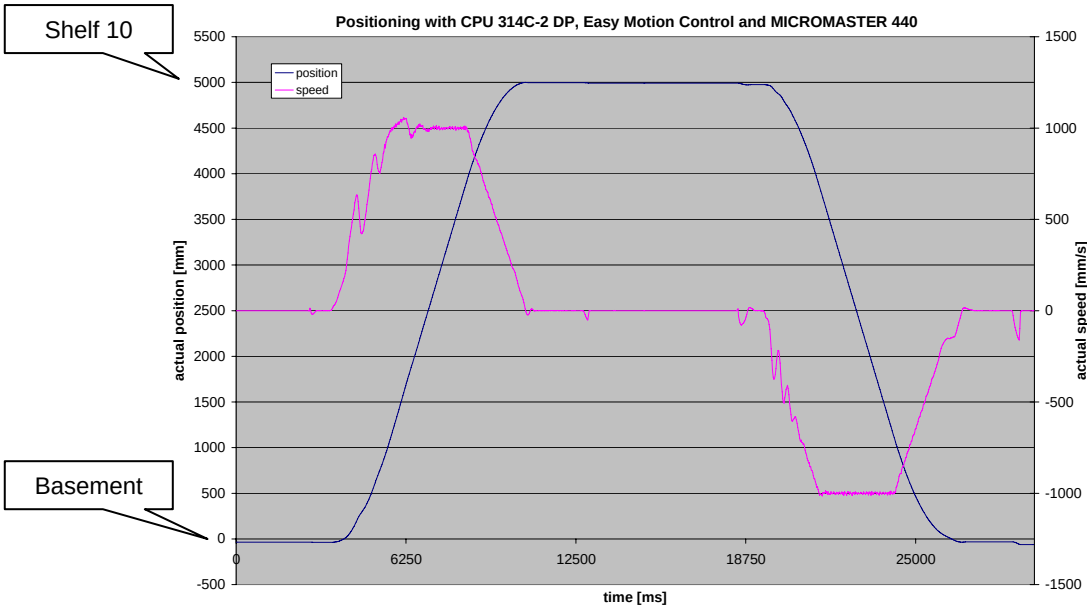


Figure 4-12 Measurement 6 V/f

4.3 What do the measurements tell us?

- MICROMASTER VC control delivers consistently good results and should, therefore, be used.
- SLVC control is unsuitable for this application.
- The whole system clearly becomes less stable in V/f setting.
  The closed-loop position controller in EASY MOTION CONTROL can compensate for this, although the setpoint speed yields much greater deflections. By the same token, there are also greater torque fluctuations and jolts.
  The result is clearly worse in motion in the direction of load (downward), in particular. This is probably due to the slip compensation. It only improves the results in motion against the load. This cannot be compensated for adequately by the higher-level closed-loop positioning control.
  Consequently, the low additional expense ought to be used for the MICROMASTER pulse encoder module, and Vector Control should be used. The pulse encoder is already in place for supplying the current position to the CPU314C.
- Unlike the V/f characteristic, the VC control parameters have to be optimized during commissioning, although there is a self-optimization function available for this.
- The measurement results can also be applied to more high-powered applications up to approx. 10 KW , complying with the following rules:
  - The power of the asynchronous motor, the holding brake and MICROMASTER must be adequately dimensioned.
  - Due attention must be paid to the thermal load, in particular (very frequent approaches to close targets at short intervals) because the motor does not achieve high speeds and, as a result, the cooling capacity of the fan is inadequate.
  - The design of the mechanical parts (gears, spindle ...) compromises the positioning accuracy.
  - The inertia torque of the motor should not be any less than half of the inertia torque of the load, otherwise fluctuations may occur between the load and the motor. If gears are used, the load-bearing capacity should be converted to the motor side with the transmission factor.
  - The encoder resolution should be 4 times higher than the positioning accuracy.
  - The max. frequency of the counter module should be approx. 20% higher than the maximum counting frequency.

Part A2: Functional Mechanisms

Overview

Content of Part A2

Part A2 looks at the detailed functional sequences of the hardware and software components involved. You only require this part if you wish to find out about the detailed sequence and the interaction between the solution components.

It contains information about the solution structure of the application. For example, the interaction between Easy Motion Control and application blocks is explained in greater detail here.

What can it be used for?

This application can be put to immediate use. You can put the application into operation without having worked through this section by following the installation instructions. However, should you wish to vary any particular parts of the application, this requires certain prior knowledge to enable you to incorporate your program sequences into the STEP7 code correctly and with minimal effort, for example.

Objective of Part A2:

Part A1 of this document is meant

- to explain all the functional elements to the reader
- highlight the components that are easy to integrate into your own applications.

Topics covered:

5 Functional Mechanisms

Functional component overview

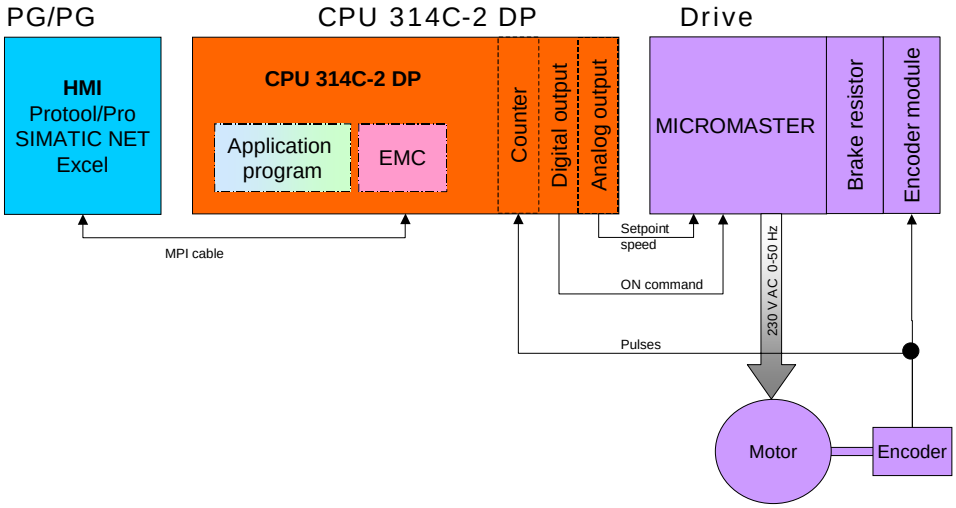


Figure 5-1 Interaction between the components

Description of the cyclical operation

- The charge or discharge order is input with the aid of the HMI.
- In the controller, this prompts a sequencer to be started in the application program, which sends the corresponding motion orders to Easy Motion Control.
- The Easy Motion Control blocks generate the voltage currently required at the analog output on the basis of the motion commands.
- The drive rotates at the speed, which corresponds to the voltage at its analog input, and moves the axis.
- The encoder signal is detected by the counter input of the CPU314C and is evaluated by Easy Motion Control. This closes the position control circuit.
- Once the target has been reached, a signal is issued by Easy Motion Control to indicate this, and the application program proceeds to the next step.
- Once the sequencer has been completely processed, the charge or discharge action is ended.

Description of the cyclical operation on a charge action

- The order to store a box in shelf 5 is input via the operator panel (ProTool/Pro RT).
- In the controller, this prompts a sequencer to be started in the application program.
  A check is performed to see if shelf 5 is still empty.
  The information "Position: 2500mm" is ascertained from "shelf 5". The value 2500 is assigned to an FB by Easy Motion Control, and the start bit is set.
- Easy Motion Control calculates the speed at which the drive currently needs to travel and outputs the requisite voltage via the analog output.
- The drive rotates at the speed, which corresponds to the voltage at its analog input, and moves the axis.
- The encoder signal is detected by the counter input of the CPU314C and is evaluated by Easy Motion Control. This closes the position control circuit.
- Once the target has been reached, i.e. the lift is now directly in front of shelf 5, the **done** bit is set by Easy Motion Control.
- The application program can tell from the **done** bit that the target has been reached and jumps to the next step.
- The transfer step takes place, where the box is placed in shelf 5 (simulated).
- The charge action is ended.

5.1 Description of the Components

The diagram below illustrates a simplified function plan. It is shown bit-by-bit in greater detail in the following sections.

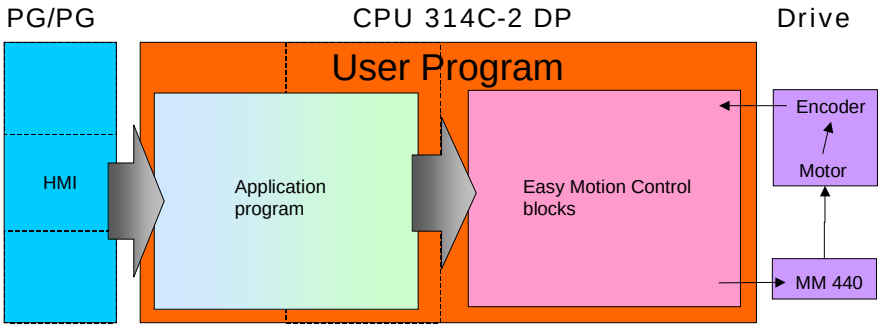


Figure 5-2 Function diagram, overview

Orders are input via HMI. The application program converts them into positioning orders for Easy Motion Control. Easy Motion Control calculates the setpoint for the drive and evaluates the encoder.

5.1.1 PG/PC (HMI)

The 'storage lift' sample application is visualized and operated with the PG/PC. A ProTool/Pro RT application has been created for this purpose. It retrieves the requisite data from the CPU314C. No extra codes are required in the CPU314 for the HMI interface.

The HMI comprises three masks:

- Manual mode
  Mode for moving the storage lift by hand and for synchronizing after switch-on
- Automatic mode
  Mode for charging and discharging the boxes
- Store
  Diagram of the store, illustrating the current stock

The measured values can also be imported into EXCEL 2000 on the PG/PC with the aid of the OPC server in the SIMATIC NET package.

The HMI area of the storage lift application is shown in correspondingly greater detail in the graphic below. The OPC server has not been shown.

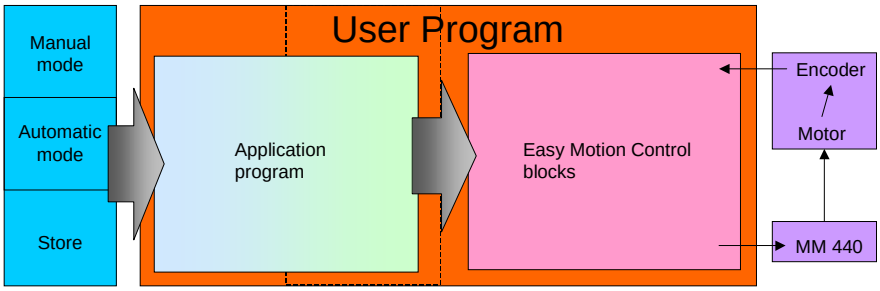


Figure 5-3 HMI function chart

5.1.2 CPU 314C-2 DP

The CPU314C-2 DP was selected on account of the integrated counter and analog output. This means that all the requisite interfaces are available. No other expansion modules have to be used. The CPU314C has 4 integrated counters, supporting the use of 4 encoders with Easy Motion Control. However, communication is only possible with two axes via each analog interface unless there are any further expansion modules.

The internal counters in the CPU314C are used for position detection purposes. Careful attention has to be paid to the maximum signal frequency. It may not be exceeded; otherwise, all the pulses cannot be detected, and the actual position is calculated incorrectly.

The user program is processed by the CPU314-2 DP. The I/O integrated in the CPU is used to count the encoder pulses or to output the target speed as a ±10V signal.

The user program is divided into the following two parts

- Application program:
  Blocks created for the sample storage lift, containing the store management and the creation of positioning orders for Easy Motion Control.
- Easy Motion Control:
  Function blocks for position-controlled positioning and gear synchronism.

The two parts of the user program are described in Sections 5.1.4 and 5.1.5.

5.1.3 Drive

The drive comprises an inverter and motor and converts the ± 10V of the setpoint speed into the corresponding speed. For this purpose, the inverter also analyzes the encoder, creating a closed-loop speed control circuit.

For reasons of costs, the setpoint speed is transmitted in the form of a ± 10V signal because both the CPU314C and the MICROMASTER have this interface, which means that there is no need for any other modules. If transmission were to take place via the PROFIBUS, an expansion module would be required for the MICROMASTER.

Brake

Once the target position has been reached, the service brake is engaged and the motor is taken out of control, i.e. switched off.
If an asynchronous motor is operated on an inverter, it is also able to apply the rated torque at low speeds or at a standstill. However, it also reaches the "rated power loss" then. Given the low speeds, the cooling by the integrated fan, which sits on the drive shaft, is inadequate, causing the motor to overheat if it has to hold the load at a standstill for an extended period of time.
This can be got around using a holding brake or with the aid of a separately driven fan. A holding brake has been selected in the application.
The MICROMASTER activates the brake. It has an integrated function that ensures that the brake is only released if the motor can apply the load torque or engages the brakes first, then switches off the motor.

Gears

The gears allow a motor to be used at higher speeds and at a lower torque. It reduces the speed and increases the drive torque at the load side.

Brake resistor and network filter

When brakes are applied to the drive, the motor works as a generator and energy flows back into the inverter. This causes an increase in the intermediate circuit voltage in the inverter. As the MICROMASTER is unable to feed the energy back into the supply network, it activates the brake resistor which converts the energy into heat, thereby lowering the intermediate circuit voltage again.
The brake resistor is also known as the chopper or pulsed resistor.

The network filter is meant to reduce disturbance from the MICROMASTER's network rectifier, which means that it can also be used in an office environment.

5.1.4 SIMATIC Easy Motion Control

Easy Motion Control is a cost-efficient package for simple position-controlled positioning and for simple gear synchronism of up to three axes. It is also suitable for use with any variable-speed standard drive, such as a frequency inverter or servo drive.
It support incremental and absolute encoders.

Easy Motion Control can run on the following CPUs:

- S7-300, CPU 314 or bigger
- S7-400
- C7, from C7-633
- WinAC

Easy Motion Control contains:

- Function blocks for position-controlled positioning and gear synchronism
- Drivers for SIMATIC modules or DP coupling with MM4:
  The actual position is read in and the setpoint speed output via standard SIMATIC program modules. There are driver modules included for the most common modules. Other modules can be linked via a free driver.
- Configuration software for parameterization and commissioning.

Easy Motion Control features the following basic positioning functions.

- *Move absolute*
  The axis moves to the specified position. In the case of rotary axes, the target can be approached in a positive or a negative direction or by the shortest route.
- *Move relative*
  The axis moves in a positive or a negative direction by the specified distance.
- *Search for reference, setting the reference point*
  Synchronizes the measuring system with the mechanical parts.
- *Gear synchronism*
  The axis runs to another axis during gear synchronism.
- *Jogging*
  The axis keeps moving while there is a control bit active for a direction.
- *Stop*
  Interrupts the motion and brings the axis to a standstill.

Other properties:

- *Simulation*
  A program can be tested in the simulation even if there is no real axis present and no actual values are read in.
- *Override*
  A factor of between 0 and 100 % can be applied to the motion velocity at any time. This has no affect on acceleration and deceleration.
- *Replacement motion*
  Any motion block can replace any other motion block. The current movement is transferred to the new movement with the specified acceleration or deceleration.
- *Configuration software for parameterization and commissioning*
  The configuration software simplifies the parameterization, commissioning and diagnostics of Easy Motion Control axes greatly.

Interaction between the Easy Motion Control blocks

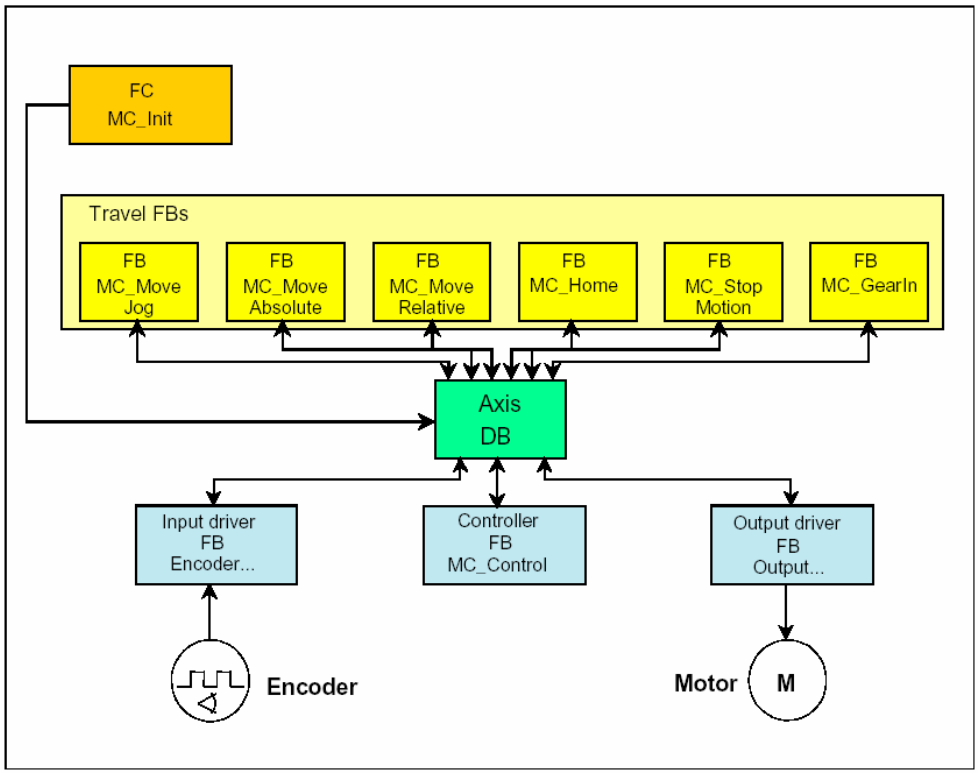


Figure 5-4 Interaction between the Easy Motion Control blocks

The Easy Motion Control blocks can be divided into three categories:

- **Input and output drivers** (FB Encoder...)
  The input driver is used to read in the encoder signal or the actual position. The output driver signals the setpoint speed to the drive or the power circuit.
  Easy Motion Control contains a total of nine input and seven output drivers. The appropriate input and output drivers are selected depending on the hardware used.
- **Position controller** (FB MC_Control)
  The closed-loop control block performs the position control.
- **Motion blocks** (FB MC_....)
  The motion blocks are used for jogging, synchronizing and moving the axis.

These function blocks are supplemented by an initialization block (FC MC_Init), an **axis DB** template (UDT) and configuration software for parameterization and commissioning.

An **axis DB** is created by the programmer with the aid of the template for each axis. This DB contains all the information about the axis and has access to all the Easy Motion Control FBs.

The configuration software for parameterization and commissioning can be used to assign parameters to the axis DB simply and clearly. It can also be used to display the current status of the axis very simply.

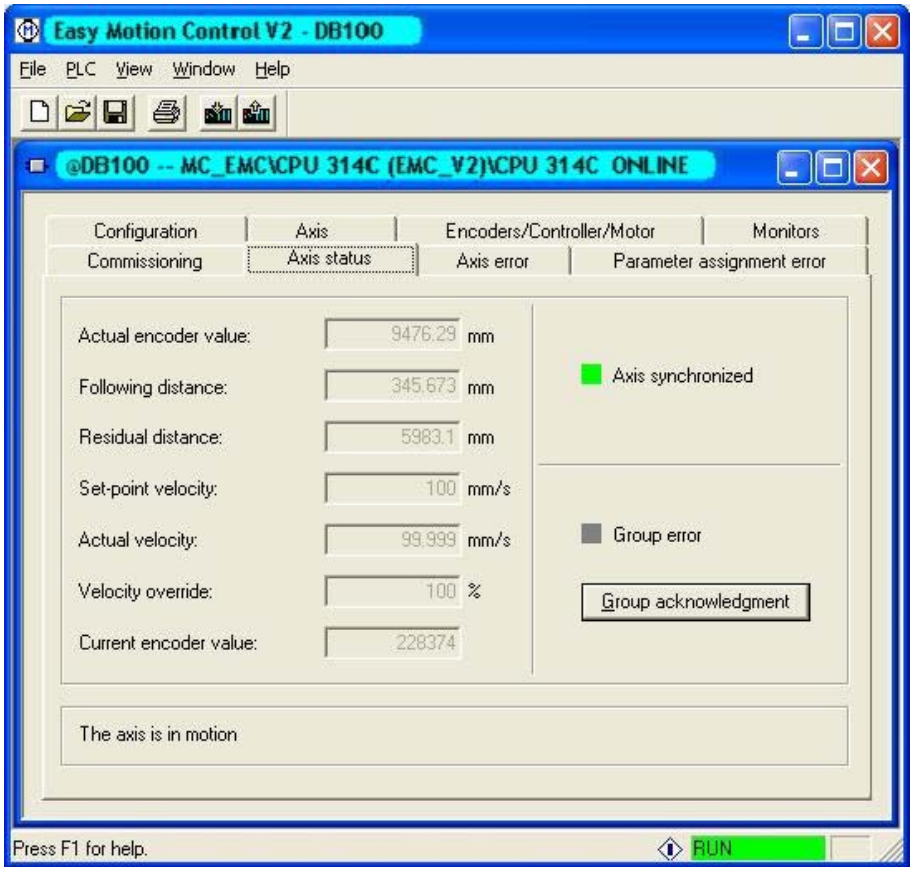


Figure 5-5 Status of an axis

Calling the Easy Motion Control blocks

In order to optimize the closed-loop control performance, the Easy Motion Control blocks have to be factored in equidistantly. Consequently, they should be factored in a watchdog interrupt, e.g. in the OB35.

In practice, it has proved best when calling Easy Motion Control blocks to only assign parameters directly to the axis DB and the initialization bit; create the positioning tasks, on the other hand, by writing directly to the instance DBs.

Note

Where reference is made hereinafter to calling the Easy Motion Control block, this means writing a positioning task to the instance DB in the Easy Motion Control block; this serves to activate the block, and positioning starts. While Easy Motion Control blocks are always factored in, they are generally (inactive).

Easy Motion Control blocks used in the sample application

Only some of the EMC blocks are used in the application, which also means that only some of the functions supported by Easy Motion Control are used.

Note

The memory requirements of the Easy Motion Control package are reduced by virtue of the fact that only blocks that are used are loaded into the CPU.

For information about the memory requirements of the individual Easy Motion Control blocks, please refer to Table 6-7 List of Easy Motion Control FBs und FCs used

Table 5-1 Blocks used in the Easy Motion Control package

Name	FB No.	Function
EncoderCPU314	FB 28	FB for evaluating the counter in the CPU 314C and, thus, for evaluating the encoder information.
MC_Init	FC 0	Initializes the EMC blocks during restart
MC_MoveJog	FB 3	Motion FB for moving the axis manually
MC_MoveAbsolute	FB 1	Motion FB for positioning the axis
MC_MoveHome	FB 4	Motion FB for the search for reference for the axis or for setting a reference point
MC_MoveStopMotion	FB 5	Motion FB for stopping motion
MC_Control	FB 11	FB for the position controller
OutputCPU314C	FB 34	FB for outputting the setpoint speed via an analog output on the CPU 314C
Axis DB	DB100	DB containing all the data on the axis; is created by the programmer with the configuration software or the UDT.

The Easy Motion Control area of the storage lift application is shown in correspondingly greater detail in the graphic below.

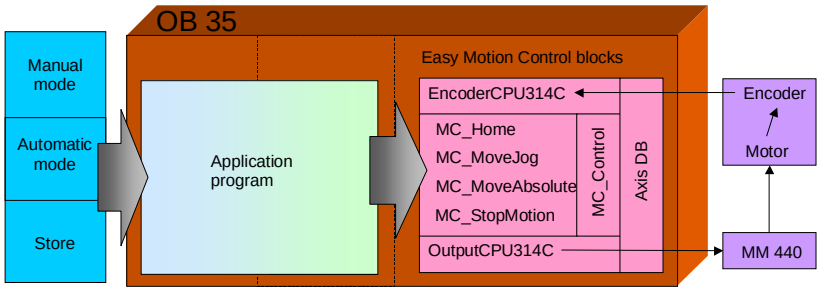


Figure 5-6 Program structure: Easy Motion Control blocks

5.1.5 Application Program

The application program comprises blocks which have been created to link or expand the functions of the Easy Motion Control blocks.
The Easy Motion Control blocks are created for achieving individual positioning tasks, although a sequence of positioning tasks is required for charging or discharging the boxes. Consequently, the application blocks have been created which call up the Easy Motion Control function currently required in sequences.

The functions in the application program are divided into two parts:

- Blocks for manual mode:
  - FB sync, for synchronizing the axis
  - FB jog, for jogging the axis
  - FB pos for positioning the axis manually
- Blocks for automatic mode:
  - FB auto, for charging and discharging boxes
  - DB stock, represents current stock.

Later sections will take a closer look at how the individual FBs work.

In order to facilitate the evaluation of signals that are only set by Easy Motion Control for a single cycle, the application blocks in the OB35 are also factored in.

The area of the application program is shown in correspondingly greater detail in the graphic below:

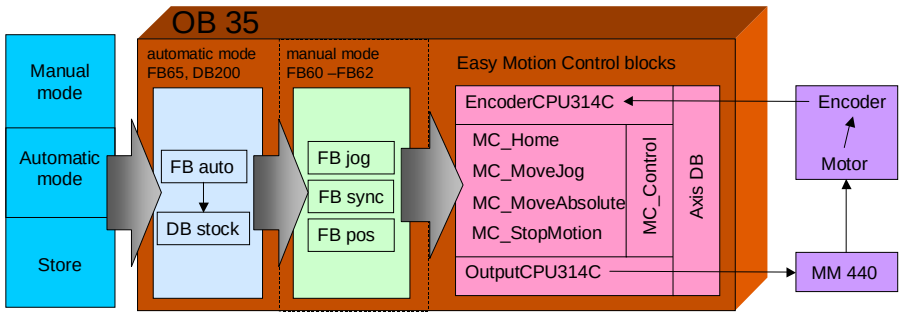


Figure 5-7 Program structure: Application program

5.1.5.1 Searching for Reference / Synchronization (FB sync)

Technical sequence

A standard incremental encoder has two pulse tracks that are counted and evaluated in the controller. Only the distance traveled after switching on or resetting , i.e. the relative position, can be determined. Consequently, a search for reference has to be performed to assign or synchronize the relative position to or with an absolute position.

For this purpose, the drive is moved until the signal is received from the reference point switch, and then the value for the actual position is set to the known value for the reference value in the controller. As a result, the axis is then synchronized.

The search for reference should always take place at low speed and in the same direction in order to increase accuracy.

Consequently, the lift is moved downwards until it reaches the lower limit switch. It is then moved upwards slowly until the signal is transmitted by the reference point switch. At this point in time, the value for the actual position is set.

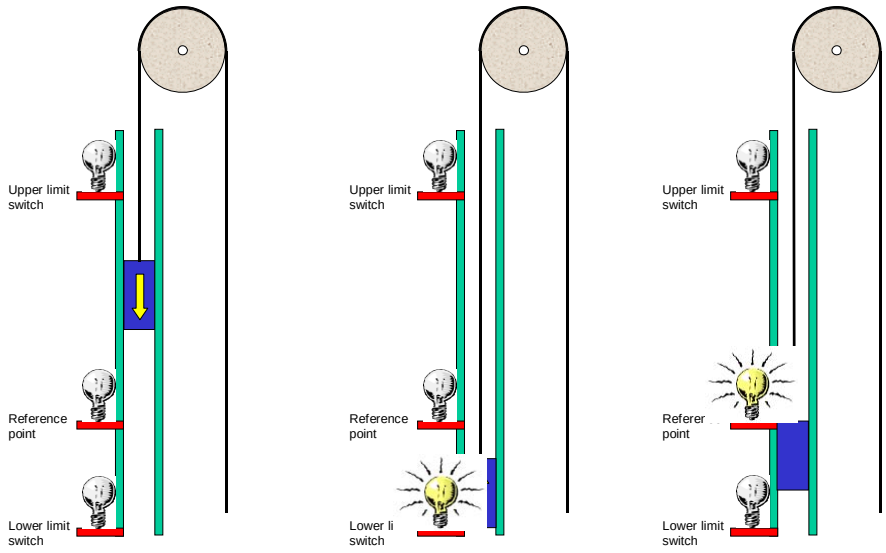


Figure 5-8 Synchronization of a lift axis

Note

The terms search for reference and synchronize each refer to the same function.

Implementation

Synchronization takes place in manual mode; accordingly, the blocks for automatic mode are disabled. The synchronization of the axis is performed by the FB sync function block.
The user calls up the synchronize function via the HMI. This is signaled directly to the FB sync. It calls the MC_jog und MC_Home Easy Motion Control blocks.

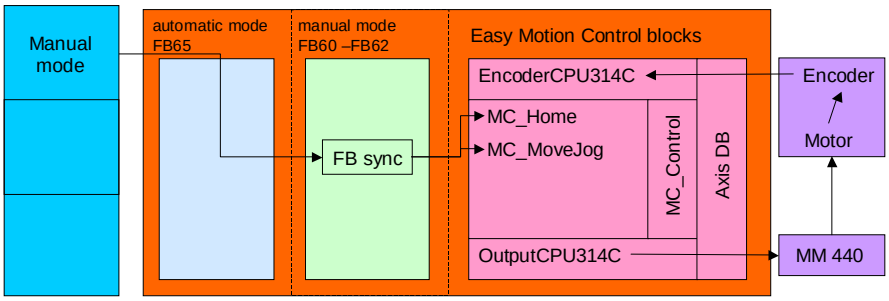


Figure 5-9 Synchronize function diagram

First of all, the axis is moved downwards with the aid of the MC_MoveJog function block until the limit switch is activated. The MC_Home synchronization block is then called. It moves the axis upwards and automatically stops after synchronization.

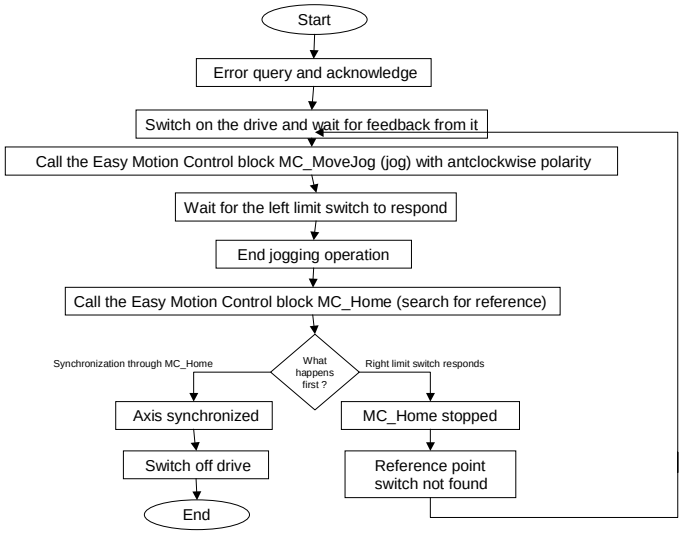


Figure 5-10 FB sync flowchart

5.1.5.2 Jogging (FB jog)

Technical sequence

Jogging is used to move the axis manually at a constant speed. The axis keeps moving while the pushbutton is pressed.

Note

The small acceleration limit means that there is an after-run. Consequently, the axis does not come to an immediate stop, it slows down with the permissible deceleration.

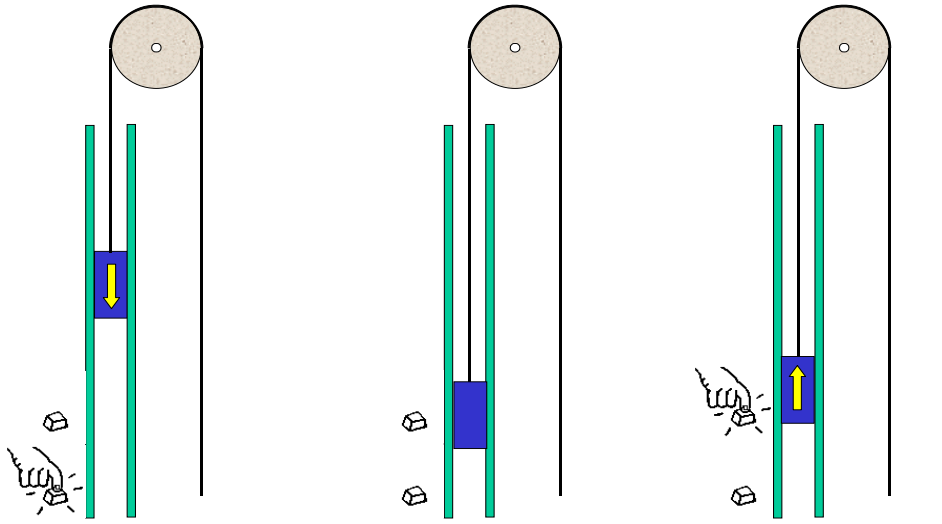


Figure 5-11 Jogging a lift axis

Implementation

Jogging takes place in manual mode; accordingly, the blocks for automatic mode are disabled. The jogging of the axis is performed by the FB jog function block.
The user calls up the jogging function via the HMI. This is signaled directly to the FB jog. It calls the MC_jog Easy Motion Control block.

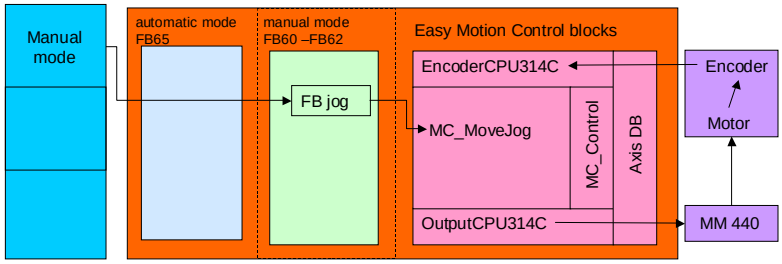


Figure 5-12 Jogging function diagram

In order to facilitate jogging, jog mode must be switched on first. As a result, the drive is automatically switched on and holds the load. The axis is moved while the corresponding jog signal is on. If there is no jog signal present, the axis stands still.

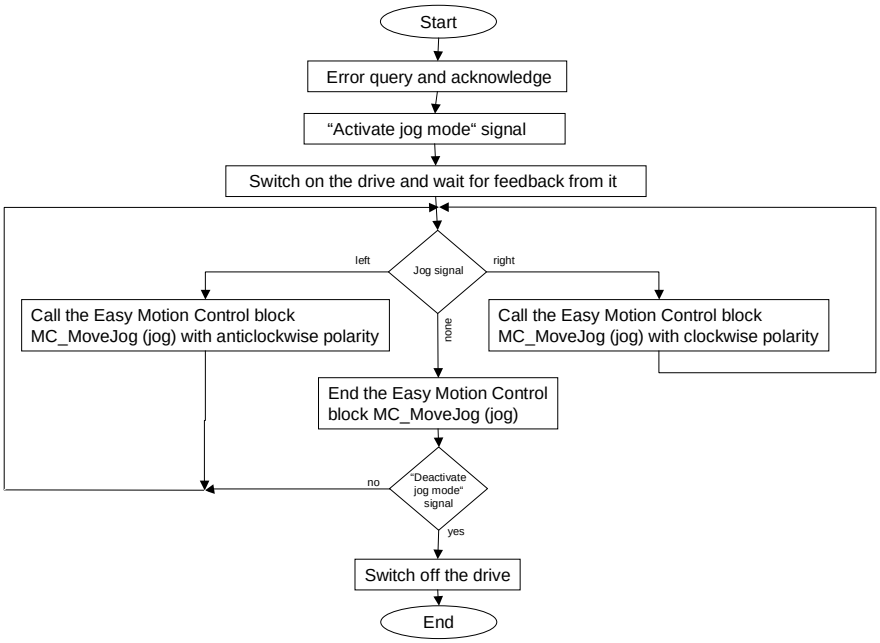


Figure 5-13 FB jog flowchart

Note

The closed-loop position controller is activated during jogging in Easy Motion Control! In other words, the encoder signal has to be connected, and the Easy Motion Control encoder parameters have to be correctly input; otherwise, the drive starts up at high speed before Easy Motion Control gets to switch it off with the error messages "Following distance exceeded" or "Standstill range exited“.

5.1.5.3 Positioning (FB pos)

Technical sequence

Positioning is used for the purpose of positioning the axis manually. The requisite position is entered, and the axis is moved to this position after the start command.

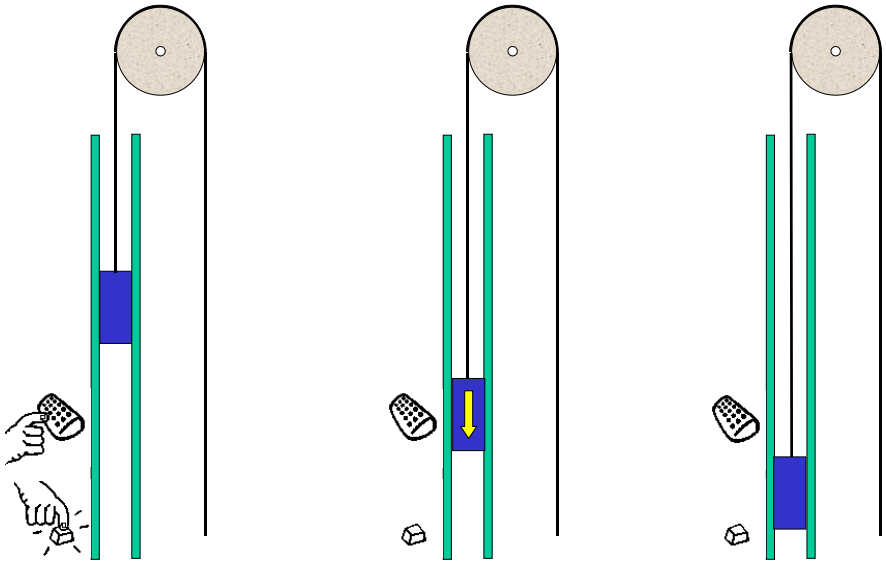


Figure 5-14 Positioning a stroke axis

Implementation

Positioning takes place in manual mode; accordingly, the blocks for automatic mode are disabled. The positioning of the axis is performed by the FB pos function block.
The user calls up the positioning function via the HMI. This is signaled directly to the FB pos. It calls the MC_MoveAbsolute Easy Motion Control block. If positioning is stopped via a HMI entry, the current positioning sequence is stopped by calling the MC_StopMotion Easy Motion Control block.

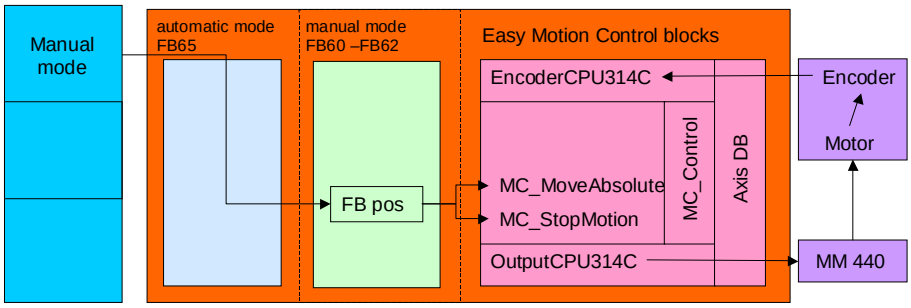


Figure 5-15 Positioning function diagram

FB pos is divided into two areas:

The area with job generation is always calculated completely. This serves to ensure that no positioning job goes missing. It is processed with an exchangeable buffer system. While one job is active, a new job can be set in the other buffers any number of times. An existing job is overwritten. The switch is made from the active job to the new job in part 2 of the FB pos.

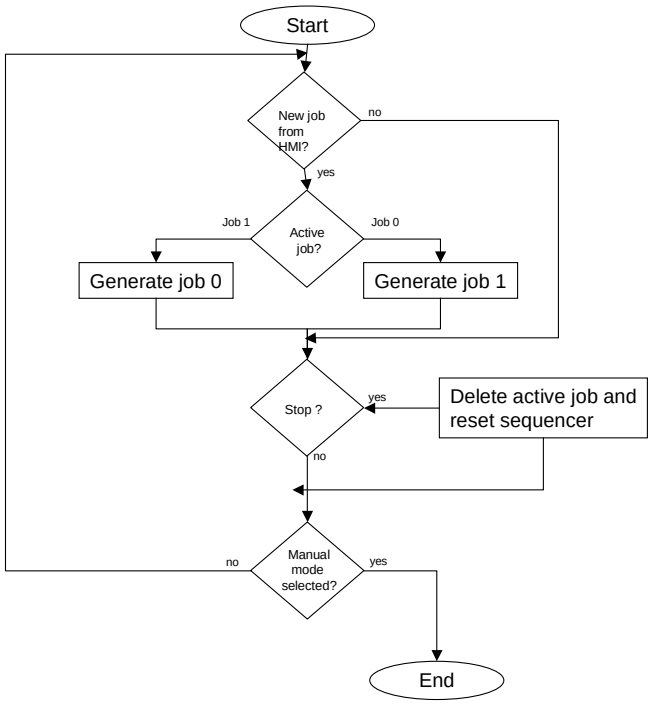


Figure 5-16 FB pos flowchart, part 1

Part 2 of the FB pos is the sequencer which facilitates the processing of the positioning jobs which are saved temporarily in job0 or job1:

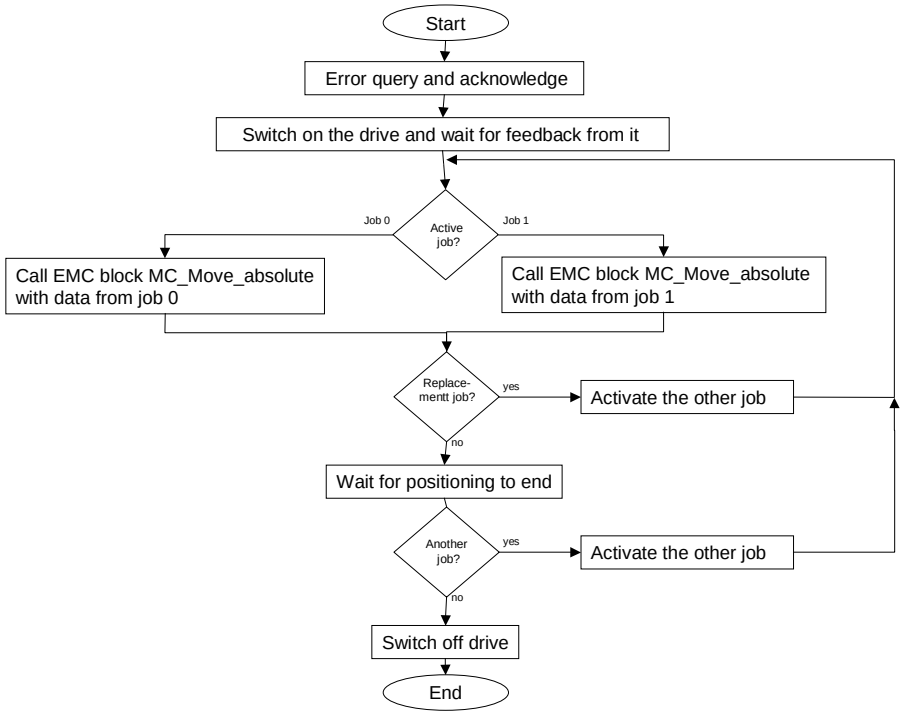


Figure 5-17 FB pos flowchart, part 2

One job is normally not activated until the other one has ended. However, if the replacement job bit is set, the new job is activated immediately. EMC switches straight from the old positioning job to the new one. Due account is taken of the maximum acceleration and velocity.

This action is observed in the storage lift sample application when a discharge job is entered into the system while the lift is returning to its idle position after a charge job.

5.1.5.4 Automatic Operation (FB auto, DB stock)

Technical sequence

The storage lift is ready for service in automatic mode and awaits incoming charge or discharge orders for processing.

Charge sequence:
The storage position is entered, and the box is placed in the lift. It travels to the requisite shelf, where the box is stored.

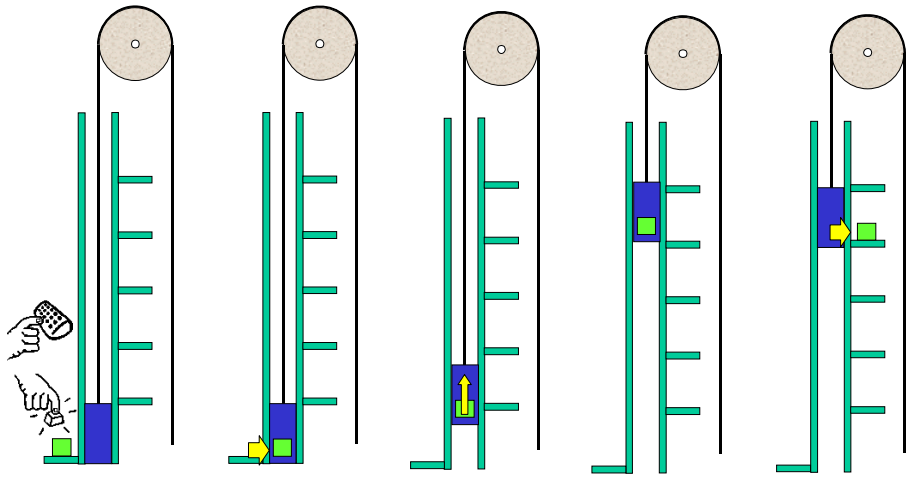


Figure 5-18 Charging a box

Implementation

The orders are entered via the HMI and are written to the instance data block of the FB auto application block.
The FB auto refers to the DB stock to check whether the order can be carried out or whether the shelf is already occupied by something else. If the order is viable, the shelf position is removed from the DB stock, and the FB pos is called with this value.
The Easy Motion Control FB MC_Absolute is called in the FB pos sequencer. FB MC_Absolute then performs the positioning tasks.

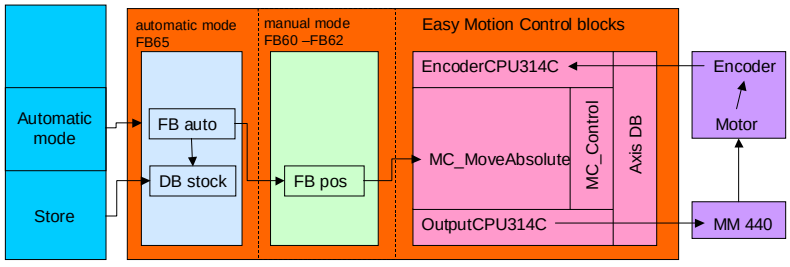


Figure 5-19 Automatic mode function diagram

DB stock does not just show the current occupancy of the store, it also specifies the positions of the individual shelves in mm. A UDT has been defined which is saved 26 times in each array (25 shelves and the loading/unloading station). The current stock can be viewed and edited via the HMI.

FB auto has two functional parts.
One part switches between modes. This part is always active, even if automatic mode is not selected. In this case, the sequencer dwells in the "Manual mode, control via HMI" step. Data is supplied to manual mode FBs (FB pos, FB jog and FB sync) via the HMI's "manual mode" page.

The only way to exit this step and to process the automatic mode parts of the sequencer is to select automatic mode. When the mode change occurs, the "Automatic Mode" page is displayed in the HMI, i.e. the part shaded in gray:

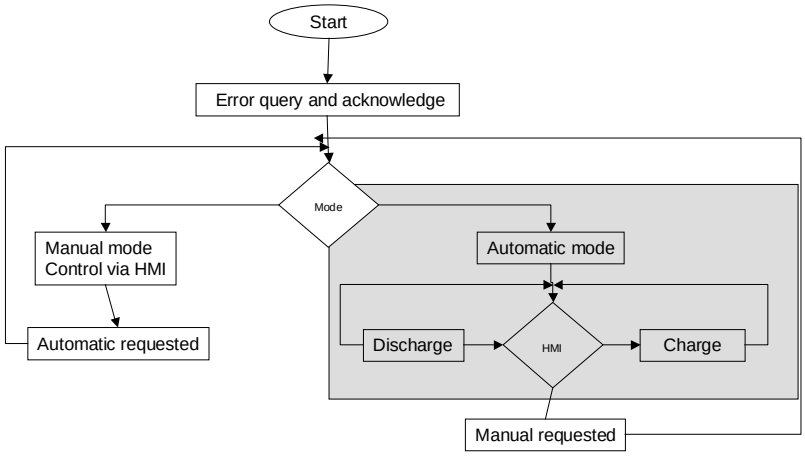


Figure 5-20 FB auto sequence, part 1

The second part is shaded in gray and is shown in greater detail below.

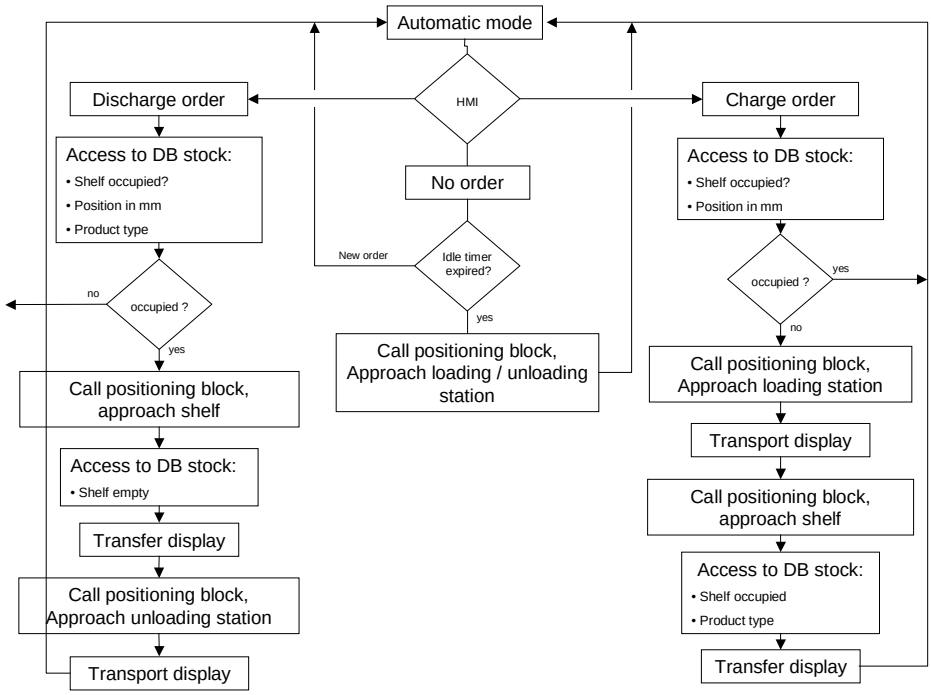


Figure 5-21 FB auto sequence, part 2

Whenever a new charge order is entered, a check is performed first of all to see if the requisite shelf is empty. If it is, the position of the charge station (shelf 0) is taken from the DB stock, and the FB pos is called with this value. Once the position is reached, this means that the charge station has been reached.
Once the box is placed in the lift (transport display), the position of the selected charge station is taken from the DB stock. The lift travels to the selected shelf.
The charge process is simulated there by the transfer display, and the shelf is marked occupied in the DB stock.
This completes the charge process.

The discharge process follows precisely the same pattern.

If, after a charge job for example, the lift is not in its idle position (basement, charge and discharge station) and if there isn't another job pending, the timer starts running. Once 10 s have elapsed on the idle timer, the lift comes back down automatically. If a discharge job is entered while the lift is en route to its idle position, replacement positioning occurs, whereby the current positioning task is transferred to the new job.

5.1.5.5 Measuring (FB measure)

Technical sequence

The current position value and the current velocity are saved by Easy Motion Control in the axis DB. They can initially be saved in two DB measure values for the purposes of evaluation of graphical display later on. This data can be accessed from EXCEL 2000 with the aid of the OPC server.

Implementation

The user can start to record values in both operating modes. The FB measure copies the actual position and the current velocity for each cycle to DB measure value 1 and 2 until recording via the HMI ends or until the value DBs are full.
The values are also copied in the OB35, creating a permanent time reference.

A macro is started in EXCEL on the PC/PG, enabling the values to be exported to the controller via the OPC server.

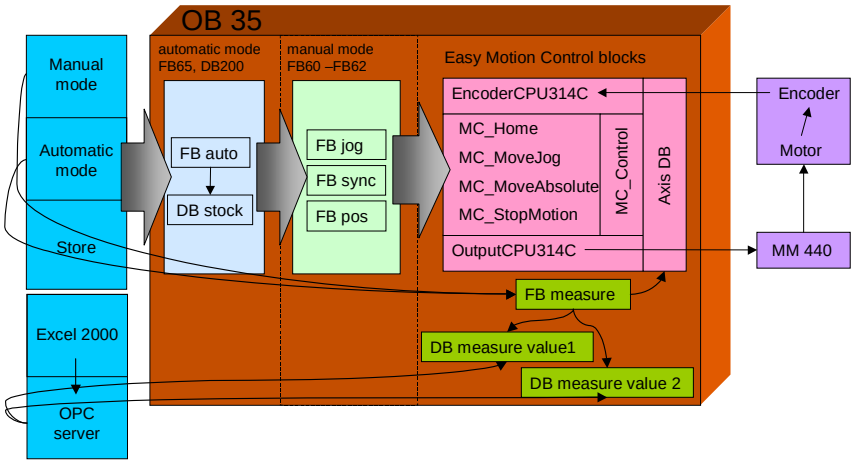


Figure 5-22 Measuring function diagram

5.1.6 Special Features of the Interaction between the Drive and EASY MOTION CONTROL

As there is a motor holding brake used in this application, a delay occurs while the drive tracks the setpoint after switch-on.

In order to prevent the closed-loop position controller in EASY MOTION CONTROL increasing the setpoint further during this time and a jolt occurring when the drive starts after the brake is released, there is a one-second delay after the motor is switched on before the first positioning job is sent to EASY MOTION CONTROL. During this time the motor builds up the torque and the brake is then opened.

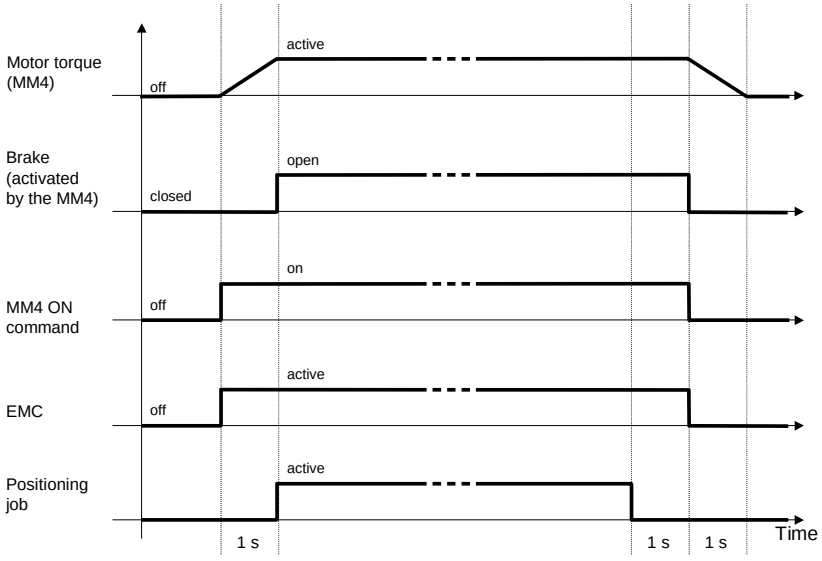


Figure 5-23 Positioning signal patterns

Comparable patterns occur during switch off. In that case the inverter and the closed-loop positioning control are only switched off one second after the end of the positioning process.

Part B: Installation of the Sample Application

Content of Part B

Part B is a step-by-step guide to setting up and commissioning the application.

Objective of Part B

Part A1 of this document is meant

- to enable the reader to install the example with all the hardware/software components
- to show the reader how to operate the application

Note

You can find further information and step-by-step instructions to

- parameterizing the MICROMASTER
- parameterizing the CPU 314C–2 DP
- parameterizing Easy Motion Control
- installing and parameterizing the SIMATIC NET OPC server

as a supplement to Part B of the document in the document entitled "Closed-Loop Controlled Positioning of an Axis with SIMATIC CPU 314C-2 DP, MICROMASTER 440 and SIMATIC Easy Motion Control, **Parameterizations**".

Topics covered:

6 Installation of the Hardware and Software

You MUST follow the safety tips in the Operating Instructions for the MICROMASTER .

Warning

The inverter conducts hazardous voltages and controls rotating mechanical parts which may be dangerous. Failure to heed warnings or to follow the advice given in these instructions may result in death, serious injury or considerable damage to property.

Only personnel with appropriate training may work on this device. They must be familiar with all the safety tips and installation, operation and maintenance procedures which are contained in these instructions. In order to be operated safely and in perfect working order, the device must be transported, installed, operated and maintained properly.

Risk of electric shock. The capacitors in the DC intermediate circuit remain charged for 5 minutes after the power supply has been switched off. Consequently, the device may not be opened for 5 minutes after the power supply has been switched off.

Caution

Children and unauthorized persons must be kept well away from the device!

The device may only be used for the purpose specified by the manufacturer. Unauthorized modifications and the use of spare parts and accessories which are not distributed or recommended by the device manufacturer may cause fires, electric shocks and injuries.

Warning

MICROMASTER inverters work at high voltage. When electrical devices are operated, some of the parts in these devices carry dangerous levels of voltage.

Emergency stop devices conforming to EN 60204 IEC 204 (VDE 0113) must remain functional in all the controller's operating modes. If the emergency stop device is reset, this must not lead to uncontrolled or undefined restarts.

Additional procedures or equipment must be incorporated in situations where short circuits in the controller may lead to considerable damage or even serious injuries (i.e. potentially hazardous short circuits) in order to guarantee or make definite that the system can be operated safely, even if a short circuit occurs (e.g. independent limit switches, mechanical locks, etc.).

Certain parameter settings could cause the inverter to start again automatically after the voltage supply has failed.

6.1 Hardware Setup

For details of the components contained in the hardware setup, refer to Section 3.1 Applied Solution: Central Controller with a CPU314C + MICROMASTER 440 + an Asynchronous Motor

Procedure

! **WARNING**

Only turn on the voltage supply to the hardware components once the steps specified in the table below have been completed.

Hardware installation procedure:

Step	Focus	Action
1	Central rack	Arrange the following hardware components on the rack from left to right: PS 307, CPU 314C-2 DP
2	MM440	Assemble the MM440 on the network filter
2	Encoder	Assemble the encoder on the motor axis
3	230V cables	Wire the power cable in accordance with Table 6-1 230 V Wiring Table
4	Signal cables	Wire the signal cables in accordance with Table 6-2 Signal Wiring Table
5	MPI cable	Wire the communication cables in accordance with Table 6-3 Communication Wiring Table
6	Switching on	Establish the power supply for all the components.

Wiring Diagram

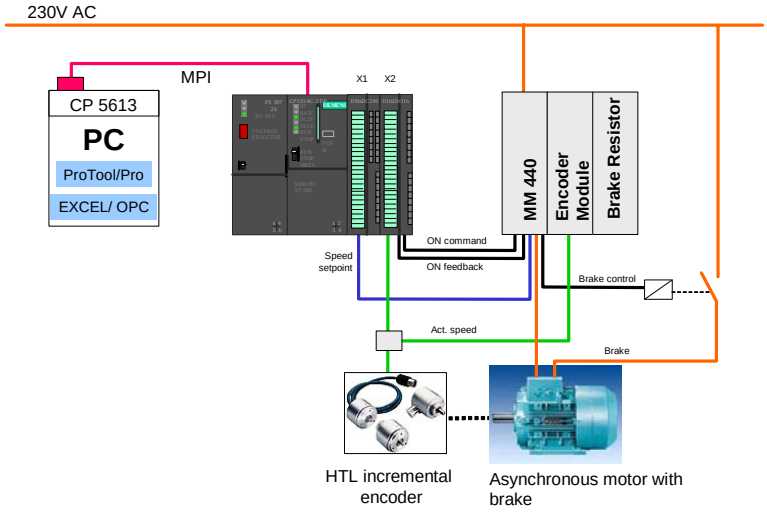


Figure 6-1 Overview

Table 6-1 230 V Wiring Table

Signal	PS307	MM440	Motor	Braking contactor
230V AC –P	PS307: L1	L		1/L1
230V AC –N	PS307: N	N		3/L2
PE	PE	PE	PE	
Inverter output		U	U1	
Inverter output		V	V1	
Inverter output		W	W1	
Holding brake P			~	2/T1
Holding brake N			~	4/T2

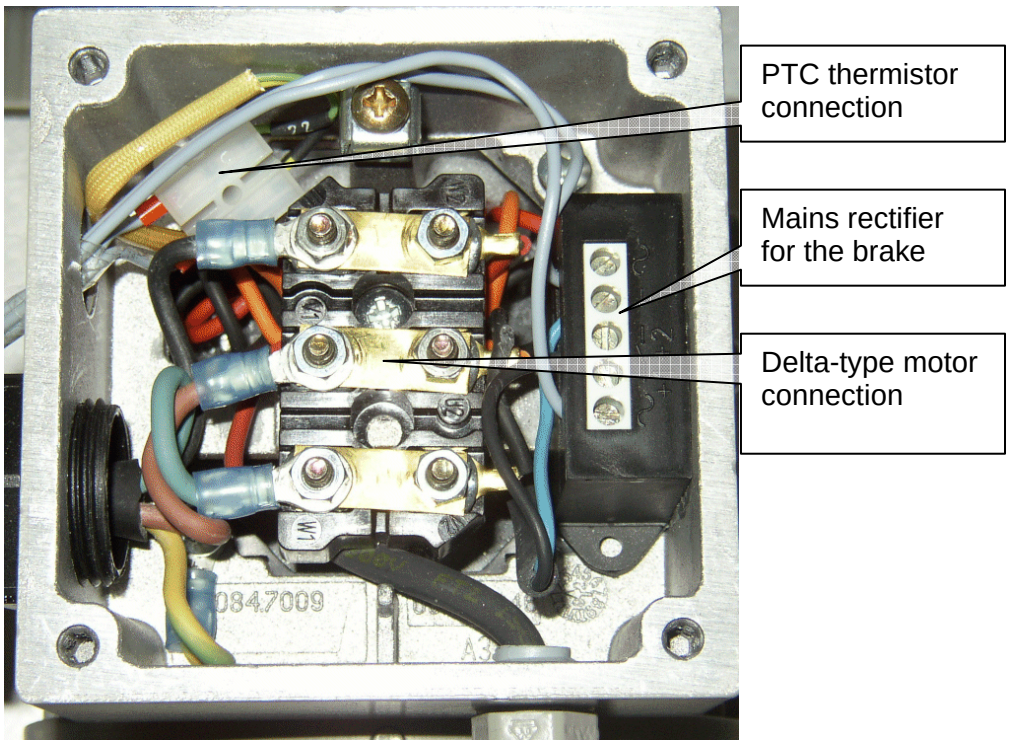


Figure 6-2 Motor terminal block

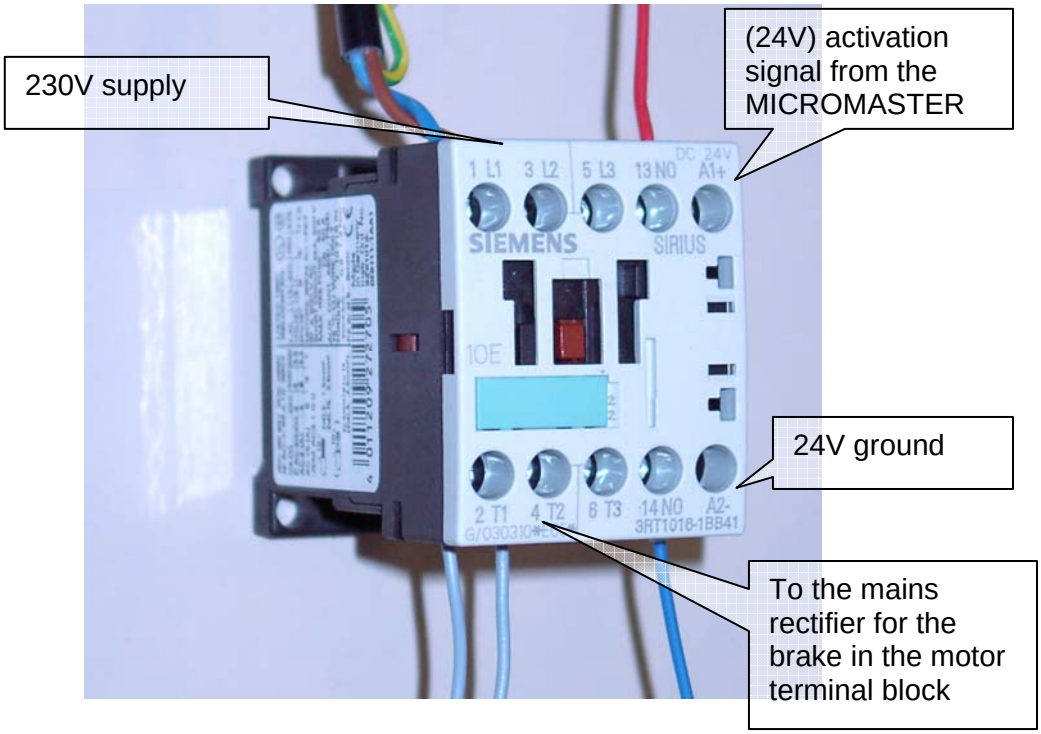


Figure 6-3 Braking contactor

Table 6-2 Signal Wiring Table

Signal	**CPU314C-2DP PS307**	**MM440**	**Motor**	**Brake contactor**	**Encoder (Pin)**
Setpoint speed +	X11: 16	3			
Setpoint speed -	X11: 20	4			
24V	L+ X12: 1, 21, 31 PS307: L+	20, 25			12
Ground	M X12: 20, 30, 40 PS307: M	28		A2-	10
On command	X12: 22	5			
On feedback	X12: 14	19			
Braking activation		24		A1+	
Encoder, A	X12: 2	EM: A			8
Encoder, NA		EM: NA			1
Encoder, B	X12: 3	EM: B			5
Encoder, NB		EM: NB			6
Lower limit switch	Jumper: X12: 32 - X12: 12				
Upper limit switch	Jumper: X12: 33 - X12: 13				
Reference point switch	Jumper: X12: 34 – X12: 16				

As there are several versions of the encoder cable, the configuration must be measured.

Table 6-3 Communication Wiring Table

Signal	CPU314C-2DP	MM440	PC/PG
MPI	MPI		MPI / DP
USS		PC connection kit	COM1

DIP switches on the MICROMASTER

Ensure that the AIN1 switch on the MICROMASTER is set to OFF to enable it to work as a voltage input.

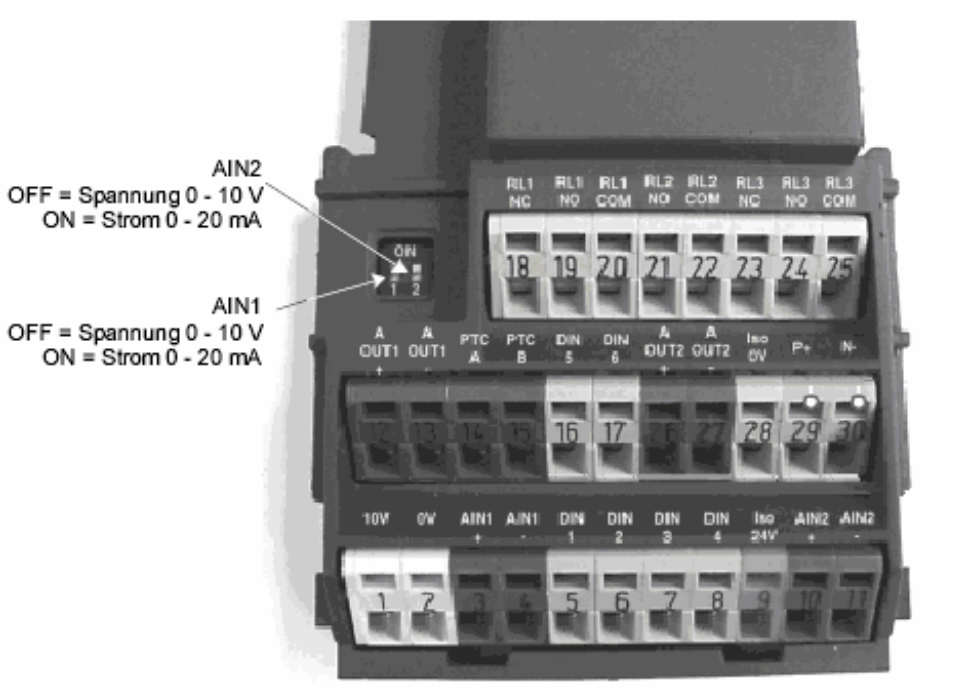


Figure 6-4 Selection function, analog input

6.2 Installation of SIMATIC Standard Software

Minimum version requirements

The following versions have been used:

Table 6-4 Versions

Component	Version
STEP 7	5.2
ProTool/Pro	6.0 + SP2
Easy Motion Control	2.0
Starter (alternative to DriveES)	3.0
DriveES (alternative to the starter)	5.2
SIMATIC NET	6.0 or 6.1 depending on the operating system
CPU314C-2DP (Firmware)	2.0

Installation

Install

- Step 7
- Easy Motion Control V2
- ProTool/Pro (with the **Integration in Step 7** option)

and optionally

- SIMATIC NET
- DriveES or STARTER

In order to do this, follow the instructions for the installation programs.

6.3 Adapting and Loading the Application Software Modules

6.3.1 Dearchive the STEP 7 Project

Start the SIMATIC Manager, dearchive the MC_EMC project from the MC_EMC.zip file and open it.

Note

The parameter lists for the MICROMASTER has also been stored in the project with DriveES. Consequently, if you do not have DriveES installed, a message appears stating that the DriveES and SIMOTION option packages are not installed.

As this has no other effect, simply acknowledge the message.

Table 6-5 Application Software

User Program	Function	Source
Control program in the statement list	Store management and positioning the stroke axis on the basis of Easy Motion Control	MC_EMC.zip
ProTool/ Pro	HMI user interface	
MICROMASTER configuration data (for DriveES)	Configuration data for the MICROMASTER“ in "Vector Control" mode	
MICROMASTER configuration data (for Starter)	Configuration data for the MICROMASTER in "Vector Control" mode	MM440_Starter.zip
EXCEL worksheet	Recording the actual position and actuating signal for the current position	MeasureData.xls

Block Lists

Table 6-6 List of DBs

Name	Functional Description	Technical Data
DB1	Instance data block for FB1	Size in the work memory: 112 bytes Size in the load memory: 590 bytes
DB3	Instance data block for FB3	Size in the work memory: 110 bytes Size in the load memory: 568 bytes
DB4	Instance data block for FB4	Size in the work memory: 104 bytes Size in the load memory: 558 bytes
DB5	Instance data block for FB5	Size in the work memory: 70 bytes Size in the load memory: 470 bytes
DB11	Instance data block for FB11	Size in the work memory: 58 bytes Size in the load memory: 450 bytes
DB28	Instance data block for FB28	Size in the work memory: 128 bytes Size in the load memory: 638 bytes
DB34	Instance data block for FB34	Size in the work memory: 52 bytes Size in the load memory: 444bytes
DB50	Instance data block for FB50	Size in the work memory: 40 bytes Size in the load memory: 94 bytes
DB60	Instance data block for FB60	Size in the work memory: 42 bytes Size in the load memory: 92 bytes
DB61	Instance data block for FB61	Size in the work memory: 42 bytes Size in the load memory: 96 bytes
DB62	Instance data block for FB62	Size in the work memory: 56 bytes Size in the load memory: 136 bytes
DB65	Instance data block for FB65	Size in the work memory: 54 bytes Size in the load memory: 150 bytes
DB66	Instance data block for FB66	Size in the work memory: 54 bytes Size in the load memory: 118 bytes
DB101	Instance data block for FB1	Size in the work memory: 112 bytes Size in the load memory: 590 bytes
DB100 Axis DB,	Axis DB, contains all the data on the storage lift axis	Size in the work memory: 318 bytes Size in the load memory: 362 bytes
DB200 DB stock	DB stock, contains the store assignment	Size in the work memory: 608 bytes Size in the load memory: 676 bytes
DB202	Measurement DB, contains measured values	Size in the work memory: 2442 bytes Size in the load memory: 2520 bytes
DB203	Measurement DB, contains measured values	Size in the work memory: 2442 bytes Size in the load memory: 2520 bytes

Table 6-7 List of Easy Motion Control FBs und FCs used

Name	Functional Description	Technical Data	Sequence environment
FC0	Initializes the EMC blocks during restart	Size in the work memory: 1086 bytes Size in the load memory: 1482 bytes Creation language: SCL	OB35
FB1	Motion FB for positioning the axis	Size in the work memory: 3924 bytes Size in the load memory: 4610 bytes Creation language: SCL	OB35
FB3	Motion FB for moving the axis manually	Size in the work memory: 3110 bytes Size in the load memory: 3706 bytes Creation language: SCL	OB35
FB4	Motion FB for the search for reference for the axis or for setting a reference point	Size in the work memory: 2886 bytes Size in the load memory: 3480 bytes Creation language: SCL	OB35
FB5	Motion FB for stopping motion	Size in the work memory: 1114 bytes Size in the load memory: 1574 bytes Creation language: SCL	OB35
FB11	FB for the position controller	Size in the work memory: 1756 bytes Size in the load memory: 2234 bytes Creation language: SCL	OB35
FB28	FB for evaluating the counter in the CPU 314C and, thus, for evaluating the encoder information.	Size in the work memory: 1476 bytes Size in the load memory: 2034 bytes Creation language: SCL	OB35
FB34	FB for outputting the setpoint speed via an analog output on the CPU 314C	Size in the work memory: 356 bytes Size in the load memory: 764 bytes Creation language: SCL	OB35
UDT1	User data type for creating an axis DB	-	PG

Table 6-8 List of the application FBs

Name	Functional Description	Technical Data	Sequence environment
FB50 FB io	Signal allocation block for I/Os (except	Size in the work memory: 100 bytes Size in the load memory: 164 bytes Creation language: STL	OB35
FB60 FB sync	Block with a sequencer for synchronizing the axis	Size in the work memory: 732 bytes Size in the load memory: 896 bytes Creation language: STL	OB35
FB61	Block with a sequencer for	Size in the work memory: 272 bytes	OB35

Name	Functional Description	Technical Data	Sequence environment
FB jog	jogging the axis	Size in the load memory: 404 bytes Creation language: STL	
FB62 FB pos	Block with a sequencer for positioning the axis	Size in the work memory: 734 bytes Size in the load memory: 928 bytes Creation language: STL	OB35
FB65 FB auto	Block with a sequencer for operating the storage lift in automatic mode	Size in the work memory: 1488 bytes Size in the load memory: 1782 bytes Creation language: STL	OB35
FB66	Block for saving the measured values	Size in the work memory: 224 bytes Size in the load memory: 304 bytes Creation language: STL	OB35

Table 6-9 List of the OBs

Name	Functional Description	Technical Data
OB35	Watchdog interrupt OB, for the equidistant calling of the FBs	Size in the work memory: 1094 bytes Size in the load memory: 1188 bytes Creation language: STL
OB100	Ramp-up OB, for initializing	Size in the work memory: 106 bytes Size in the load memory: 178 bytes Creation language: STL

Table 6-10 List of other blocks

Name	Functional Description	Seq. env.
man_pos	Value table for manual positioning	PG
man_synchronization	Value table for synchronization	PG
MC__Home	Value table for MC_Home	PG
MC__Jog	Value table for MC_MoveJog	PG
MC__MoveAbsolute	Value table for MC_ MoveAbsolute	PG
MC__MoveAbsolute2	Value table for both instances of the MC_ MoveAbsolute for monitoring job switches and job replacements	PG
mode	Value table for checking mode switches	PG
OPC_Test	Value table for the OPC server test	PG
signal_check	Value table for switching on the drive when the signal is on	PG
stock_db_view	Value table for	PG
UDT3	UDT for the stock DB	PG

6.3.2 Copying the Easy Motion Control Blocks

Table 6-11 Copying the Easy Motion Control blocks

Step	Action
1	Open the **EMC2 Easy Motion Control** library: Open the **File** menu and then **Open**. Select the **Libraries** page and select **EMC2 Easy Motion Control**.
2	Copy the following blocks into the CPU314C's block container: • FC0 MC_Init • FB1 MC_MoveAbsolute • FB3 MC_MoveJog • FB4 MC_Home • FB5 MC_StopMotion • FB11 MC_Control • FB28 EncoderCPU314C • FB34 OutputCPU314C While the blocks can normally be renamed, this is not possible in the same application because the instance DBs cannot be used.

6.3.3 Loading the STEP 7 Project

- Open the S7 project MC_EMC using the SIMATIC Manager.
- Delete all the data on the MMC (delete all the CPU blocks in online mode) and reset the CPU.
- Load the program onto the CPU314C.
- If you use SIMATIC NET, follow the instructions below in **Adding and Parameterizing the PC-Station**

6.3.4 Adding and Parameterizing the PC-Station

If you have not installed SIMATIC NET or if you do not wish to use it for the time being, you can skip this section, in which case you cannot import any measured values into EXCEL.

The idea behind the PC-Station is to integrate the PG/PC into the automation network as an independent component, such as a controller.

The SIMATIC PC-Station consists of several components. The configuration is defined with the component configurator.

It corresponds to inserting modules into a mounting rack. The component configurator corresponds to the rack, and the applications and CP(s) (e.g. CP5611) correspond to the modules.

Consequently, all applications that run in-service on the station have to be entered in the component configurator, too.
In this application, it is the OPC server and ProTool/Pro RT.

STEP7 and ProTool/Pro CS are not configured because they are configuration software but are not used in-service.

In addition, the CP(s) (e.g. CP5611) are configured which is/are used to connect the SIMATIC PC-Station to the plant.

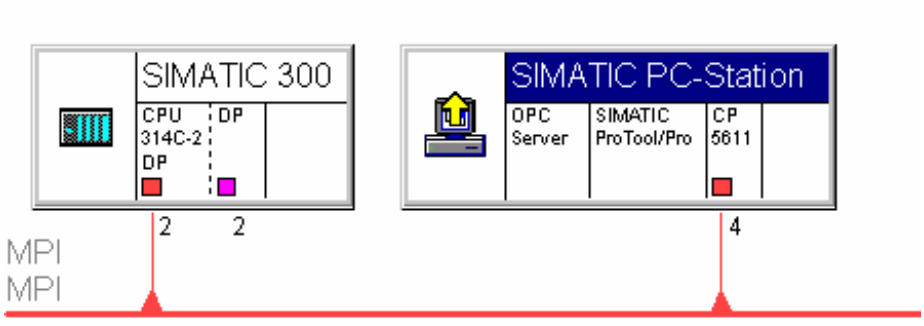


Figure 6-5 SIMATIC PC-Station and applications

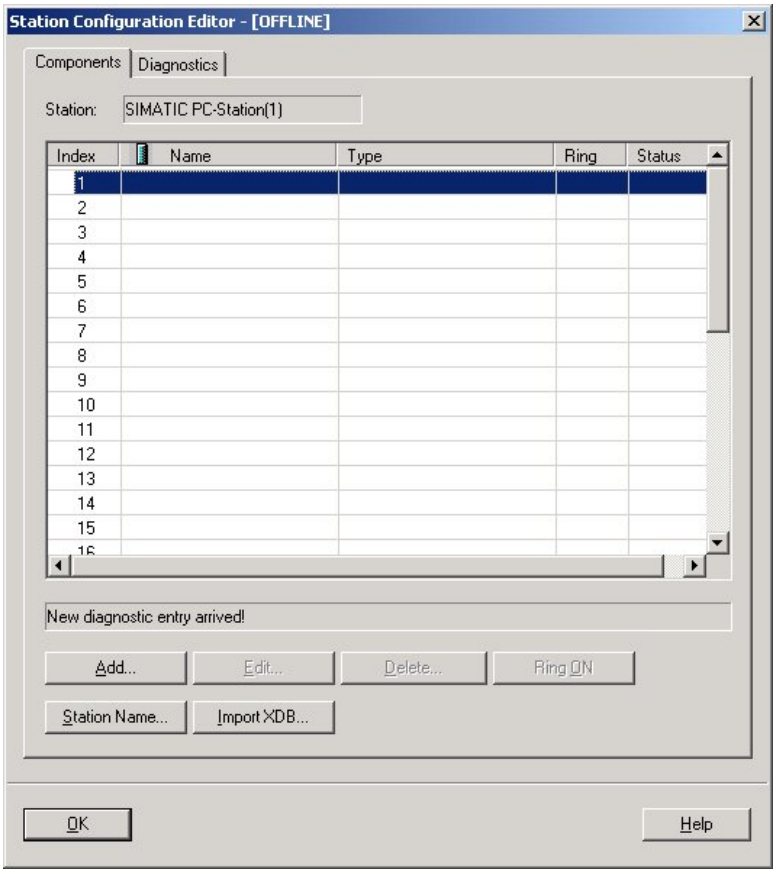
As different CPs can be used and the version of the OPC server is also operating system-dependent, these components are not contained in the archived project.

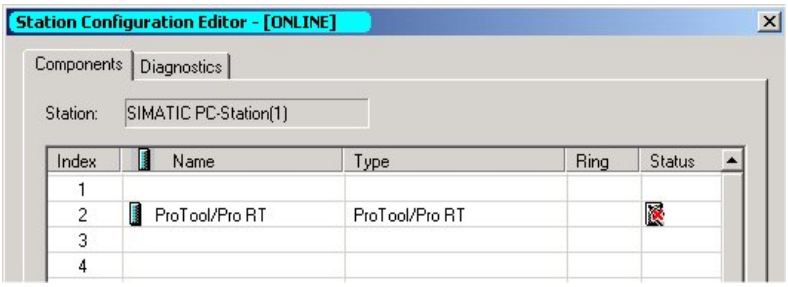
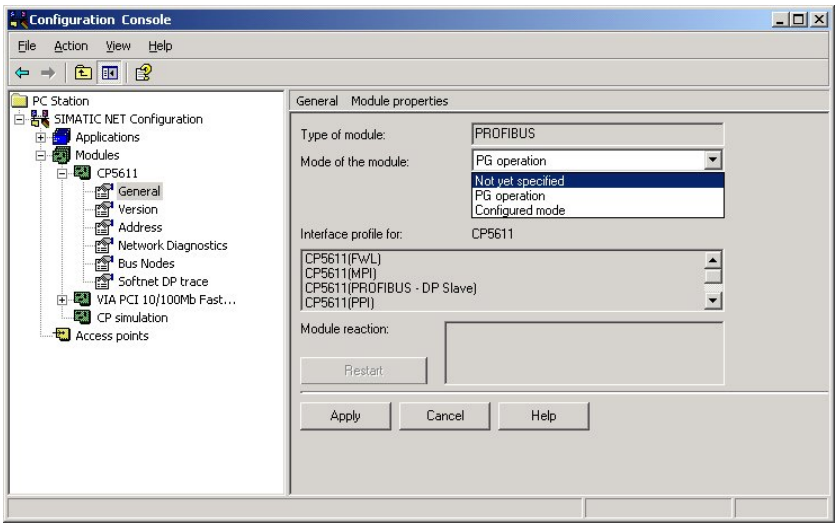
These components have to be integrated into the existing project with the aid of the SIMATIC NET Commissioning Wizard!

Note

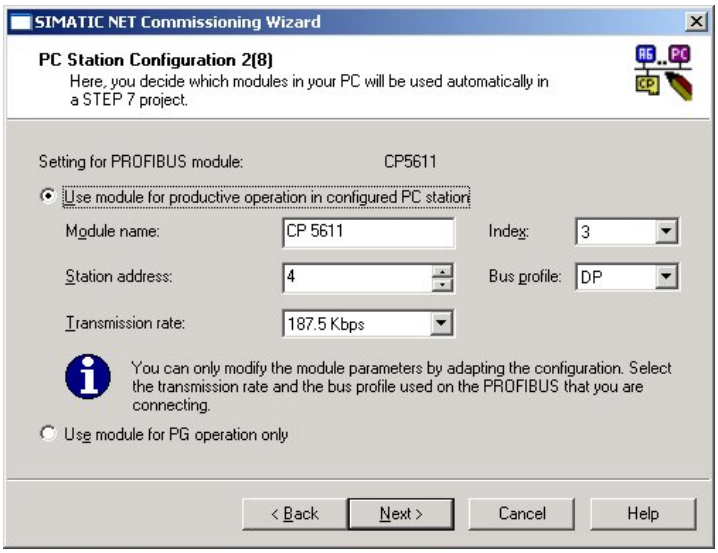
The PC-Station is reconfigured. All the existing configuration data is deleted.

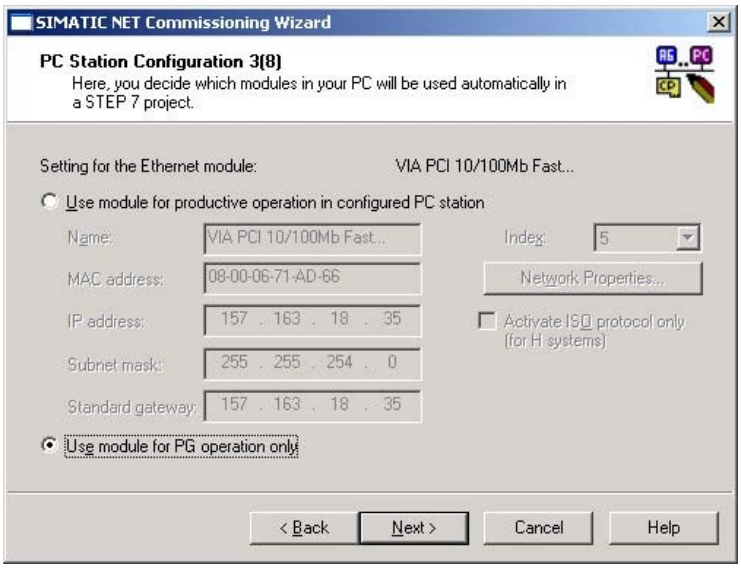
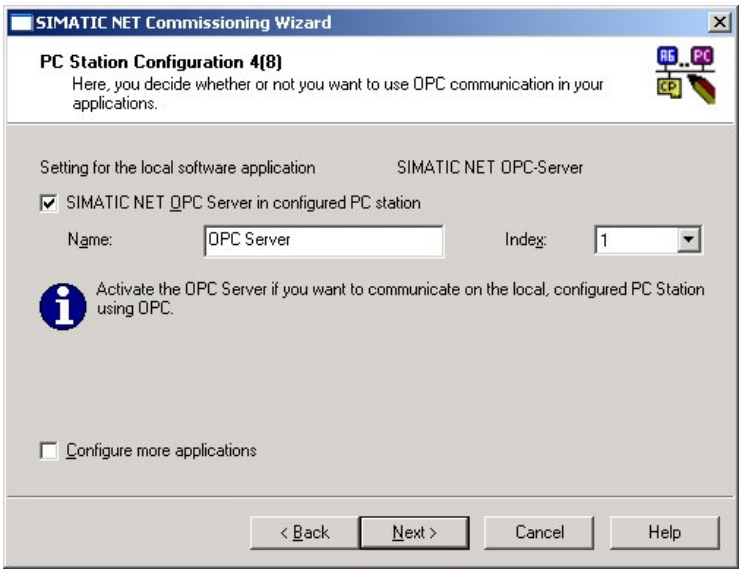
Table 6-12 OPC connection setup

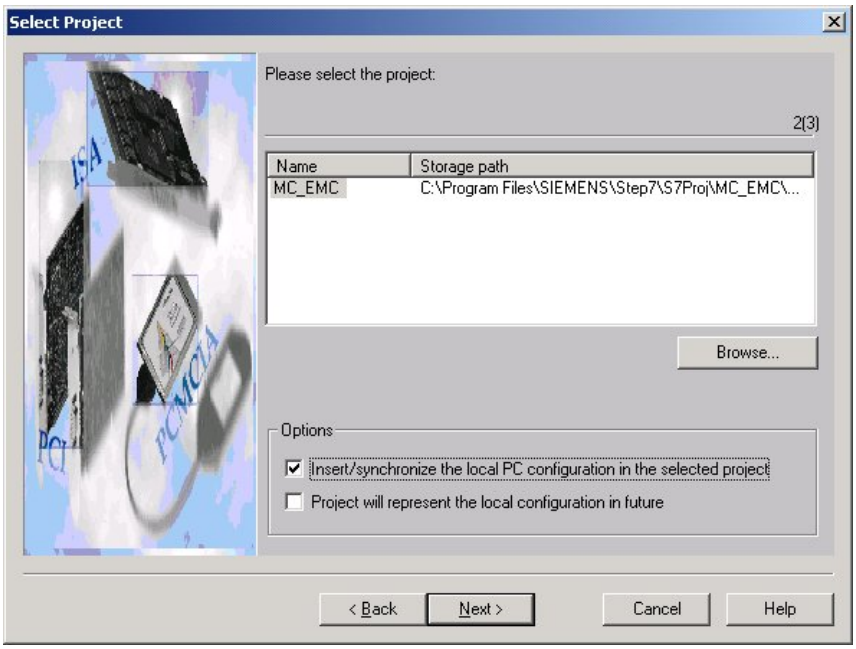
Step	Action
1	The ProTool/Pro RT programming has to be inserted first because it is already contained in the project. If there are any other components contained there, please delete them. Open the **Station Configuration Editor** program via the system tray.  Figure 6-6 Open the Station Configuration Editor After double-clicking on this icon, the Station Configuration Editor is displayed: Click on station name and ensure that the station is called **SIMATIC PC-Station(1)**, as in STEP7 Project MC_EMC.  Figure 6-7 Station Configuration Editor

Step	Action
2	Insert ProTool/Pro programming into the configuration in slot 2: • Click on the **Add...** button • Select index **2** • Select **ProTool/Pro RT** in the "Type" field • Assign it a name, e.g. HMI • Click on **OK** to confirm • Close the Station Configuration Editor with **OK**  Figure 6-8 Paste HMI
3	If you wish to run through the SIMATIC NET Wizard completely, there may not be any configuration in SIMATIC NET. In order to do this, open the **Configuration Console** program via the Start menu: Start, SIMATIC, SIMATIC NET and Settings. Set the mode for all CPs to **Not yet specified** and save this by clicking on **Apply.** Then close the program.  Figure 6-9 Delete the CP configuration

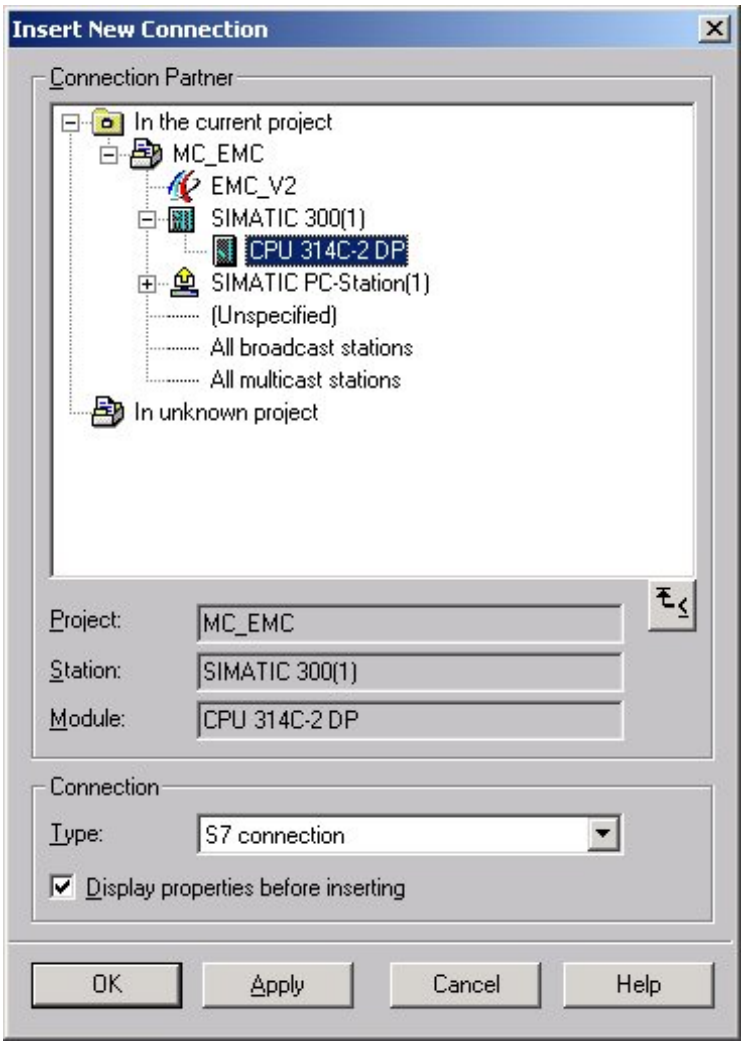
Step	Action
4	Now start the SIMATIC NET Commissioning Wizard: • If you are using the CP5511, insert it. The Commissioning Wizard appears. • If you use the CP5611 (PG) or if the Wizard has not started automatically, open the **Commissioning Wizard** program via the Start menu: Start, SIMATIC, SIMATIC NET and Settings • Click on **Next**  Figure 6-10 CP configuration If "Welcome to the SIMATIC NET Commissioning Wizard **1 (4)**" is displayed, there is configuration data present. Perform step 2 first. If "Welcome to the SIMATIC NET Commissioning Wizard **1 (7)**" is displayed, there is only one CP present. Then skip the configuration of the Ethernet CP (step 6). More than 8 steps are displayed if there are more than 2 CPs installed. Follow the instructions in step 5 for the CP that you wish to use in the application. Follow the instructions in step 6.

Step	Action
5	A configuration dialog appears for each of the installed CPs. Check which CP it involves. Follow the instructions in step 6 for an Ethernet CP or for a network card. Set **productive operation** and **Index 3** for the CP5511 or 5611 that you wish to use for this application. The other parameters are irrelevant. Once you have configured the CP, click on **Next**.  Figure 6-11 Configuration of the CP5611

Step	Action
6	Check which CP it involves. Follow the instructions in step 5 for a PROFIBUS CP. Enable 'Use module for PG operation only' in the case of Ethernet CPs or network cards. They are then not available in SIMATIC NET. Click on **Next.**  Figure 6-12 Configuration of other CPs
7	Insert a SIMATIC NET OPC server at Index **1** and the click on **Next**.  Figure 6-13 Insert an OPC server

Step	Action
8	Save the module configuration by clicking on **Next**. Confirm the message that the existing database will be lost. The OPC server and the CP are therefore now inserted into the Configuration Editor and, thus, into the SIMATIC PC station.
9	Once the PC station "hardware" has been "set up", the configuration in the STEP7 project is now defined. Click on the **PC Station Wizard...** button.
10	If a message appears stating that the access point (the PG/PC interface) is being switched to **PC-internal**, confirm this.
11	If a message appears stating that the PC station name does not match the configuration, copy the proposed station in the selection box.
12	Select **Edit saved configuration** in the PC Station Wizard and click on **Next**.
13	Select the **MC_EMC** project. If necessary, navigate to the project via **Browse**. Check the **Insert/synchronize the local PC configuration...** option and uncheck the **Project will represent the local configuration…** option. Click on **Next**. 
14	Confirm the message that that the configuration is being balanced with the Station Manager (corresponding to the Station Configuration Editor). Consequently, the components of the SIMATIC PC station are entered in the HW Config.

Step	Action
15	Now select **Change hardware configuration** and the PC station name **SIMATIC PC-Station(1)** and click on **Finish**. The HW Config opens with the configuration of the SIMATIC PC station.
16	Open the Object Properties for the CP5611 (or for the CP5511) and change the interface type from **PROFIBUS** to **MPI.** Confirm that you wish to network with MPI. Select **Properties ...**,select the existing MPI network and click on **OK** to confirm. Click on **OK** to exit the object properties dialog for the CP5x11. Save and compile the configuration and close the HW Config.
17	Click once again on the **PC Station Wizard...** button in the SIMATIC NET Commissioning Wizard.
18	Select the **MC_EMC** project. If necessary, navigate to the project via **Browse**. Check the **Insert/synchronize the local PC configuration...** option and uncheck the **Project will represent the local configuration…** option. Click on **Next**.
19	Now select **Edit network and connection configuration** and the PC station name **SIMATIC PC-Station(1)** and click on **Finish**. NETPRO opens.

Step	Action
20	Highlight the OPC server in the SIMATIC PC station. Select **Insert New Connection** via the context-sensitive menu (right-click the mouse) for the OPC server. Check that the CPU 314C-2 DP is highlighted and that the S7 connection is selected.  Figure 6-14 Text Click on **OK** to confirm.

Step	Action
21	Check that the local ID is "S7 connection_1" and click on **OK** to confirm.

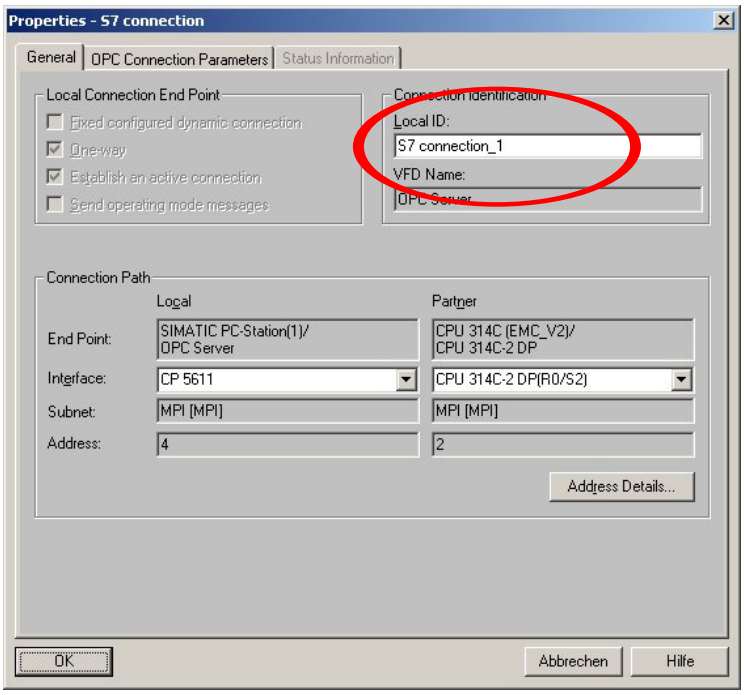


Figure 6-15 Create an S7 connection

This name is also saved in the EXCEL program. EXCEL requests data from the OPC server via this connection. If the connection name saved in EXCEL does not match the configured connection name, no data is transmitted.

Save and compile the configuration.

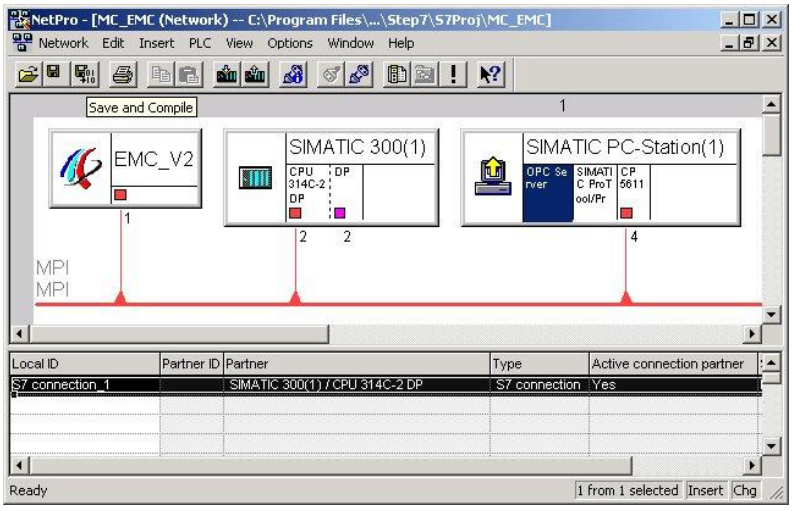
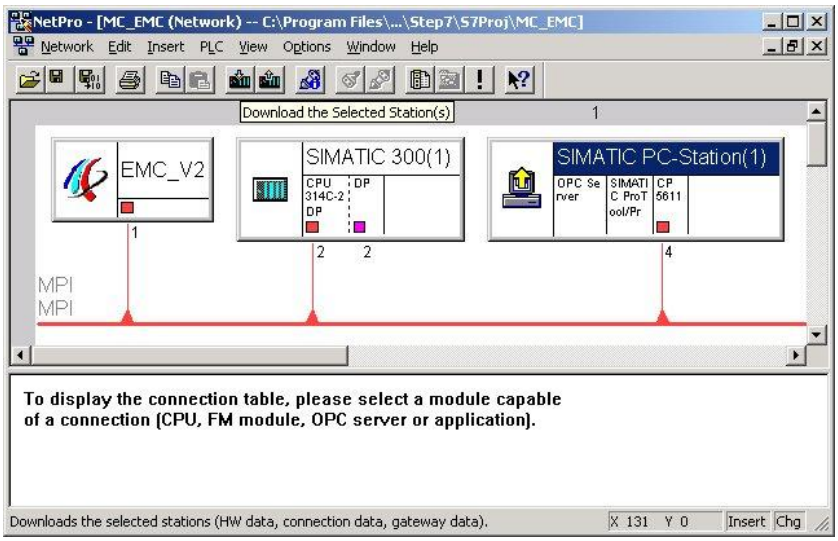


Figure 6-16 SIMATIC PC station with an OPC server and S7 connection

Step	Action
22	Select the SIMATIC PC station and download the configuration.  Figure 6-17 Download the SIMATIC PC station Also repeat this with the CPU 314C.
23	Highlight the SIMATIC 300 and download the configuration.
24	Close NET Pro and return to the SIMATIC NET Commissioning Wizard. Click on **Next**.
25	Select **Use no symbol file** and click on **Next**.
26	Click on **Finish**. Open the configuration console with the "Set PC station" option. Close the configuration console.
27	The configuration of the SIMATIC PC station is now ended.

Note

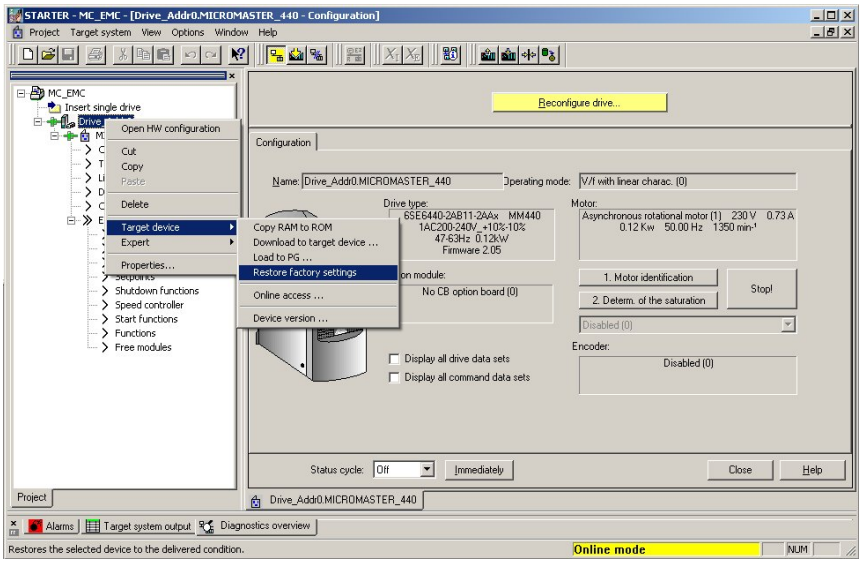
The document entitled "Rapid/Creep Speed Positioning via Integrated Technology Functions of the CPU 314C, **Parameterizations**" contains a guide to checking the function of the OPC connection.

6.3.5 Downloading the MICROMASTER Parameterization Data

Downloading the parameter set only makes sense if you have connected a 1LA7063-4AB10 motor. Otherwise, follow the guide in the document

entitled "Closed-Loop Controlled Positioning of an Axis with SIMATIC CPU 314C-2 DP, MICROMASTER 440 and SIMATIC Easy Motion Control, **Parameterizations**".

Table 6-13 Downloading MICROMASTER with a parameter set

Step	Action
1	If you are using the Starter (stand-alone), start it via the icon and open the MICROMASTER project. If you are using DriveES, start the Starter by double clicking on the drive icon in the SIMATIC Manager.
2	Set the PC/PG interface to **USS**.
3	Go online.
4	Restore the factory settings:  Figure 6-18 Restore the factory settings
5	Download the configuration to the MICROMASTER:  Figure 6-19 Download project

Note

After downloading the parameter set, you should

- check the encoder and motor polarity and
- optimize the MICROMASTER speed controller in order to adapt this to suit the prevailing conditions.

Follow the instructions for this in the document entitled "Closed-Loop Controlled Positioning of an Axis with SIMATIC CPU 314C-2 DP, MICROMASTER 440 and SIMATIC Easy Motion Control, **Parameterizations**"

- Section 3.8 from step 4 and
- Section 3.9

6.3.6 Commission Easy Motion Control

Some parameters in Easy Motion Control need to be adapted during commissioning because they are dependent on the mechanical or electrical setup, such as controller gain and the encoder polarity.

6.3.6.1 Wiring Test

In the wiring test, the motor and encoder polarity are checked, and appropriate switches are made in the axis DB.

Note

The drive is normally set up to rotate clockwise with a positive setpoint. The polarity is defined looking from the load machine.

Figure 6-20 Motor rotating clockwise

- Retrieve the application from the SIMATIC Manager and load it into the controller.
- Open the OB35 in the LAD/FBD/STL editor.
- Enable the **BEA** command in the 2nd line of network 3 by deleting the comment characters (***//***). Load the changed OB35 into the controller.
- Call up the DB100 axis DB with the Easy Motion Control software by double-clicking on the DB100 in the SIMATIC Manager.
- Select **Commissioning** and the **Wiring test**.
- Start the drive with the aid of the **signal_check** variable block by setting **"idb_io".Drive_enabled** to **1** or **true**.

Note

The motor may start to turn slowly on account of the analog setpoint transmission!

- Follow the Wizard's instructions.

Figure 6-21 Wiring test

- Stop the drive with the aid of the **signal_check** variable block by setting **"idb_io".Drive_enabled** to **0** or **false**.
- Save the axis DB and load it into the controller.

Note

If the drive starts turning slowly after being released, define the offset compensation, see Section 6.3.6.2 Offset Compensation.

- Disable the **BEA** command in the 2nd line of network 3 by deleting the comment characters (*//*) at the start of the line. Save the changed OB35 und and load it into the controller.

6.3.6.2 Offset Compensation

If the drive starts turning slowly after being released, define the offset compensation:

- In order to do this, start the drive once again via the variable table and call up the wiring test, as described in Section 6.3.6.1 Wiring Test (disable the instruction in OB35, etc.)
- Now, slide the elevator bar to select the setpoint until the drive comes to a stop with the move button pressed.
- Take a note this value and stop the wiring test.
- Enter this value into the offset compensation field on the Encoders/Controller/Motor tab of the Easy Motion Control parameterization software.
- Disable the **SPA END** jump command in the 2nd line of network 3 by deleting the comment characters (***//***) at the start of the line. Save the changed OB35 und and load it into the controller.

Note

The offset compensation does not take effect while the output driver is not factored in, e.g. in the wiring test; consequently, it cannot be checked.

If the output driver is factored in, the closed-loop position control is also activated; it automatically also compensates for an offset, which also means that the offset compensation cannot be checked.

However, if an offset has already been compensated for via the offset compensation, this does not have to be done by the closed-loop position controller, improving the control performance.

6.3.6.3 Closed-Loop Position Control Optimization

The optimum controller gain can be determined experimentally at the axis.

- Call up the DB100 axis DB with the Easy Motion Control software by double clicking on the DB100 in the SIMATIC Manager.
- Select the **Encoders/Controller/Motor** tab.
- Move the axis with the aid of the HMI, e.g. by jogging (see Section Jog the Axis).
- Raise the controller gain in increments of 1.0 until the axis starts to fluctuate in motion or at a standstill.
- If this happens, reduce the controller gain until there is no further tendency to fluctuate visible.
- Save the axis DB and load it into the controller.

6.4 Parameter Lists

6.4.1 Parameters to be Configured in the MICROMASTER 440

Table 6-14 Parameters changed in the MICROMASTER

Parameter	Description	Value	Meaning
P 300 to P 311	*Motor data according to the motor rating plate*		
P 400	Encoder type selection	2	Increm. encoder, 2 tracks
P 408	Encoder pulses per revolution	1000	1000 pulses / rev
P 491	Speed loss response	1	Change to SLVC
P 700	Command source	2	Terminal strip
P 733	Digital output 3 function	52.C	Holding brake activated
P 756	Analog input 1 ADC type	4	± 10V
P 757	X1 value, ADC scaling	-10	-10V
P 758	Y1 value, ADC scaling	-100	-100%
P 1120	Ramp-up time 1	0	0s
P 1121	Ramp-down time 1	0	0s
P 1135	AUS3 return time	0	0s
P 1215	Release motor holding brake	1	Released
P 1237	Resistance braking	1	5% duty cycle
P 1240	VDC controller	0	Blocked
P 1300	Controller type	21	Vector controller with encoder

Important

Once parameters have been assigned to the MICROMASTER, the motor identification procedure should be performed so as to enable the MICROMASTER settings to be adapted to the application. If possible, the closed-loop speed control optimization should be performed with a load machine.

6.4.2 Parameters to be Configured in Easy Motion Control

Parameters are best assigned to the Easy Motion Control blocks via the user interface (supplied). The following parameters are entered for the application:

Table 6-15 Easy Motion Control parameters

Parameter	Description	Value	Meaning
196.0 LengthUnit	Unit of length	mm	
202.0 InputModuleType	Input driver	CPU314C	
58.0 InputModuleInAddr	Input driver, input address	768	In accordance with the setting in the HW Config
58.0 InputModuleOutAddr	Input driver, output address	768	In accordance with the setting in the HW Config
62.0 InputChannelNo	Input driver, channel	0	
228.0 OutputModuleType	Output driver	CPU314C	
80.0 OutputModuleOutAddr	Output driver, address	752	In accordance with the setting in the HW Config
84.0 OutputChannelNo	Output driver, channel	0	
4.0 AxisType	Axis type	Linear axis	
0.0 Sample_T	Sample time	0.25	Is entered automatically
10.0 AxisLimitMin	SW limit switch – start	-500 mm	
6.0 AxisLimitMax	SW limit switch – end	13000 mm	
14.0 MaxVelocity	Max. axis velocity	1000 mm/s	
100.0 Override	Velocity override	100%	
18.0 MaxAcceleration	Max. axis acceleration	450 mm/s^2	
22.0 MaxDeceleration	Max. axis deceleration	450 mm/s^2	

Parameter	Description	Value	Meaning
64.0 StepsPerRev	Encoder, steps per revolution	4000	Quadruple evaluation
68.0 DisplacementPerRev	Axis displacement per encoder revolution	471.21 mm	
74.0 PolarityEncoder	Encoder direction adaptation		Is determined during commissioning
44.0 FactorP	Controller, controller gain	18	Is determined during commissioning
48.0 ManVelocity	Setpoint velocity in the HB	10 mm/s	
88.0 DriveInputAtMaxVel	Motor, reference for max. axis velocity	10 V	
92.0 OffsetCompensation	Offset compensation	0	Is determined during commissioning
86.0 PolarityDrive	Drive direction adaptation		Is determined during commissioning

7 Application Controls

The application controls are very simply and clearly laid out using a ProTool/Pro user interface. There are three masks in the operator panel:

- Manual mode
  Mode for moving the storage lift by hand and for synchronizing after switch-on.
- Automatic mode
  Mode for charging and discharging the boxes.
- Store
  Diagram of the store, showing the current stock.

7.1 Requirements

The conditions defined in the section on the "Installation of the Hardware and Software" (Section 6) must be complied with. All the devices must be switched on, and the parameterization or program must be loaded.

7.2 Start the HMI Application

If you are using SIMATIC Net, start the operator panel via the SIMATIC Manager, **SIMATIC PC-Station**, **SIMATIC ProTool/Pro RT** and **EMC_V2**. Then right-click on **EMC_V2** and select **Start Runtime**.

The OP programming at project level (directly under MC_EMC) is only suitable for use without SIMATIC NET.

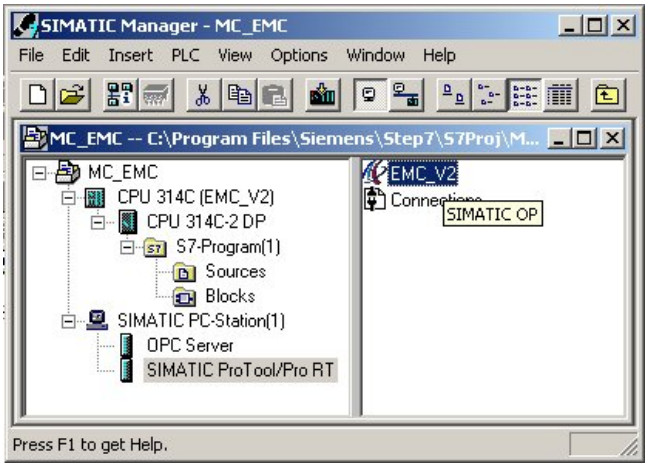


Figure 7-1 Start OP with SIMATIC NET

If you aren't using SIMATIC Net, start the operator panel via the SIMATIC Manager, **MC_EMC** and **EMC_V2**.
Then right-click on **EMC_V2** and select **Start Runtime**.

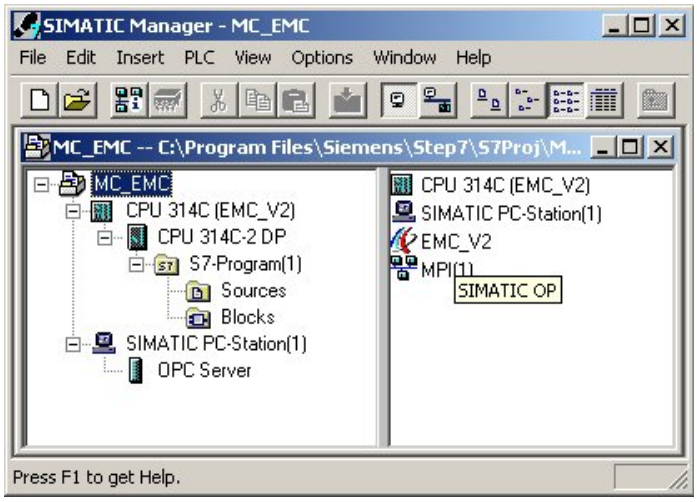


Figure 7-2 Start OP without SIMATIC NET

7.3 Behavior in the Event of an Error in the MICROMASTER

If an error occurs in the MICROMASTER, it is displayed directly on the MICROMASTER and also has to be acknowledged directly there.
The error is not reported to the controller and is not displayed by ProTool/Pro RT either.

If an error occurs in the MICROMASTER, it can be acknowledged as follows:

Option 1: Disconnect the inverter from the mains and then reconnect it again

Option 2: Press the Fn button on the AOP or BOP

Option 3: Via digital input 3 (in the case of standard parameterization)

7.4 Control Functions in Manual Mode

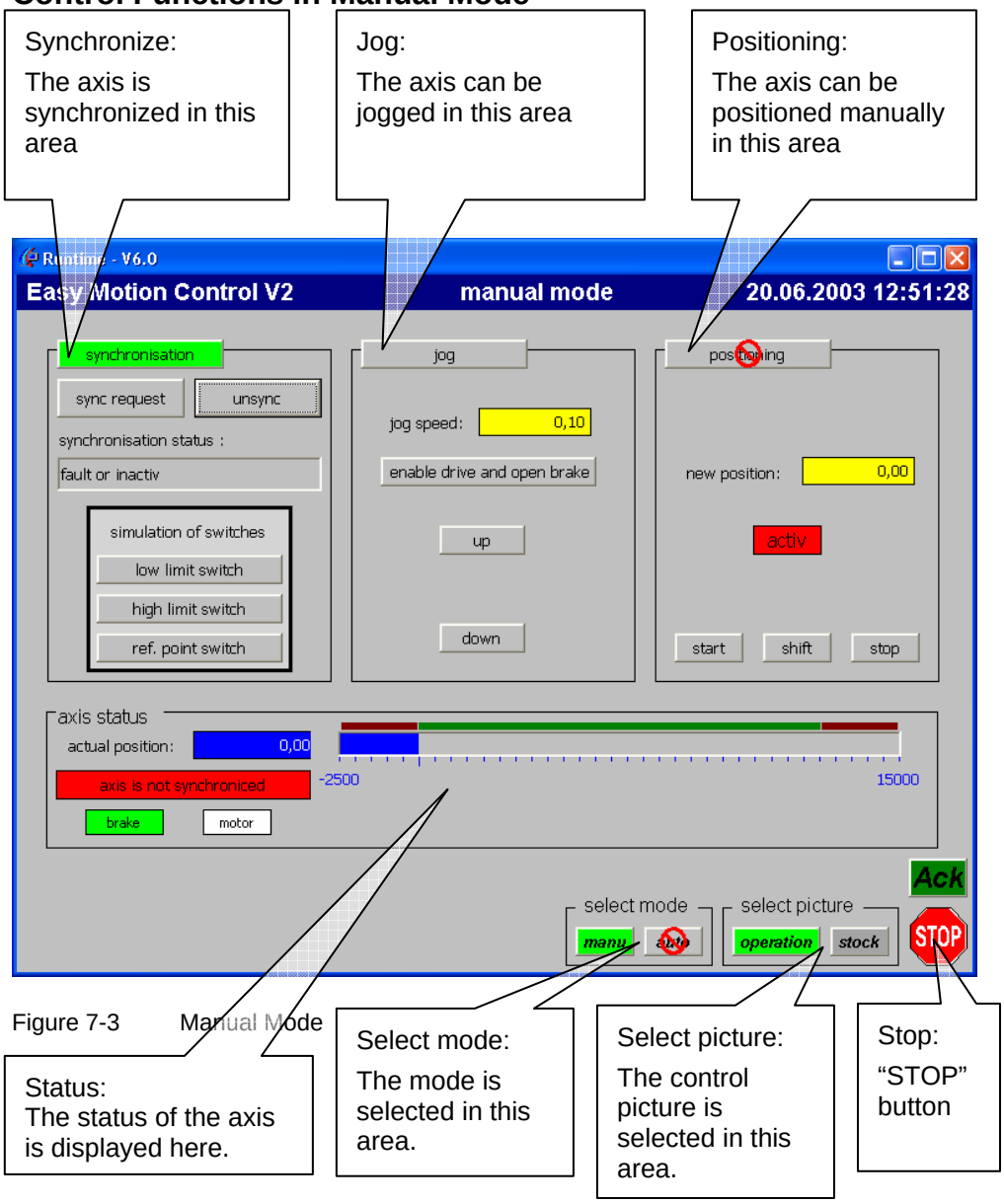


Figure 7-3 Manual Mode

Only the synchronization and jogging functions are available while the axis is not synchronized. Following synchronization, manual positioning or automatic mode may also be used.

The current position of the drive is indicated in mm in axis status. The bar also shows the position in mm.

The function range is active; this is indicated by a green heading. In order to select the active range, click on the heading.

Note

If you click on the **STOP** button in the bottom right-hand corner, the motor is switched off immediately and the brake is activated. The **STOP** button starts flashing. You can cancel the stop status in manual mode by clicking on the **Ack** button.

7.4.1 Jog the Axis

The axis can only be jogged in manual mode.

Table 7-1 Jog

Step	Action
1	Switch to manual mode by clicking on the **manu** button and wait until the drive comes to a stop.
1	Click on **jog** to activate this area. The heading turns green.
2	Enter a jog speed in mm/s, e.g. 50.
3	Enable the drive by clicking on **enable drive and open brake**.
4	Click on the **up** and **down** buttons to jog the axis. Pay attention to the axis status to view the changes.
5	Disable the drive again by clicking on **disable drive and close brake**.

Note

If, after clicking on **enable drive and open brake**, the axis immediately moves for approx. 1 sec, then stops, use the Easy Motion Control parameterization software (Axis Error tab) to check whether the **Standstill range exited** error has been activated. If so, you need to perform the wiring test, see Section 6.3.6.1 Wiring Test.

7.4.2 Search for Reference / Synchronization

The axis can only be synchronized in manual mode.

Table 7-2 Synchronization

Step	Action
1	Switch to manual mode by clicking on the **manu** button and wait until the drive comes to a stop.
2	Click on **synchronization** to activate this area. The heading turns green.
3	Click on **Sync request** to start the synchronization process. The drive runs in a negative direction until it reaches the lower limit switch.
4	If you haven't connected a real limit switch, click on **low limit switch** to simulate this.
5	The drive stops and then automatically starts turning in a positive direction until the reference point switch responds.
6	If you haven't connected a real limit switch, click on **ref. point switch** to simulate this.
7	The drive is now synchronized, and all the operating modes are available.
8	If the reference point is not found and the upper limit switch has been activated, synchronization automatically starts again from the start. Click on the **unsync** button to delete the synchronization process.

Note

If, after clicking on **sync request**, the axis immediately moves for approx. 1 sec, then stops, use the Easy Motion Control parameterization software (Axis Error tab) to check whether the **Following distance exceeded** error has been activated. If so, you need to perform the wiring test, see Section 6.3.6.1 Wiring Test.

7.4.3 Positioning

The axis can only be positioned manually in manual mode and if the axis is synchronized.

Table 7-3 Positioning

Step	Action
1	Switch to manual mode by clicking on the **manu** button and wait until the drive comes to a stop.
2	Click on **positioning** to activate this area. The heading turns green. The axis has to be synchronized for this.
3	Enter the position job in mm in the **new position** field.
4	Click on **start** to activate the position job.
5	While one active positioning job is running, another can be entered. To do this, enter another position and click on **start** again. This job is activated automatically as soon as the active job has been completed. A job that is waiting to be activated can be overwritten by entering a new job.
5	While one active positioning job is running, another can be entered to replace the current job. To do this, enter another position and click on **shift**. The active job is cancelled instantly, and the new job is activated.
6	Click on **stop** to interrupt the active job. The drive comes to a standstill.

7.5 Control Functions in Automatic Mode

Caution

If the axis isn't at position 0.0, a timer runs for 10 s (idle timer) when you switch to automatic mode. Once this time has elapsed, the axis move automatically to idle position without any further inputs from the user.

Note

Automatic mode can only be selected once the axis has been synchronized.

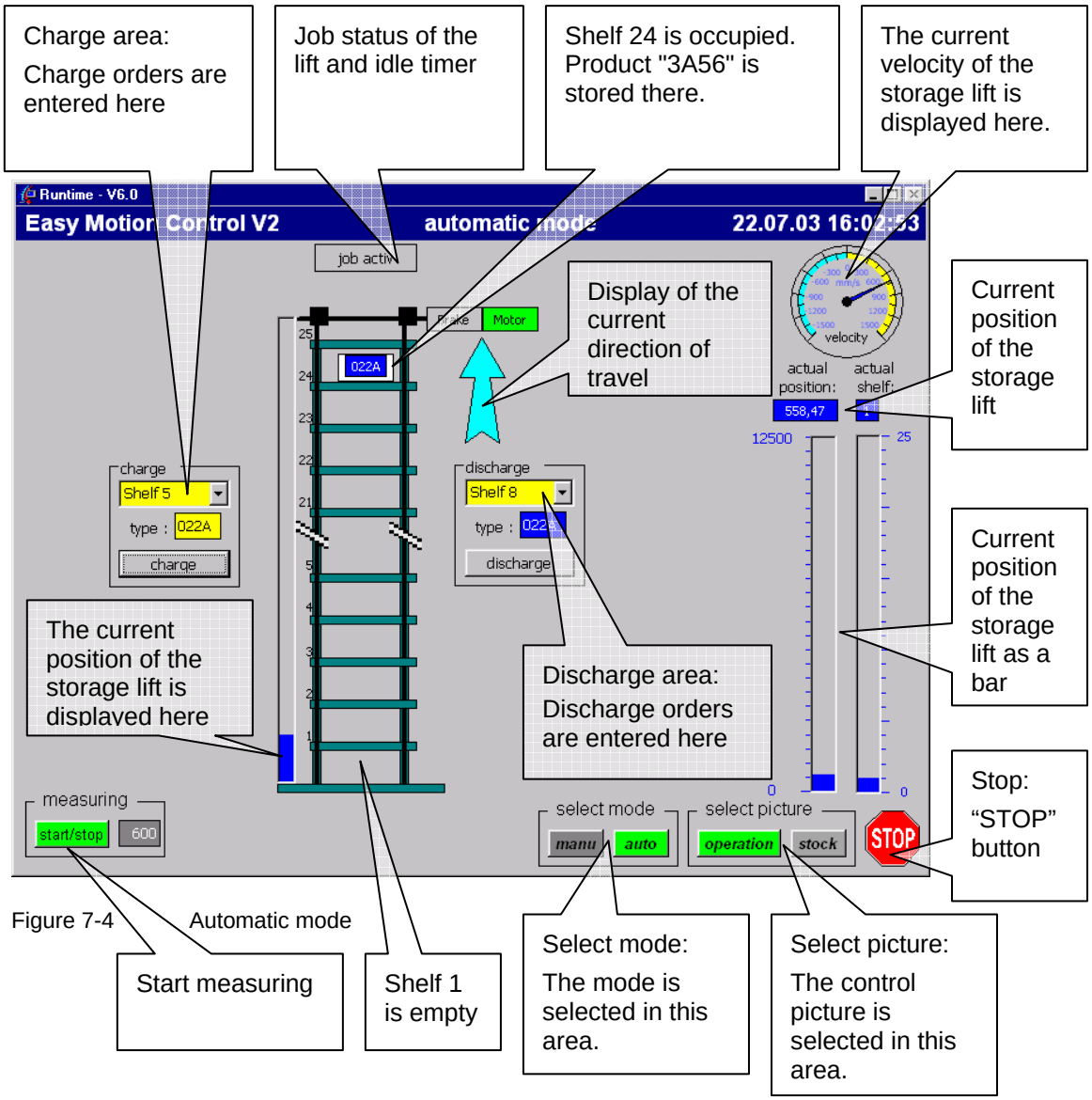


Figure 7-4 Automatic mode

7.5.1 Charge a box

Table 7-4 Charging

Step	Action
1	Click on the **auto** button to select automatic mode.
2	Choose an (empty!) shelf.
3	Indicate the type of product that you are placing in storage (max. four-digit hexadecimal number).
4	Click on **charge**.
5	The following steps are now performed automatically: • If the lift isn't in the basement, it travels there. • A store operator brings over the product, puts it into a box and places the box onto the lift. • The lift travels to the required shelf. • The box is pushed onto the shelf.
6	If there is another job pending, it is now started. If there isn't another job pending, a timer (the idle timer) starts to run. If another job isn't entered before its time has elapsed, the lift moves automatically to the idle position, right in the basement.

Note

An alternative positioning job can be generated while the lift is moving down to its idle position if the operator places an instruction to discharge a box. Otherwise, a new job is not started until the current job has been completed. The only action that can be overridden is the journey to the idle position.

7.5.2 Discharge a Box

Table 7-5 Discharging

Step	Action
1	Click on the **auto** button to select automatic mode.
2	Choose an (occupied!) shelf.
3	Click on **discharge**.
4	The following steps are now performed automatically: • The product type is displayed. • The lift travels to the required shelf. • The box is taken from the shelf. • A store operator takes the box off the lift and takes the product out of the box.

7.5.3 View and Edit Stock

The stock can be viewed any time by clicking on the **stock** button.

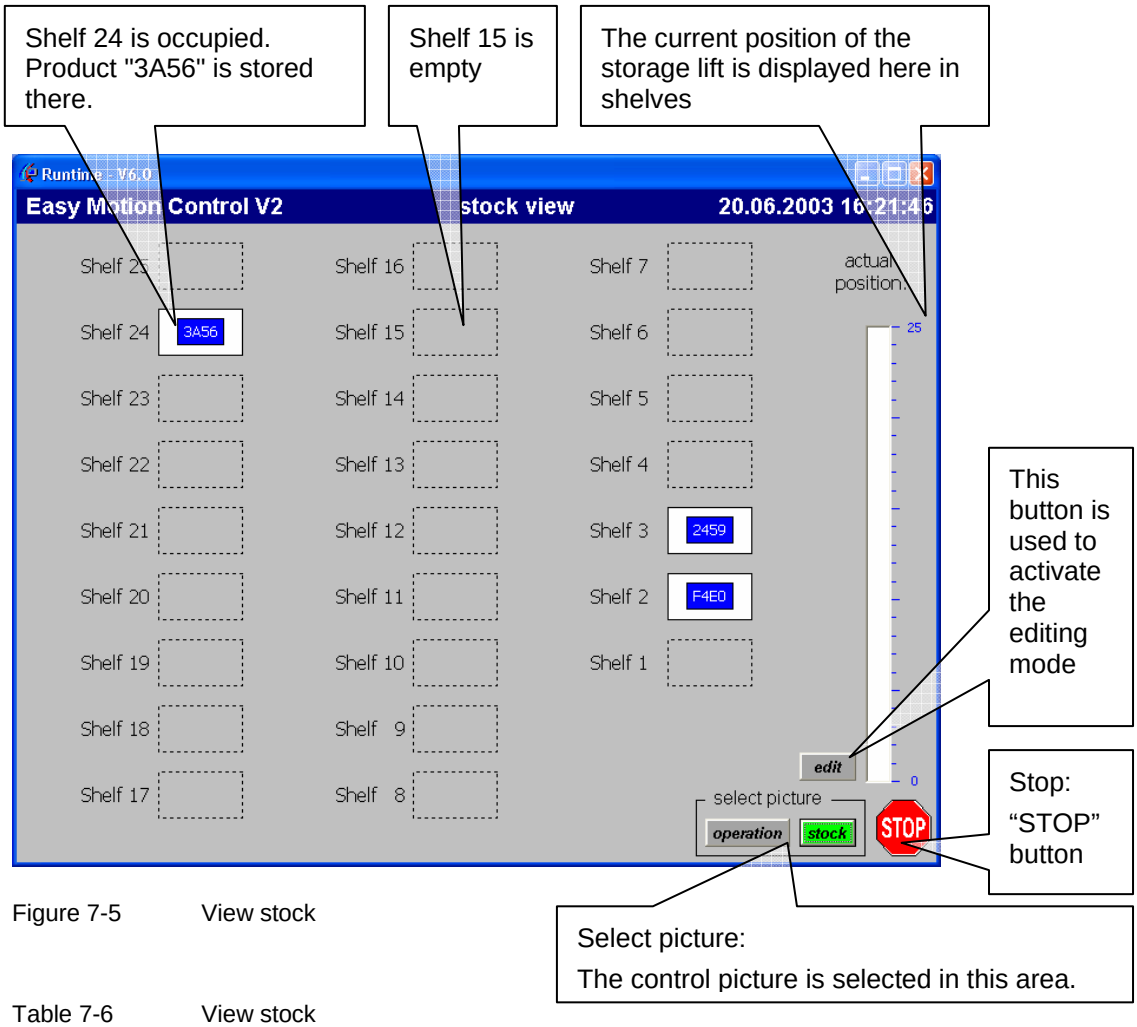


Figure 7-5 View stock

Table 7-6 View stock

Step	Action
1	Click on the **stock** button to select the **view stock** function.
2	Click on the **edit** button to start the editing mode.
3	Click on the **operation** button to switch back to the mode display.

You can edit the stock in editing mode.

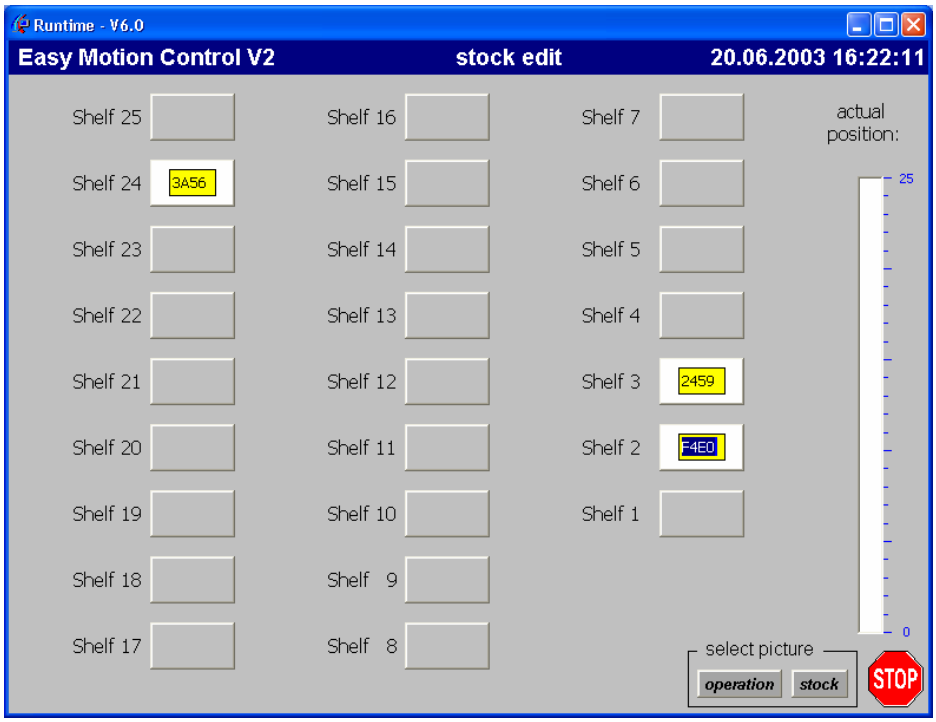


Figure 7-6 Edit stock

In order to fill a shelf, click on the corresponding gray rectangle. A white box now appears. Click on the yellow field to change the type of product being charged.

Click on a white box to empty the corresponding shelf.

7.6 Record and Evaluate the Positioning Task

The actual position and velocity are recorded in two DBs. The recording process is started via the HMI.

Figure 7-7 Start or stop measurement

SIMATIC NET must be installed in order to install into EXCEL. The OPC server contained in SIMATIC NET enables Windows programs to access the data in the controller.

The measuring control field is located in the bottom left of the control masks. Click on **start/stop** to start or stop the measurement. While a measurement is running, the **start/stop** button is highlighted in green. The serial number of the last measurement is shown. Once the maximum possible number of valued have been measured, the measuring process automatically ends.

Every new measurement overwrites the previous measurement. Therefore, data should be imported into an EXCEL worksheet and saved after each measurement sequence.

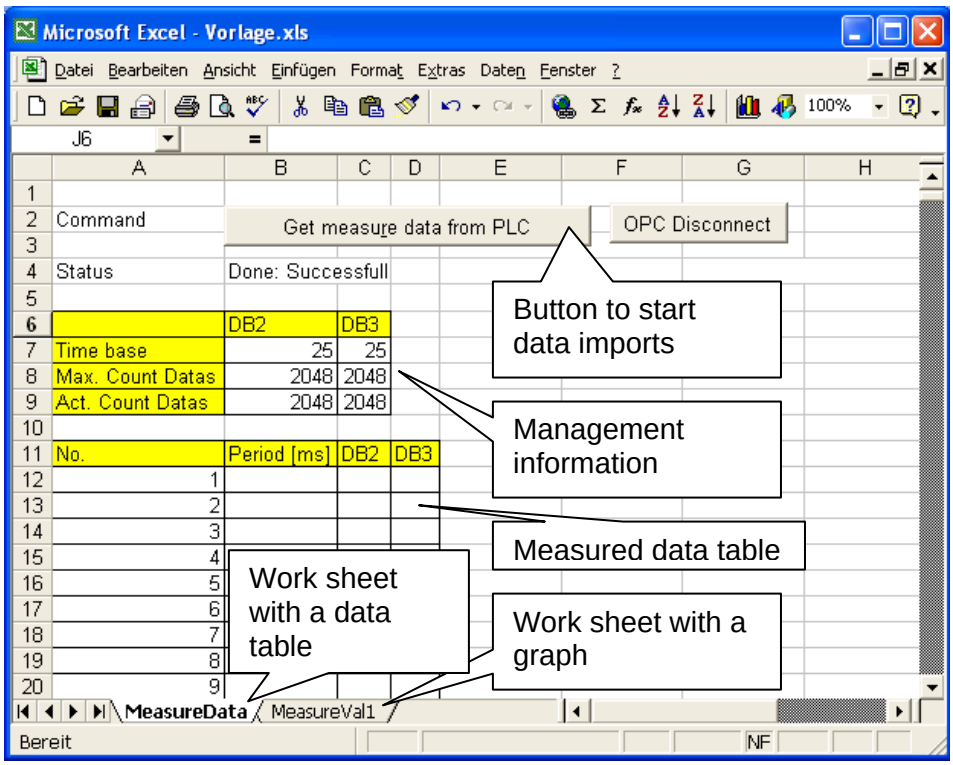


Figure 7-8 Evaluation in EXCEL, worksheet

Table 7-7

Step	Action
1.	Open the **Vorlage.xls** file.
2.	Click on the **Get Measure data from PLC** button.
3.	Once all the data has been imported, **Ready** appears in the status line again.
4.	Switch to sheet 2, **MeasureValue1.** The measured values are displayed graphically on this page.
5.	Save the EXCEL worksheet under a different name to be used later on.

The pattern of the position and velocity of the axis is plotted in the axis. The position is shown on the Y-axis on the right and the velocity is shown on the Y-axis on the left.

The axis has been moved from 2800mm to 5800mm in the following example:

The velocity rose at the predefined acceleration of 450 mm/s^2 and, after 2.2s, reached the maximum value of 1000 mm/s. It travels a short distance at constant velocity before then starting to decelerate at -450 mm/s^2 and, after 2.2 s, the axis reaches the target position.

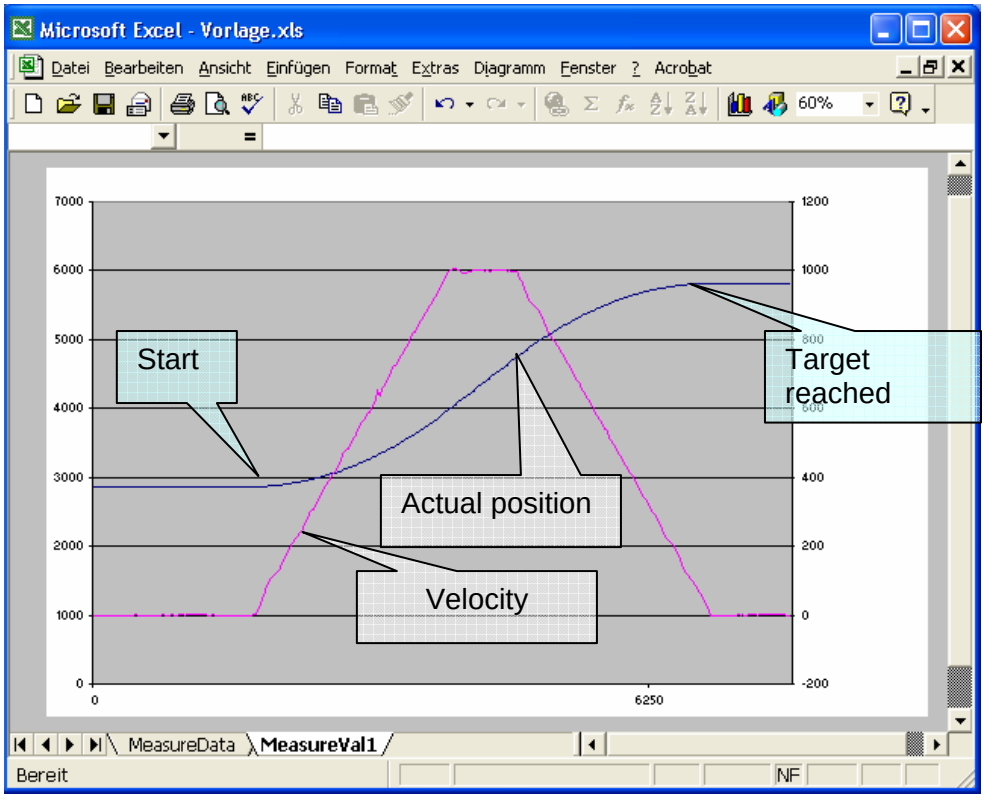


Figure 7-9 Evaluation in EXCEL, graph

7.7 Notes on Troubleshooting

- Encoders and/or motor polarity are incorrectly oriented
  The most frequent cause of errors is when encoders and/or the motor polarity are incorrectly oriented. If this is the case, the closed-loop position controller attempts to compensate for a positive deviation, for example, by making the motor rotate in a negative direction. However, due to the incorrect orientation, the motor rotates in a positive direction. The deviation is increased, and the controller makes the motor rotate more quickly, increasing the deviation yet further. Easy Motion Control ultimately switches the motor off with the message **Following distance exceeded** or **Standstill range exited**. The monitor that responds depends on the operating mode.
  Easy Motion Control possesses parameters which can be used to adapt the direction of rotation of the motor and encoder without the need to rewire. These parameters are defined through the wiring test described in Section Wiring Test.

- MICROMASTER errors
  In the storage lift application, MICROMASTER errors are not evaluated by the controller and are, therefore, only displayed by the MICROMASTER.
  The MICROMASTER's digital output 1, which is factory-set to **Error active**, can be used to report the error status to the controller.
  Acknowledge a MICROMASTER error by pressing the Fn button, acknowledge it via the Starter software, by activating digital input 3 or by disconnecting from the mains and then reconnecting again. If the reason for the error still exists, the error cannot be acknowledged.
- It is best to use the parameterization user interface to evaluate errors that Easy Motion Control has discovered. If the drive is not enabled/switched on, Easy Motion Control treats this as an error.

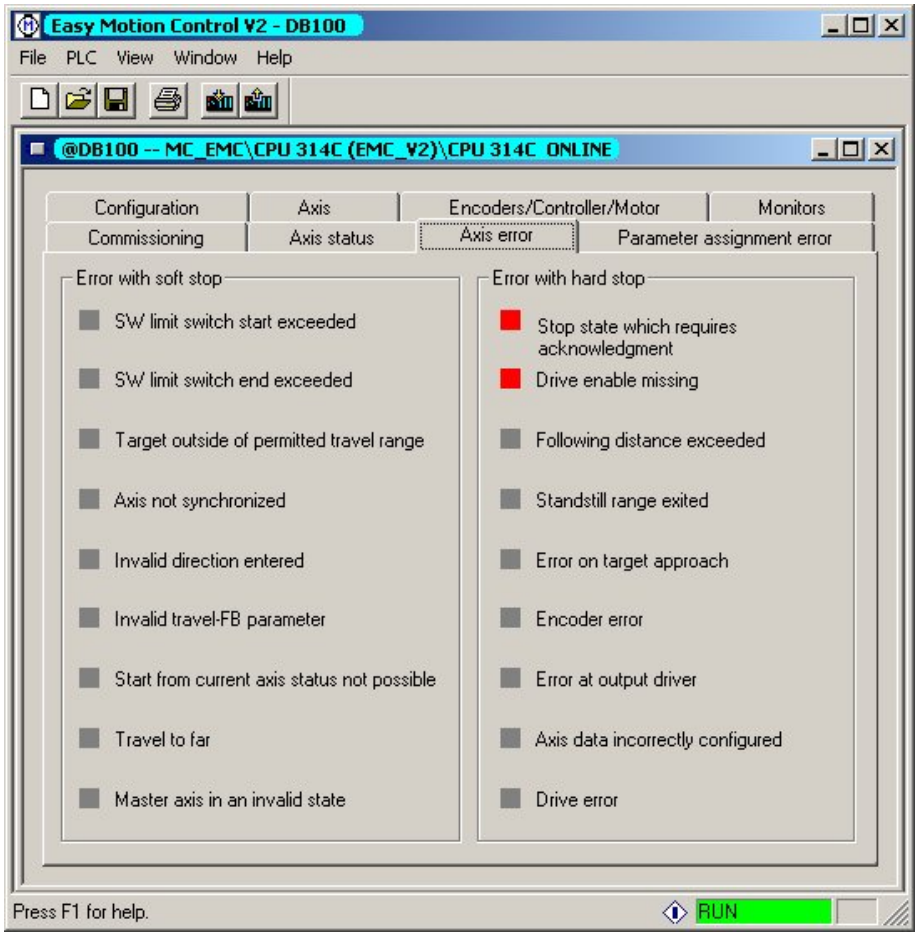


Figure 7-10 Easy Motion Control: Axis error

If the drive is not enabled, Easy Motion Control treats this as an error. Consequently, this error is normal if the motor is not switched on. Once the drive has been enabled, the Easy Motion Control error is always acknowledged by the sequencers in the storage lift application.

Part C : Description of the Program

Overview

Content of Part C

Part C is of interest to anyone wishing to extend/adapt their system on the basis of the existing software.

Objective of Part C:

This part of the documentation is meant to

- explain the details of the code of some core parts of the program to the reader
- supply information on where expansions would make sense.

Requirement

This is not an introduction to the STEP 7 language STL. The reader should have a command of the fundamentals of this language.

It is helpful to read the sections in Parts A1 and A2 before the description of the code.

Topics covered:

8. Description of the Program

In addition to the overview in Section **5.1.5 Application Program**, the code is also annotated in detail in the STL editor. There is no need for any further description.

9. Changes in the STEP7 Program

9.1 Using a PROFIBUS Interface to Activate the MICROMASTER

Using the PROFIBUS instead of analog transmission will prevent parasitic interference to the setpoint and render offset compensation unnecessary.

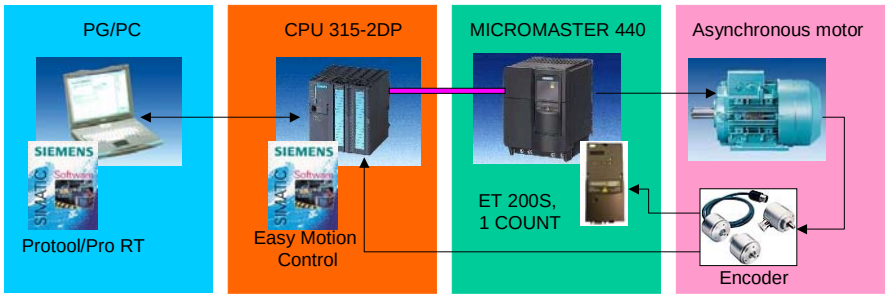


Figure 9-1 Overview

Requirements

- The MICROMASTER is equipped with the PROFIBUS module 6SE6400-PB00-0AA0.

Note

If the DriveES Basic software package is installed, the MICROMASTER may be inserted in the SIMATIC Manager as an object. Parameters can then be assigned to the MICROMASTER with the Starter software and via PROFIBUS. There is then no need to switch the PC/PG interface.

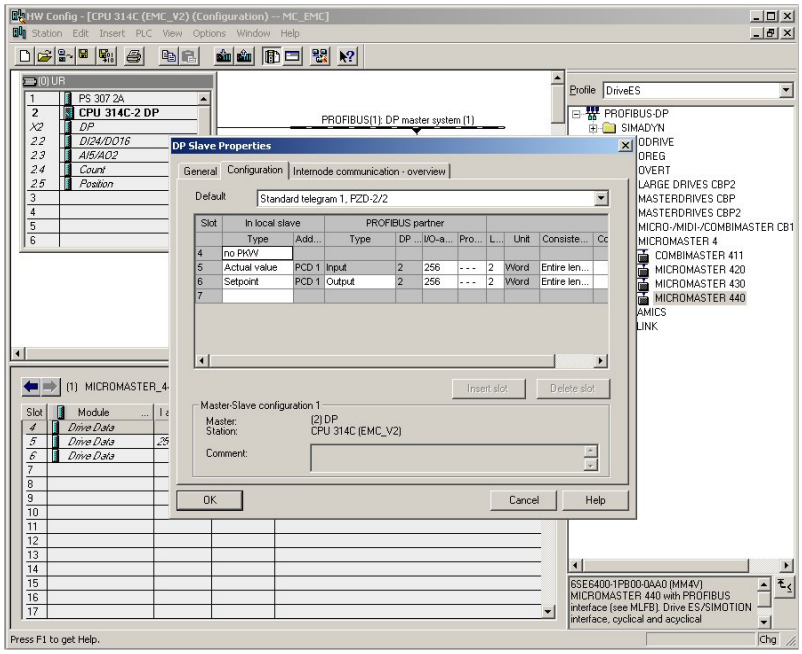
Note

In case you will use the function “motor holding brake” of the MICROMASTER 4, you are desperated to pay attention to Hotfix1 of SIMATIC Easy Motion Control V2.0. The download is available free of charge with the following URL

http://www4.ad.siemens.de/WW/view/en/17661771

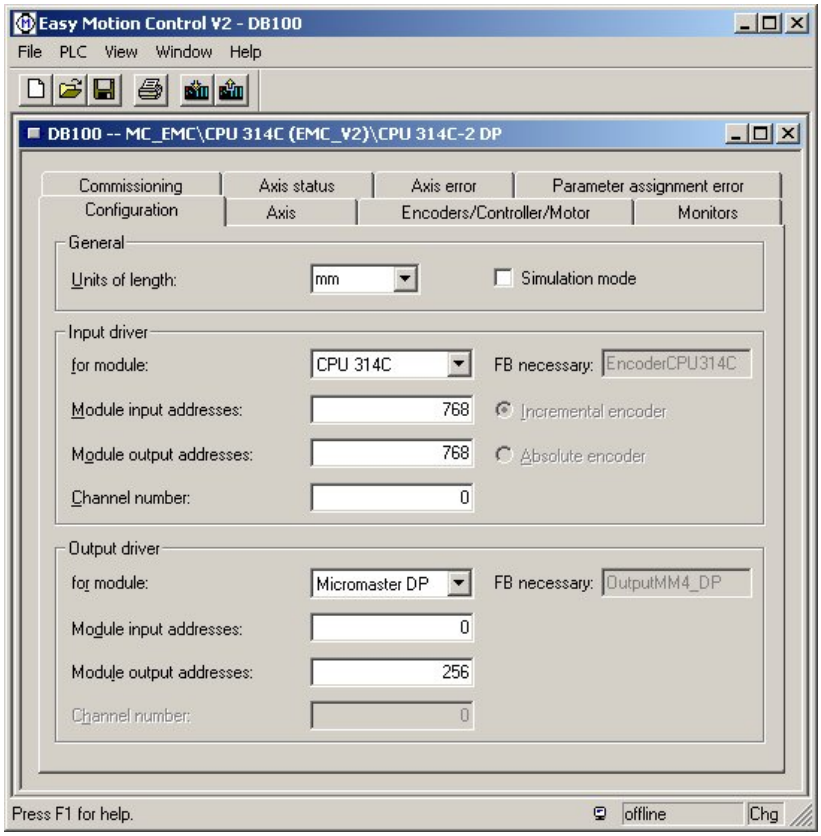
Adapt the STEP 7 blocks

Table 9-1 Adapt the STEP 7 blocks

Step	Action
1	Open HW Config.
2	Create a DP master system.
3	Look for MICROMASTER 4 in the Step7 hardware catalog under: **PROFIBUS DP** > **Additional Field Devices** > **Drives** > **SIMOVERT** > **MICROMASTER 4** or under **PROFIBUS DP** > **SIMOVERT** > **MICROMASTER 4** You may have to install the appropriate GSD file first of all.
4	Place the MICROMASTER 440 on your DP master system and assign it the required DP address. Change to the Configuration tab for the PROFIBUS slave. Insert one of the following modules in slots 0 and 1: **0 PKW, 2 PZD (PPO3)**, **Standard telegram 1, PZD-2/2** or **PPO-Type3, PZD-2/2**. Use the suggested I/O addresses or select addresses of your own.  Figure 9-2 HW Config, MICROMASTER Configuration (with the DriveES Basic package installed)
5	Delete **FB 34 OutputCPU314C** and insert **FB37 OutputMM4_DP** from the **EMC2 Easy Motion Control** library into the CPU block container in place of it.

Step	Action

6	Replace the call for **FB 34 OutputCPU314C** from OB35 with **FB37 OutputMM4_DP**:
	`//  CALL  "OutputCPU314C" , "idb_OutputCPU314C"`
	`//    EnableDrive:=#Drive_enabled`
	`//    OutErr     :=FALSE`
	`//    Axis       :="db_lift_axis".Ax`
	`//    Init       :="db_lift_axis".Ax.Init.I7`
	
	`CALL  "OutputMM4_DP" , DB37`
	`EnableDrive:=#Drive_enabled`
	`OutErr     :=FALSE`
	`Axis       :="db_lift_axis".Ax`
	`Init       :="db_lift_axis".Ax.Init.I7`

7	Save and compile the configuration. Load it into the CPU.
8	Open your axis DB (DB100) using the EMC software.
9	Select **MICROMASTER DP** as the output driver. Copy the I/O addresses to the **starting address of the module output addresses**.  Figure 9-3 Axis DB for MICROMASTER connection via PROFIBUS
10	Save the axis DB and load it into the controller.

Step	Action
11	Assign parameters to the MICROMASTER to suit your requirements. (DP address, operation as a DP slave, ...) on the basis of the MICROMASTER documentation.
12	Ensure that the reference frequency entered in parameter P2000 of the MICROMASTER is identical to the **Reference value for 100% speed** ("DriveInputAt100") in the axis DB for your axis.

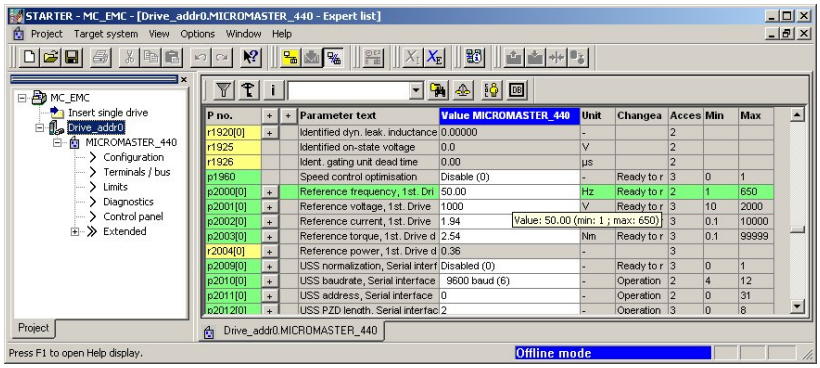


Figure 9-4 P2000 in the Starter, Expert list

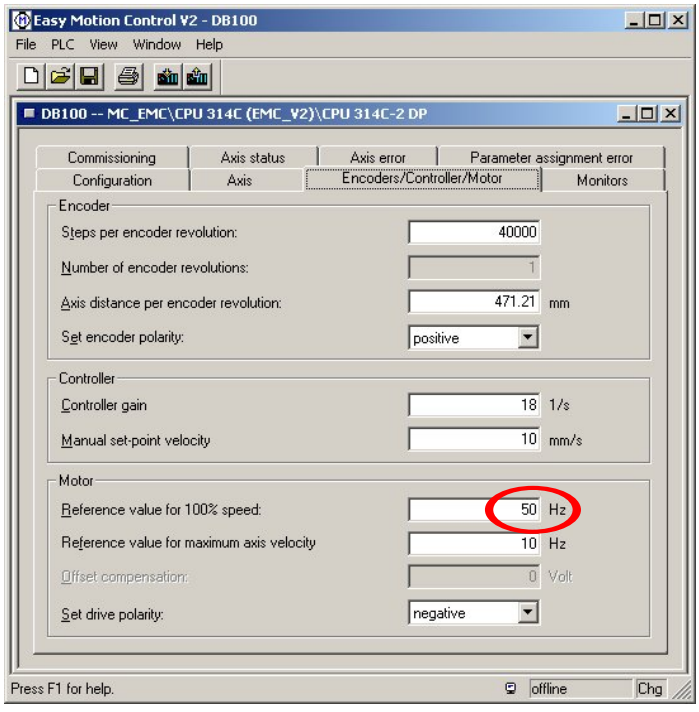


Figure 9-5 Axis DB, Reference value for 100% speed

9.2 Using another OB35 Time Reference

If the OB35 has to be called at intervals other than 25ms for the watchdog interrupt, enter the corresponding value in the HW Config under the Properties for the CPU 314C. Then compile the new configuration and load it into the CPU.

As Easy Motion Control has to know the sample time at which the watchdog interrupt is called, there is a parameter (DBD 0) for this in the axis DB.

In this application the sample time is exported during the first OB35 pass and written to the axis DB. This serves to ensure that this parameter is automatically always correctly set.

The first pass is detected by means of a flag which is set in the start-up OB100. When the OB35 is called, this flag is checked. If it is set, the time value is converted to the requisite format and written to the axis DB. The flag is reset in the end.
The next time the OB35 is called, this part of the program is skipped because the flag is no longer set.

As the measured values are also saved in the OB35, the time reference is also written to the measurement DB. The EXCEL macro, which imports the values with the aid of the OPC server, creates the values from the time axis for this.

This is an adequate way of setting the time interval in the HW Config. The application automatically copies the value to all the requisite places.

Part D: Appendix

Overview

Content of Part D

Part D contains definitions of some basic terminology, as well as literature and Internet links.

Content of Part D:

This part of the documentation is meant to

- show the reader where they can find other information
- provide details on where they can download the relevant freeware.

Topics covered:

10. Appendix

10.1 Important Basic Terminology

Closed-loop speed controller

The closed-loop speed controller is implemented in the inverter. It ensures that the motor keeps turning at the required speed, specified by the automation system.

Closed-loop position controller

The closed-loop actual position controller is generally implemented in the controller. It compares the setpoint position and the actual position and supplies the setpoint speed for the drive. It is also known as the position controller.

Frequency inverter: V/f, VC and SLVC

Different types of control are used in the inverter, depending on requirements:

- V/f control is the simplest form of operation and is the easiest type to commission. The assignment of parameters for the inverter is essentially limited to entering the motor data according to the rating plate.
  As the motor voltage is in proportion to the frequency in the asynchronous machine according to the principles of electrical machines, the output voltage rises in proportion to the output voltage until 400V is reached at 50 Hz, for example.
  The motor consumes as much current as is required for the load. The inverter has no feedback about the actual motor speed.

- In Vector Control (VC) mode the inverter works internally with a multi-dimensional machine model. This enables it to calculate the prevailing conditions in the asynchronous motor, such as the rotor speed or the current magnetization of the motor, from the measurement of the gain and phase angle of the output currents and voltages and from the motor data entered. This enables higher drive dynamics to be achieved. However, the parameterization of the inverter is more detailed, and the closed-loop speed controller has to be optimized.
  This model works with pointers, which are also called vectors, which is why closed-loop speed control with this model is also called Vector Control.

- The term "Sensor Less Vector Control (SLVC)" is used if there is no additional speed encoder and if only the vector model is used for determining the speed. This means that SLVC isn't as precise or dynamic as VC.

Brake, chopper or pulsed resistor

When brakes are applied electrically to the drive, the motor works as a generator and energy flows back into the inverter. This causes an increase in the intermediate circuit voltage in the inverter. As the MICROMASTER is unable to feed the energy back into the supply network, it activates the brake resistor which converts the energy into heat, thereby lowering the intermediate circuit voltage again.

Note

You can find out basic information, about the design of the drive for example, in the **e-Infoshop** on "Simple Positioning".

10.2 Literature

This list is by no means complete and only reflects a selection of the suitable literature available.

Table 10-1 Literature list

	Topic Area	**Link**
	STEP7	
\1\		Automation with STEP7 in STL and SCL Hans Berger Published by Publicis MCD Verlag ISBN 3-89578-113-4

10.3 Internet Links

This list is by no means complete and only reflects a selection of the suitable Internet links available.

Table 10-2 Internet links

	Topic Area	**Link**
\1\	Link to this Application	http://support.automation.siemens.com/WW/view/en/21669390
\2\	Easy Motion Control Manual	www.ad.siemens.de/support Select product support Open the following directories in the tree: • Automation technology • SIMATIC industrial automation systems • SIMATIC industrial software • Software for SIMATIC S7/C7/WinAC • Runtime software • Easy Motion Control Look here under **Manuals / OIs**
\3\	MM440 Operating Manual	www.ad.siemens.de/support Select product support Open the following directories in the tree: • Drive technology • AC inverter • Low-voltage inverter • MICROMASTER 4 • MICROMASTER 440 Look here under **Manuals / OIs**

	Topic Area	Link
\4\	STARTER (stand-alone) for MICROMASTER	www.ad.siemens.de/support Select product support • Drive technology • (Engineering) software • Low-voltage inverter • STARTER commissioning tool Look here under **Downloads**
\5\	DriveMonitor (stand-alone) for MASTERDRIVE	www.ad.siemens.de/support Select product support Open the following directories in the tree: • Drive technology • (Engineering) software • Low-voltage inverter • Drive Monitor (SIMOVIS) commissioning tool Look here under **Downloads**