TM410

Working with Integrated Motion Control

PERFECTION IN AUTOMATION
www.br-automation.com

Prerequisites and requirements

Training modules	TM210 – Working with Automation Studio TM400 – Introduction to Motion Control
Software	Automation Studio 4 Automation Runtime 4.04
Hardware	X20 controller ACOPOS / ACOPOSmulti / ACOPOSinverter / X20 stepper motor module

TABLE OF CONTENTS

1 INTRODUCTION

B&R's motion control solution is **fully integrated in Automation Studio**, ensuring consistency throughout the entire process of development and commissioning.

This training module describes the steps in Automation Studio that are necessary to develop, configure and implement a motion control solution.

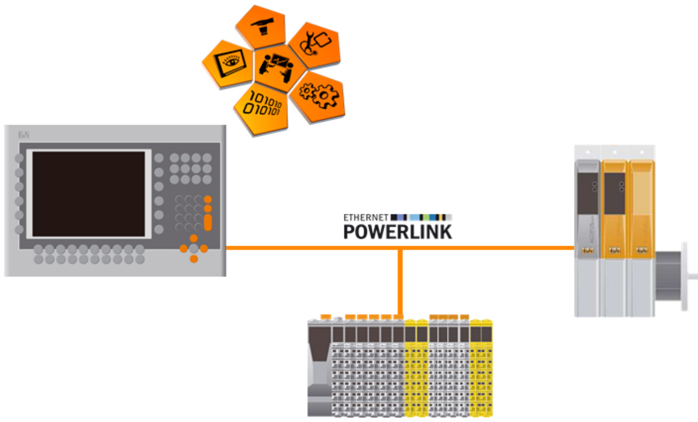


Figure 1: The integrated automation solution from B&R

During this training course, we will be taking a closer look at the role played by the various elements of a motion control solution. The diagnostic tools included in Automation Studio also provide an ideal environment for efficient testing and commissioning. The exercises in this training module are designed to help you become familiar with the fundamentals of working with integrated motion control. The Automation Studio help system will support you as you develop your project by providing detailed descriptions of all necessary parameters.

1.1 Learning objectives

This training module uses selected examples to demonstrate the process of setting up a motion control system in Automation Studio and then commissioning it with NC Test.

- You will learn how to add a B&R drive in the Physical View.
- You will learn about the configuration files used for a motion control system.
- You will learn where to look in the Automation Studio help system to locate the necessary hardware components and their descriptions.
- You will learn how to operate a drive axis in NC Test.
- You will learn how to work in the NC Test window and use NC Trace to collect drive data.
- You will learn about the drive simulation options available in Automation Studio.

1.2 Safety notices and symbols

Unless otherwise specified, the descriptions of symbols and safety notices listed in "TM210 – Working with Automation Studio" apply.

2 DRIVE COMMUNICATION

This section provides a brief description of how controllers and drives communicate and an overview of the various drive parameters involved. It also explains which interface the motion control application and diagnostic tools use to communicate with a drive axis.

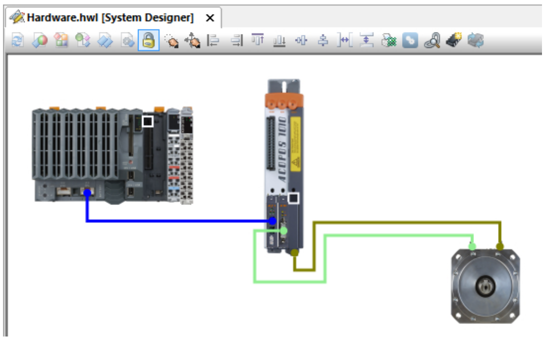


Figure 2: A motion control system as seen in System Designer

The image above shows the components that make up a motion control system. An ACOPOS servo drive is connected via POWERLINK to an X20 controller. The ACOPOS servo drive is connected to the motor using a motor cable and an encoder cable.

The motor cable transfers power to the motor, while the encoder cable is used to provide positioning feedback to the ACOPOS servo drive.

Project development in Automation Studio

Procedure	Description
Project development	The hardware for the motion control system is configured in Automation Studio. This involves, for example, adding an ACOPOS servo drive in the Physical View or System Designer (3.1 "Adding a drive and motor" on page 8).
Configuration	The drive configuration must be adapted to match the actual components being used (4 "Components of the motion control system" on page 25).
Diagnostic and testing tools	NC Test and the System Diagnostics Manager (SDM) provide the user with a variety of diagnostic and commissioning tools. These allow positioning sequences to be started and values to be recorded for analysis (5 "Commissioning and diagnostics" on page 31).
Simulation	Automation Studio allows you to simulate motors and servo drives (6 "Simulation options" on page 54).
Software development	The software for the B&R drive solution is generated using the PLCopen motion control library. More information can be found in the "TM440 – Motion Control: Basic Functions" and "TM441 – Motion Control: Multi-axis Functions" training modules.

Table 1: Overview of project development in Automation Studio

2.1 Drive parameters and drive communication

Drive parameters

Drive parameters can be divided into two general groups. The first group includes all the parameters necessary for the hardware configuration. The second includes the parameters for positioning sequences.

Parameters for the hardware configuration	Parameters for positioning sequences
Motor parameters	Movement parameters
Encoder interfaces	Speed/acceleration values
Supply voltage	Axis limit values

Table 2: Example listing of how drive parameters can be grouped

Drive communication

The drive is operated through the use of parameters, with data exchanged between the controller and the ACOPOS servo drive via POWERLINK. The **NC Manager** serves as the link between the application and the **NC operating system** on the ACOPOS servo drive. The ACOPOS servo drive provides the motor with setpoint values and receives feedback via an encoder interface.

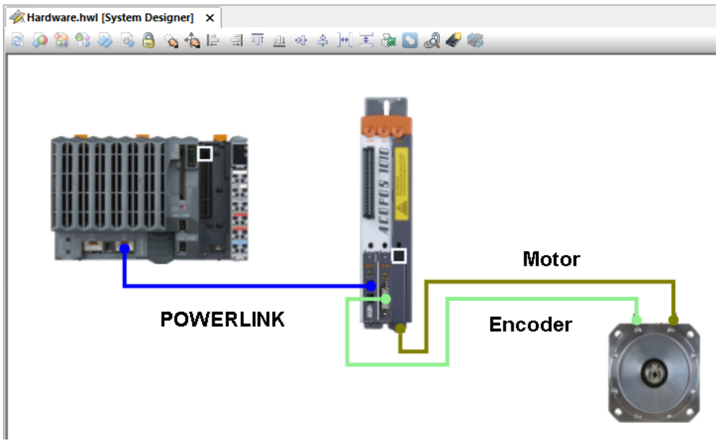


Figure 3: Data exchange between the controller and ACOPOS servo drive via POWERLINK. The connection between the ACOPOS servo drive and the motor is made using a motor cable and an encoder cable.

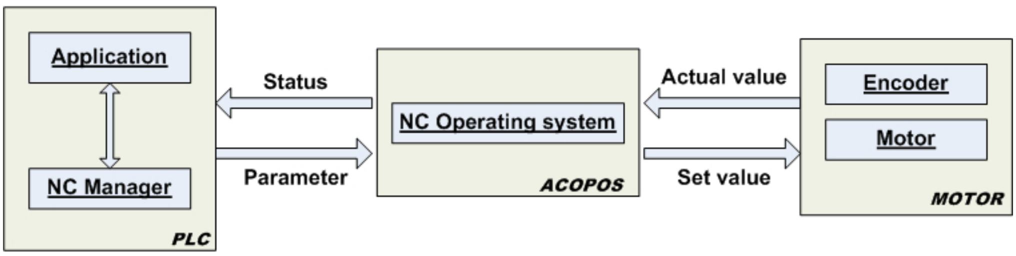


Figure 4: Communication between the drive and the controller. Setpoints are sent to the ACOPOS servo drive, which then returns feedback regarding position.

The data needed for drive communication can be divided into two general categories.

Drive parameters and commands	Status data
Drive parameters are sent by the controller to the drive. Commands are used to communicate positioning information to the drive.	The drive sends status data to the controller containing the status of the drive control loop and positioning sequences.

Table 3: Differentiating between parameters, commands and status data

The NC operating system and the parameters needed to configure the drive are transferred to the drive automatically. This also happens when the connection to a drive is lost or when a drive is replaced. No other steps are required to update the drive.

Easy-to-use PLCopen function blocks in the ACP10_MC library are provided as a programming interface in the motion control application (see "TM440 – Motion Control: Basic Functions").

2.2 Software concept and software interfaces

Regardless of which particular drive components are used, axes are all addressed in the same way in the software. An **NC object** is used to represent each axis – whether it is a servo drive, frequency converter, stepper motor drive or hydraulic drive. An NC object has a standardized data structure that is independent of the type of drive so that the same application can be used for many different axes.

The NC object is used by both the motion control application and the diagnostic tool to address the corresponding axis. NC objects are divided into real and virtual axes for single-axis applications, couplings via electronic gears and cam profiles. The "TM1111 – Integrated motion control" training module describes how to configure and operate path-controlled movements.

Real axis

This NC object is used to operate a real servo drive with a motor and position encoder, for example. The servo drive generates setpoints for the motor, and motor movement is monitored using the position encoder.

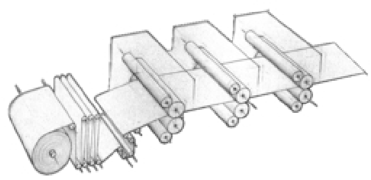


Figure 5: Motor driving a winder drive

Virtual axis

In addition to real axes, ACOPOS also offers the option of operating a virtual drive. This drive works solely as a type of "setpoint generator" (i.e. generates setpoint values for position and speed). It is operated in the same way as a real axis. A real axis can be coupled to the position of a virtual axis.

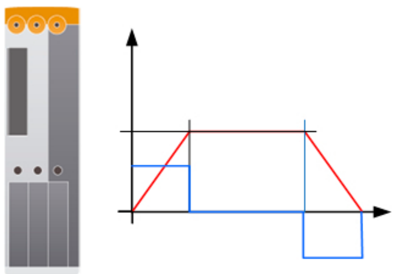


Figure 6: Virtual axis

Motion \ Reference manual \ ACP10 \ NC objects

3 YOUR FIRST PROJECT

This chapter will help you create your project step by step – from adding a drive to Automation Studio to performing the first movements in the NC Test window. At the end, you will complete a training exercise.

The following steps must be carried out:

1. Create an Automation Studio project with an X20 controller.
2. Add a drive on the POWERLINK or X2X interface.
3. Follow the wizard for setting up the drive configuration in Automation Studio.
4. Generate the data for a CompactFlash card.
5. Open the testing environment.
6. Perform the necessary movements.

3.1 Adding a drive and motor

This section will explain how to add an ACOPOS servo drive[1] on a POWERLINK interface. Before you can do this, you will need an Automation Studio project containing an X20 controller. Consult the Automation Studio help system for instructions on how to create a new Automation Studio project. To add an ACOPOS servo drive on a POWERLINK interface, you will need to select it in either the Physical View or in System Designer.

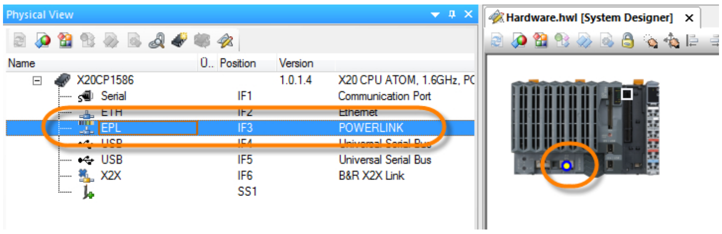


Figure 7: Selecting the POWERLINK interface in the Physical View or System Designer

When the POWERLINK interface is selected, all available devices are preselected in the Hardware Catalog. Set the "Drive" and "ACOPOS" filters to narrow down the selection even further. Any of the listed devices can then be moved to the POWERLINK interface of the X20 CPU using drag-and-drop.

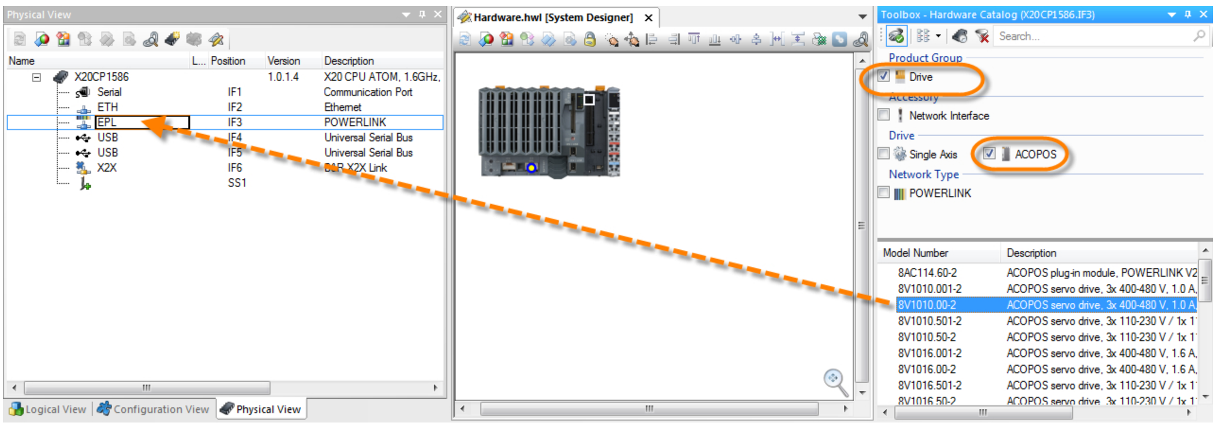


Figure 8: Narrowing down the selection using filters and adding the ACOPOS drive using drag-and-drop

[1] Other devices in the B&R drive family such as ACOPOSmulti, ACOPOSinverter and stepper motor module are also added to the Physical View from the Hardware Catalog.

Automation software \ Getting started

- Creating programs in Automation Studio \ X20 CPU example project
- Creating a motion application in Automation Studio \ Motion application example

Motion \ Project development \ Motion control \ Setting up an axis

The windows in the wizard may vary depending on the drive hardware being used. Individual settings will be explained in the following sections.

- 3.1.1 "ACOPOS configuration wizard" on page 9
- 3.1.2 "ACOPOSmulti configuration wizard" on page 11
- 3.1.3 "Wizard for configuring X20 stepper motor modules" on page 15
- 3.1.4 "ACOPOSinverter configuration wizard" on page 18

3.1.1 ACOPOS configuration wizard

When you add an ACOPOS drive on the POWERLINK interface, a wizard will open to guide you through the process of configuring the new drive. Since the drive is assigned to the POWERLINK interface, the correct interface card has already been added on SS1[2]. The next steps of the wizard help you select the encoder interfaces, motor and basic axis configuration.

Selecting an encoder interface

An EnDat encoder interface has been selected for SS2 of the ACOPOS drive. The SS3 slot should remain empty, so we can move on by selecting "Skip this page".

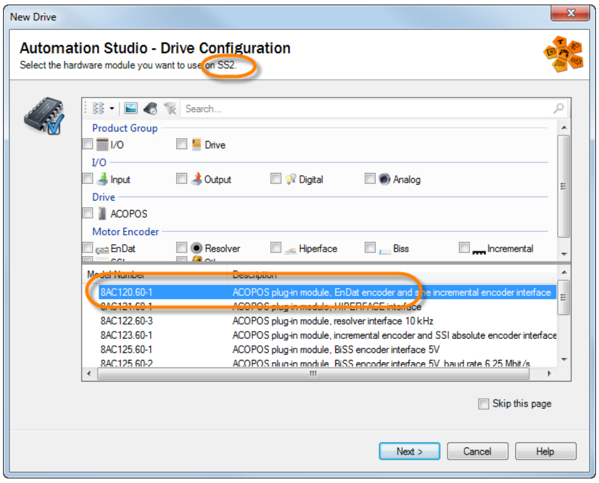


Figure 9: Selecting the encoder interface – EnDat in slot SS2

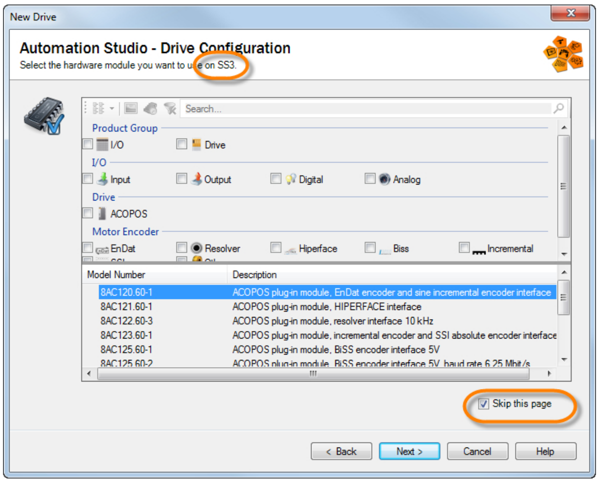


Figure 10: Leaving the SS3 slot empty for an additional plug-in card

2 The abbreviation SS stands for subslot.

Selecting a motor

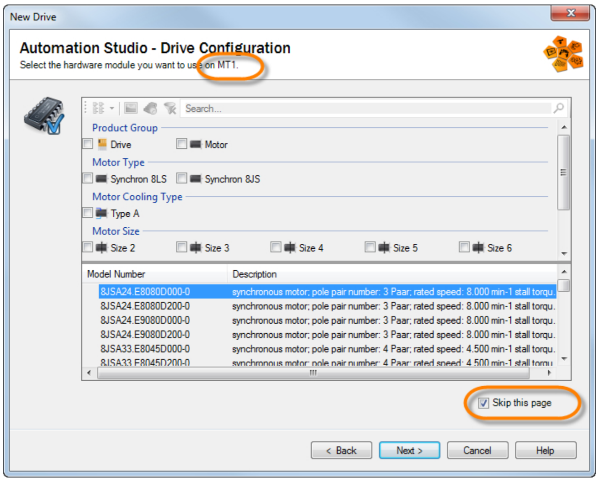


Figure 11: Data for B&R motors is saved to the EnDat encoder interface.

On the previous page, we selected an EnDat interface. If a B&R motor with an EnDat interface is used, then all of its motor data is stored in the encoder. The ACOPOS system reads the motor data from the encoder interface. In this case, the motor selection window can be bypassed by selecting "Skip this page".

Configuring basic axis parameters

The next few pages of the wizard will help you make some basic parameter settings for the drive. First, you can set the language in which axis error messages are displayed. You can then give the axis a reference name. This unique name will be used to address the axis in the motion control application. The "Usage" option is preselected for the PLCopen programming interface to operate the axis later. You can also configure the drive's digital signals.

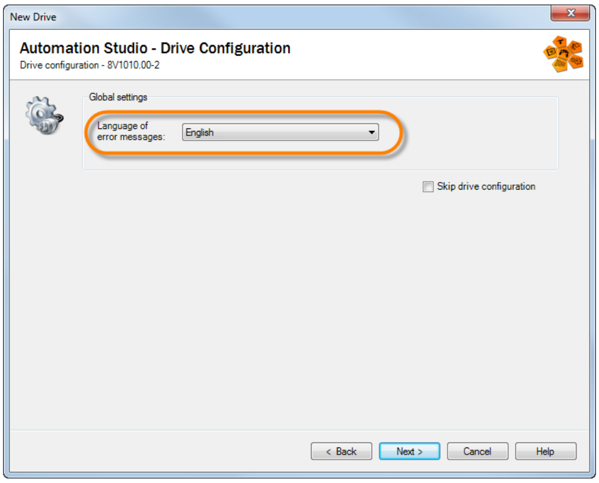


Figure 12: Setting the language for drive error messages

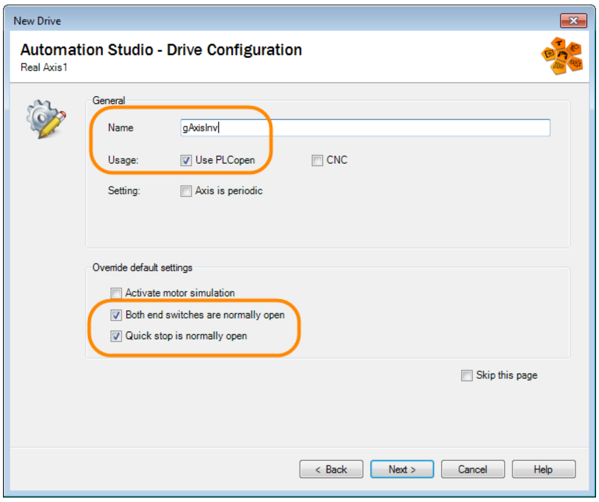


Figure 13: Axis reference name "gAxis01", programming access via PLCopen, configuring the ACOPOS digital input signals

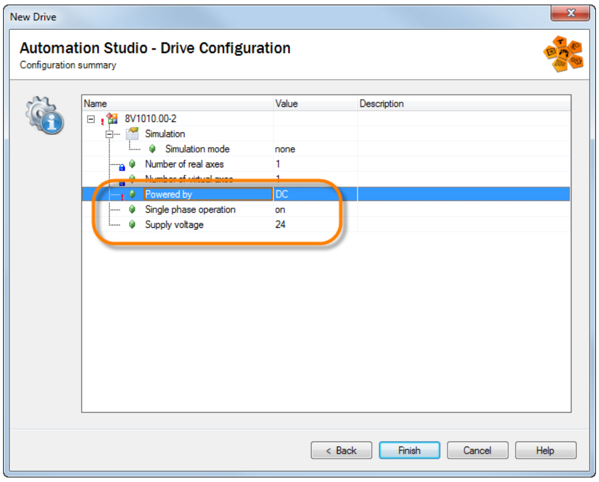


Figure 14: Configuring the supply voltage and phase failure monitoring

The final page of the wizard includes a few settings for the power supply.[3]. In addition to the limit for voltage monitoring, it is also possible to set the mode for phase failure monitoring.
For additional information about ACOPOS servo drive power monitoring and various device types, please refer to the Automation Studio help system.

Motion \ Reference manual \ ACOPOS drive functions \ Power transmission \ Supply \ Monitoring

Hardware \ Motion control \ ACOPOS \ Technical data \

- ACOPOS servo drives \ Indicators
- ACOPOS plug-in modules

3.1.2 ACOPOSmulti configuration wizard

A passive or active power supply module is required to operate ACOPOSmulti inverter modules. Like the inverter modules, these power supply modules are added to the POWERLINK network. The wizards for configuring power supply modules[4] and inverter modules are very similar. Depending ont he ACOPOSmulti module being used, some configuration options do not exist.

Hardware \ Motion control \ ACOPOSmulti \ General information

Hardware \ Motion control \ ACOPOSmulti \ Technical data

- Configuration of an ACOPOSmulti drive system
- 8B0P passive power supply modules
- 8BVP power supply modules
- 8BVI power inverter modules

[3] For testing purposes, the ACOPOS system can also be supplied with 24 V via the DC bus.
[4] Active power supply modules come equipped with slots for plug-in cards. This configuration option doesn't exist for passive power supply modules.

When you add an ACOPOSmulti module[5] on the POWERLINK interface, a wizard will open to guide you through the process of configuring the new drive. The next steps of the wizard help you select the encoder interfaces[6], the motor[7] and basic axis configuration.

Selecting an encoder interface

An EnDat encoder interface has been selected for SS1[8] of the ACOPOS drive. The SS2 slot on the ACOPOSmulti module should remain empty, so we can move on by selecting "Skip this page".

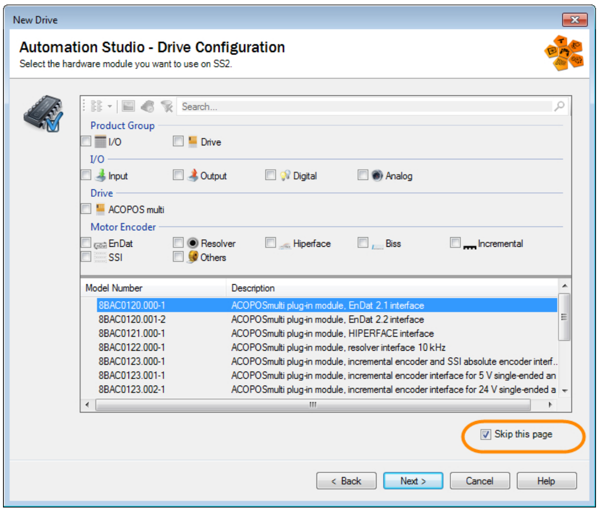


Figure 15: Selecting the encoder interface – EnDat in slot SS1

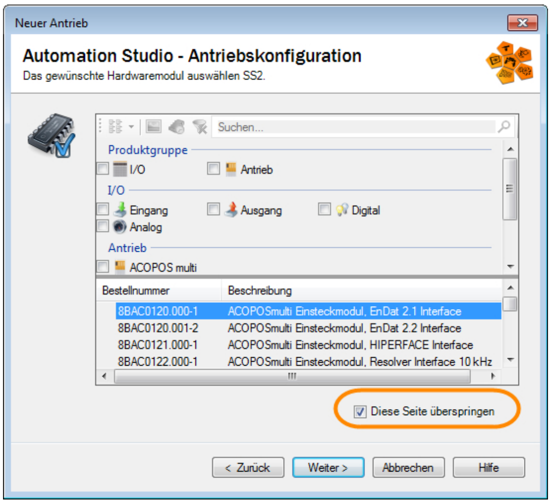


Figure 16: Leaving the SS2 slot empty for an additional plug-in card

Hardware \ Motion control \ ACOPOSmulti \ Technical data \ 8BAC plug-in module

5 Power supply modules and inverter modules are referred to generally as ACOPOSmulti modules in this documentation.
6 This option doesn't exist for passive power supply modules.
7 This option doesn't exist for any power supply modules.
8 The acronym SS stands for subslot.

Power supply and error texts for power supply modules

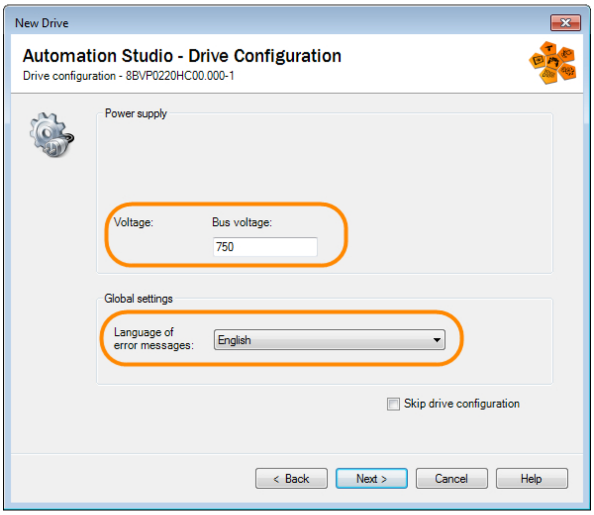


Figure 17: Selecting the bus voltage and language for error texts

When configuring ACOPOSmulti power supply modules, it is necessary to configure the bus voltage for the ACOPOSmulti rack. When using passive power supply modules, the bus voltage does not need to be configured.
It is also necessary to configure the language to be used for error texts.
For power supply modules, the motor configuration and specific settings for the inverter module are not necessary.
The next step in this case is to configure the axis reference and digital input signals.

Hardware \ Motion control \ ACOPOSmulti \ Technical data \ 8BVP power supply modules

Selecting a motor

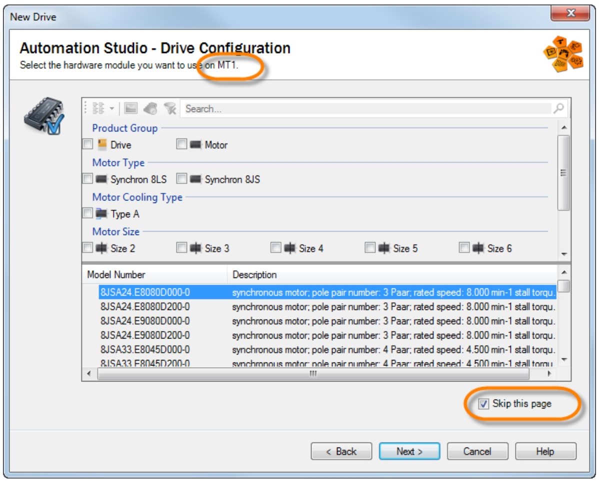


Figure 18: Data for B&R motors is saved to the EnDat encoder interface.

On the previous pages, we selected an EnDat interface. If a B&R motor with an EnDat interface is used, then all of its motor data is stored in the encoder. The ACOPOSmulti system reads the motor data via the encoder interface. In this case, the motor selection window can be bypassed by selecting "Skip this page".

Power supply and error texts for power supply modules

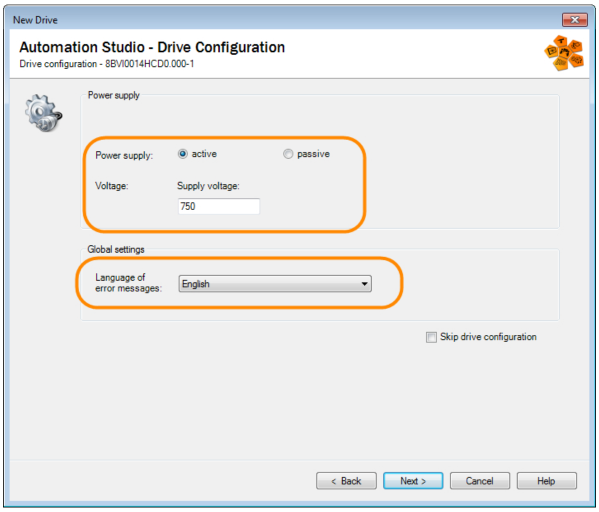


Figure 19: Configuring the supply voltage and language for error texts

The supply voltage of the inverter modules is configured here. The ACOPOSmulti system monitors the specified voltage independently.
It is also necessary to configure the language to be used for error texts.
For additional information about ACOPOSmulti inverter modules and various device types, please refer to the Automation Studio help system.

Motion \ Reference manual \ ACOPOS drive functions \ Power transmission \ Supply \ Monitoring

Axis reference and digital input signals

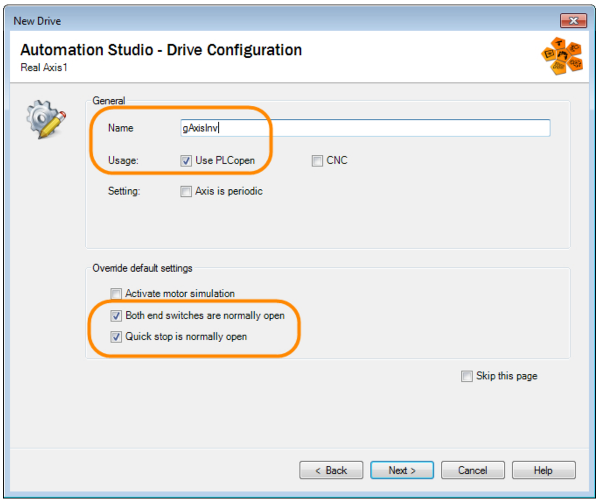


Figure 20: Axis reference name "gAxis01", programming access via PLCopen, configuring the ACOPOSmulti digital input signals

This is where the name of the axis reference is defined. This unique name is used to access the ACOPOSmulti module in the drive application. The "Usage" option is preselected for the PLCopen programming interface to operate the axis later.
It is also possible to configure the digital input signals on the ACOPOSmulti module.

Motion \ Reference manual \ ACOPOS drive functions \ Digital inputs and I/O modules

Completing the Drive Configuration wizard

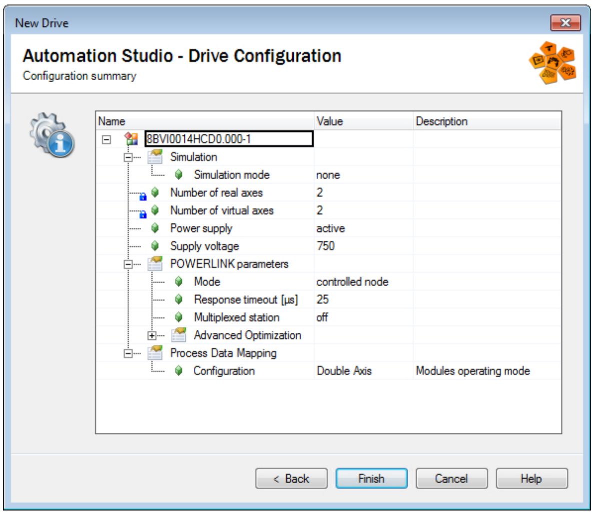


Figure 21: Summary of configuration settings, POWERLINK settings

The closing configuration overview shows the type of power supply as well as the monitored voltage. It is also possible to make device-specific settings for the ACOPOSmulti module on the POWERLINK network in this window. For example, it is possible to have the node number on the POWERLINK network assigned automatically.

> ? Motion \ Reference manual \ ACOPOS drive functions \ Network, position coupling and axis cross-link \ DNA (dynamic node allocation)

3.1.3 Wizard for configuring X20 stepper motor modules

When you add an X20 stepper motor module on the X2X interface, a wizard will open to guide you through the process of configuring the new drive.

Selecting the function model and language for error texts

In the standard function model, the X20 stepper motor module is operated like all other B&R drives. In other function models, the X20 stepper motor module is operated solely using the I/O mapping. The next dialog box allows you to select the language for the drive's error texts.

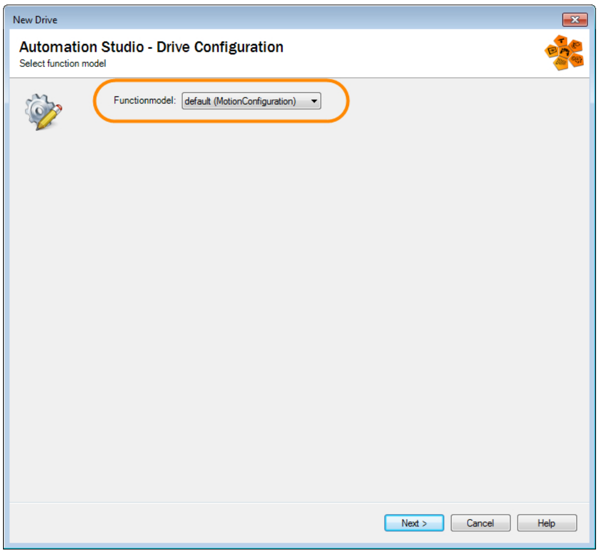


Figure 22: Selecting the function model

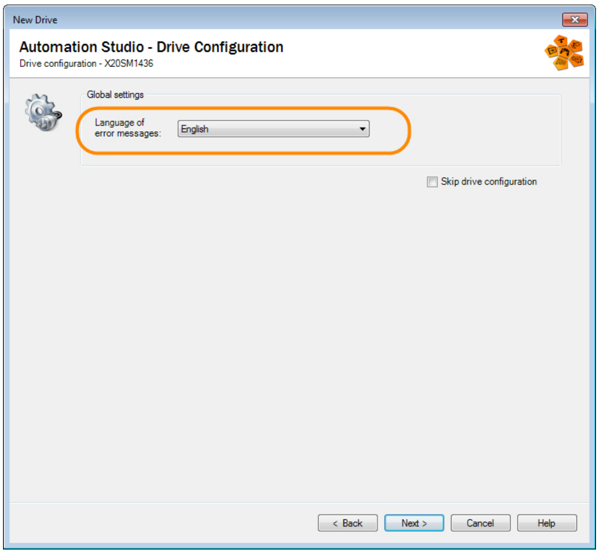


Figure 23: Selecting the language for error texts

Axis reference and digital inputs

A global axis reference must be created to operate the stepper motor axis. This name is used to access the axis in the application software. The levels for E-stop and limit switches will be configured later on. If the logic of the inputs is incorrect or the inputs haven't been wired, then the axis will be prevented from moving.

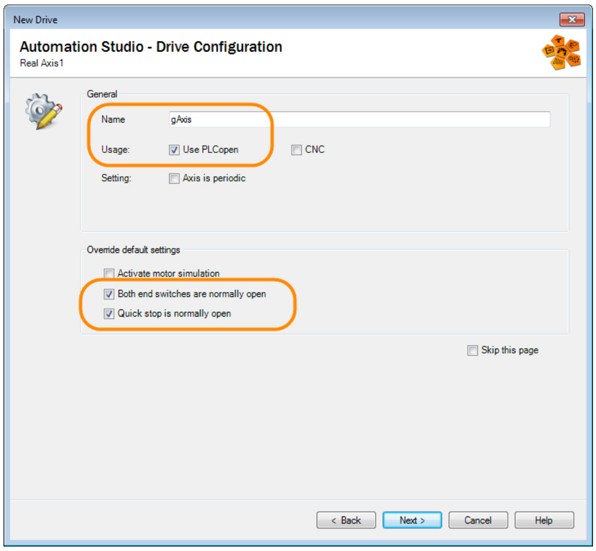


Figure 24: Configuring the axis reference and digital inputs

Configuring basic drive parameters

1. Configuring the current:
   The current must be configured correctly[9] in order to control a stepper motor. The holding current, nominal current and maximum current are configured here. The configuration takes place using the percentage of the current that is being provided by the X20 stepper motor module. A higher holding current causes the stepper motor to heat up at standstill. If the nominal current is not configured high enough, then it may not be possible to reach the configured acceleration or speed setpoint during positioning.
2. Position feedback:

[9] The operating point of the motor should be selected so that the current is less than or equal to the nominal current. An incorrect configuration may cause the motor to become overloaded. This setting will vary depending on the application at hand.

A stepper motor can be positioned either with or without encoder feedback. In the configuration example shown here, an external position encoder is not being used. In this case, the value for the "Position sync" parameter must be set to "Stepper Counter 01". The actual position of the motor is then calculated internally.

3 Units within a motor rotation:
Stepper motors are positioned in microsteps. Each full step has a resolution of 256 microstep. If a motor rotation takes 200 full steps, then this will correspond to 51,200 microsteps. This value must be configured for the "Encoder increments per revolution" parameter if an external position encoder is not being used. If an external encoder is being used, then the number of increments returned by the encoder per motor revolution must be entered here.
The system will automatically convert the units to correspond to the load. In the example given, 1000 units corresponds to 51,200 microsteps, which is the same as one motor revolution.

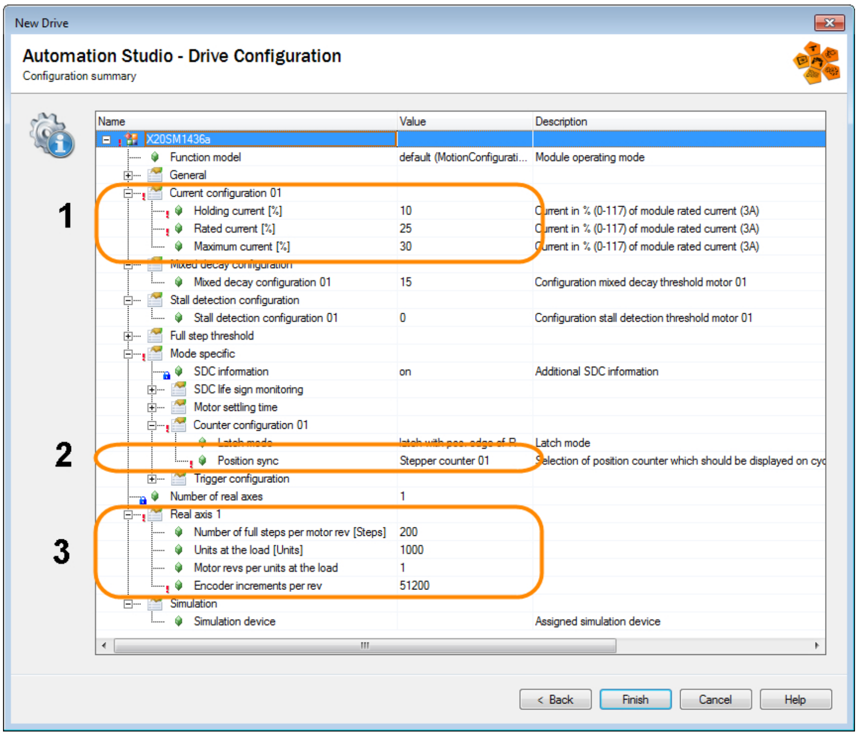


Figure 25: Configuring currents, encoder feedback and units

The parameters that were configured in the wizard can then be called up and edited from the X20 stepper motor module's shortcut menu in the Physical View.

A detailed description of all parameters is available in the "Standard function model" section of the X20 stepper motor module data sheet.

In addition to the configuration objects (4 "Components of the motion control system" on page 25), a global variable declaration, the I/O mapping for the X20 stepper motor module and the "ncsdcctrl" program have been generated. Additional information can be found in the ACP10SDC documentation available in the Automation Studio help system.

Hardware \ X20 system \ X20 modules \ Motor modules \ X20SM1436

- Technical data
- Register of the Standard and Standard+SDC function models

Hardware \ Motion control \ Stepper motors

Motion / Reference manual / ACP10 / SDC overview / System overview

3.1.4 ACOPOSinverter configuration wizard

When you add an ACOPOSinverter module on the POWERLINK interface, a wizard will open to guide you through the process of configuring the new drive.

Selecting the function model and language for error texts

The first two windows of the Drive Configuration wizard are used to configure the function model for the ACOPOSinverter module and the language for error texts. Go ahead and select the "Standard" function model.

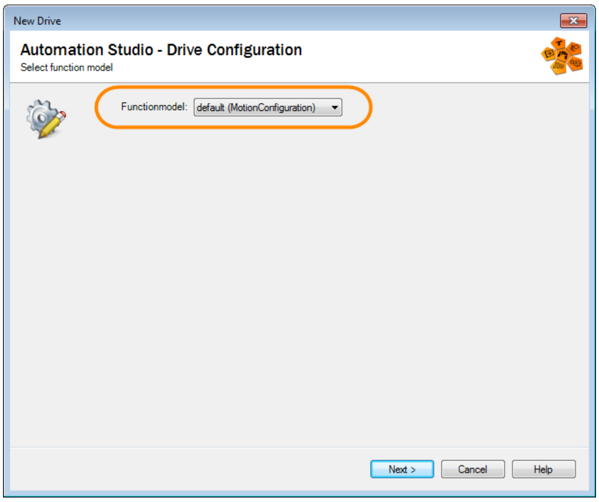


Figure 26: Selecting the "Standard" function model

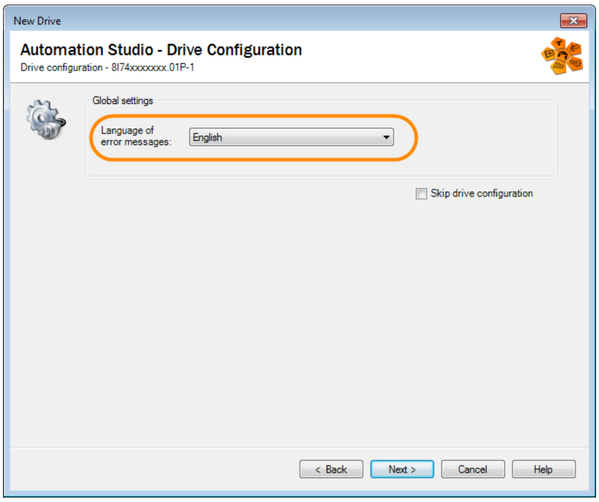


Figure 27: Selecting the language for error texts

Axis reference and digital inputs

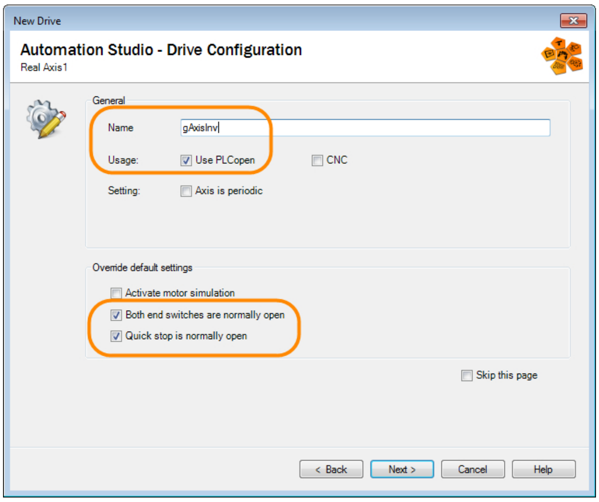


Figure 28: Configuring the axis reference and digital input signals

A global axis reference must be created to operate the axis. This reference is used to access the axis in the application software. The levels for E-stop and limit switch signals will be configured later on. If the logic of the inputs is incorrect or the inputs haven't been wired, then it will not be possible to move the axis.

Configuring basic axis parameters

The next page of the wizard will help you make some basic parameter settings for the drive and motor configuration. This includes entering the data from the motor type plate in the drive configuration. Some parameters are specified on the type plate as a delta (Δ) or star connection. In the following example, the motor is connected to the ACOPOSinverter module using a delta connection (Δ). The following table lists the names and data of a motor type plate as well as the corresponding parameters in the drive configuration.

Name Nameplate	Volts Δ	Volt Y	Hz	rpm	kW	Cos φ	A Δ / Y
Value Nameplate	230	400	50	1310	0.18	0.78	1.03 / 0.59
Parameter ACOPOSinverter	UNS	-	FRS	NSP	NPR	COS	NCR / -

Table 4: Names and data from the motor type plate, assigning in the drive configuration

The image shows the values from motor type plate being used in the drive configuration.

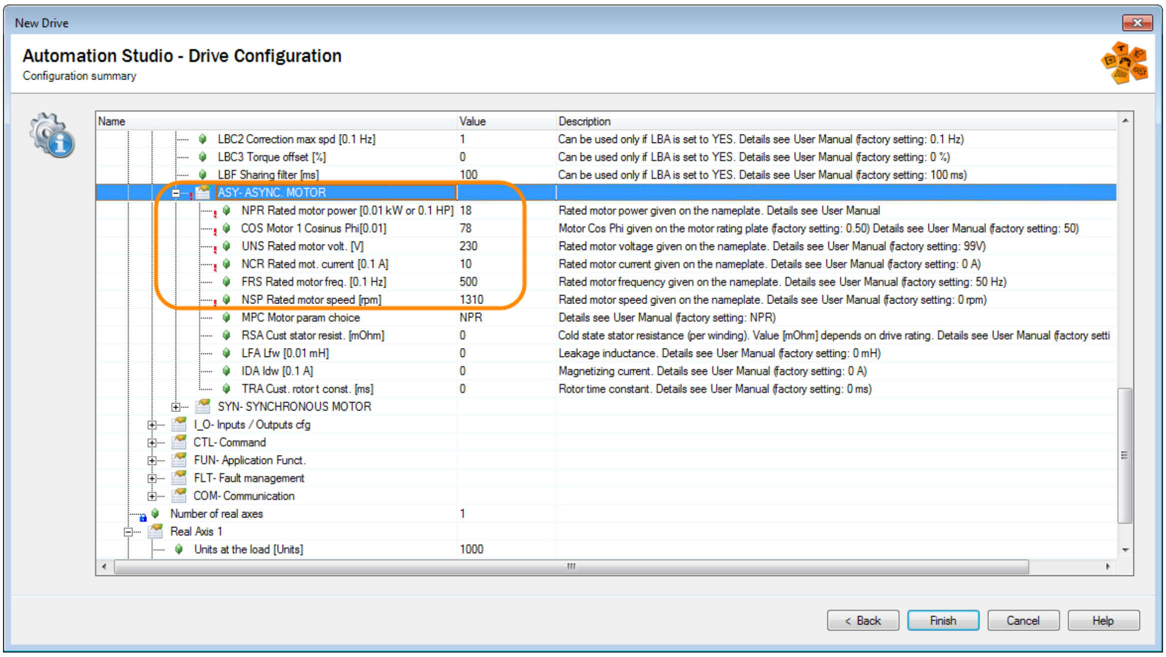


Figure 29: Entering the motor data in the wizard for the drive configuration

After the wizard is completed, it is possible to open and edit the ACOPOSinverter module configuration from the module's shortcut menu in the Physical View. This configuration data is transferred automatically each time the controller is booted or the ACOPOSinverter module is detected on the POWERLINK network. Any existing data will be overwritten. A description is available for all parameters in the respective user's manual. The POWERLINK node number is configured using the integrated operator terminal on the ACOPOSinverter. Drive information is available at runtime via the I/O mapping of the ACOPOSinverter module.

Hardware \ Motion control \ ACOPOSinverter P74 \ Installation guide \ Installation \ POWERLINK interface (8I0IF108.400-1) \

- LED status indicators
- POWERLINK station number

"Hardware \ Motion control \ ACOPOSinverter P74 \ Programming guide \ Programming \ Configuration mode (ConF) \ Menu \

- Factory settings (FCS-)
- All parameters \ Drive data (drC-) \ Asynchronous motor parameters (ASY-)

Hardware \ Motion control \ ACOPOSinverter P74 \

- Programming guide \ Overview \ Operation with SDC
- Register description

3.2 Setting the node number and generating CompactFlash data

The drive can now be seen in the Physical View and System Designer. It is possible to add or replace encoder interfaces by selecting the slots[10] on the drive.

When you added the drive in the Physical View, Automation Studio automatically created all the necessary configuration objects in the Logical View. A description of these objects can be found in a later section (4 "Components of the motion control system" on page 25).

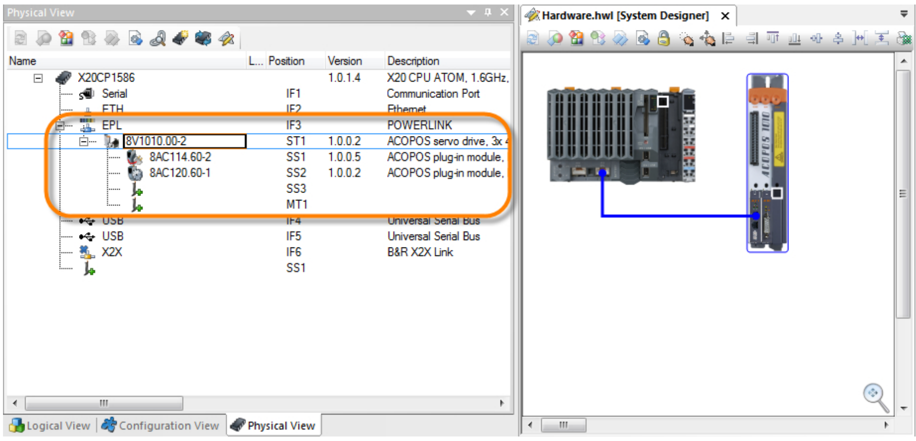


Figure 30: ACOPOS drive now added to the Physical View and System Designer

Setting the POWERLINK node number

Automation Studio assigns POWERLINK station node numbers sequentially. The node number set in Automation Studio must match the one on the communication module of the drive. The node number can be changed in the shortcut menu of the drive in the Physical View or System Designer.

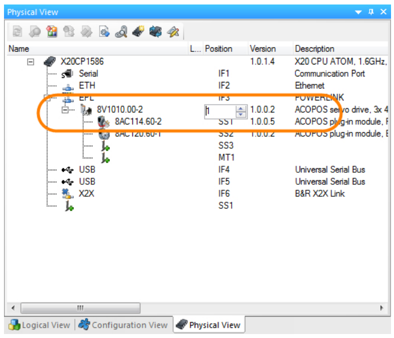


Figure 31: Changing the node number in the Physical View

Generating data for a CompactFlash card – Checking LED status indicators

The modified hardware configuration can be loaded to the target system using a CompactFlash card or via the online connection. When the X20 controller is restarted, all parameters will then be transferred to the drive. Once the configuration has been transferred and the restart is complete, the "READY" LED on the drive should be lit green.

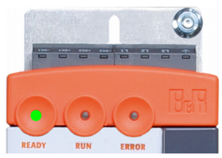


Figure 32: ACOPOS servo drive is ready for operation

10 Additional encoder options and plug-in cards can be connected to available slots on ACOPOS and ACOPOSmulti systems.

The location and meaning of the LED status indicators for the various devices in the B&R drive solution are explained in the respective user's manual.

Project management \ Hardware management \ Physical View \ Editing operations \ Changing the node number

Automation Software \ Getting started \ Creating programs with Automation Studio \ First project with an X20 CPU \ Ethernet settings on the PC

3.3 Preparing a drive and performing movements

The hardware configuration for the X20 controller and drive has been prepared in Automation Studio and transferred to the controller. Now it's time to get the motor moving.

Open the NC Test window

The NC Test window in Automation Studio can be used to put any drive axis into operation. It can be opened by right-clicking on the drive in the Physical View or System Designer and selecting "Test" from the shortcut menu.

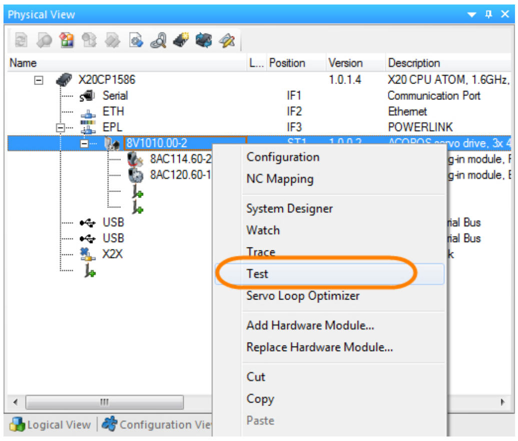


Figure 33: NC Test is opened from the shortcut menu of the ACOPOS servo drive

Depending on the ACOPOS module, it is possible to control multiple motors and virtual axes[11]. Select the correct axis object (in this case "gAxis01").

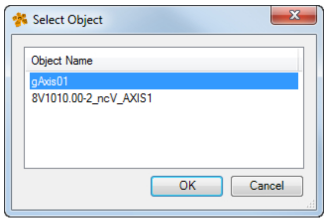


Figure 34: Selecting the axis object

[11] A virtual axis is offered for each real axis. Dual-axis modules therefore have two virtual axes. A total of four axis objects are displayed in the selection dialog box.

Figure 35: Confirmation dialog box for opening NC Test in exclusive mode

When you open NC Test, a confirmation dialog box appears. Here you should select "Exclusive Mode". An explanation of the different modes can be found in later sections (5 "Commissioning and diagnostics" on page 31).

Switching on the controller and performing homing

After opening the NC Test window, there are a number of actions that can be selected in the command interface. To execute an action, select it in the command structure and press **<Enter>** or click the icon in the menu bar to execute it.

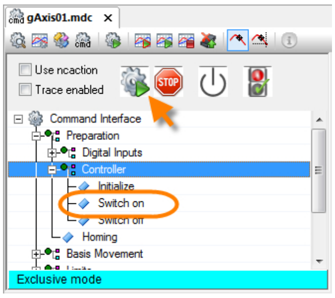


Figure 36: Switching on the controller

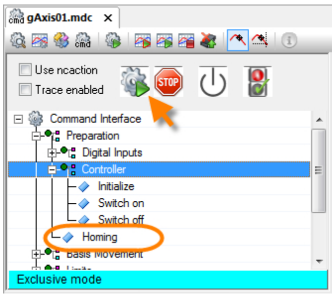


Figure 37: Homing

Perform a movement.

Once the controller has been switched on and homing is complete, the drive is ready to perform movements. Basic movement commands can be used to rotate the motor clockwise, for example.

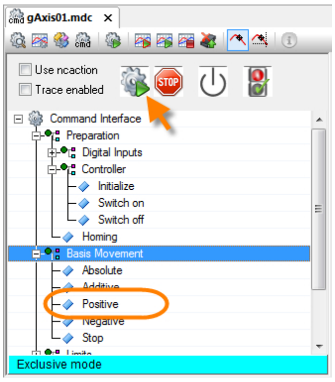


Figure 38: Starting a movement in the positive direction

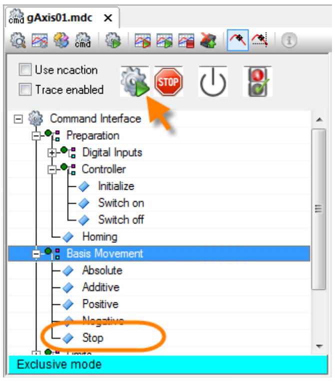


Figure 39: Stopping a movement

Once you have completed these steps, it will be possible to move the motor connected to the drive. A detailed description of all the windows and options in the NC Test environment can be found in one of the later sections (5 "Commissioning and diagnostics" on page 31).

Automation software \ Getting started \ Creating a motion application in Automation Studio \ Motion application example \ Moving an axis in the test window

Motion \ Commissioning

Motion \ Diagnostics \ NC Test \ Command interface

3.4 Task definition: Creating your first project

In this section, we will use the Automation Studio help system to create a motion project, transfer it to a CompactFlash card and control the axis using NC Test.

Exercise: Creating a new project with the support of the help system

The corresponding section of the help system can be opened directly from the start page in Automation Studio.

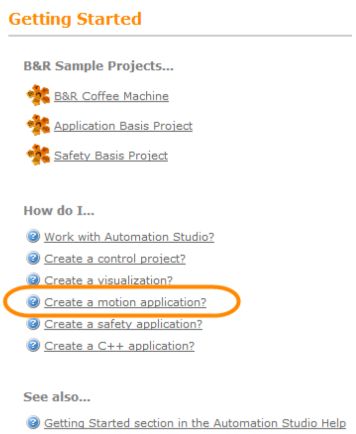


Figure 40: The Automation Studio start page - Getting Started

1) Click on the "How do I create a motion project?" link on the start page.
2) Select the "First motion application" example.
3) Work through each of the steps.
   - Create an Automation Studio project
   - Add an ACOPOS servo drive
   - Generate CompactFlash data
   - Open the NC Test window
   - Initialize the drive, turn on the controller and perform a homing procedure.
   - Perform a movement.

Automation software \ Getting started \ Creating a motion application in Automation Studio

Motion \ Project development \ Motion control \ Setting up an axis

4 COMPONENTS OF THE MOTION CONTROL SYSTEM

When you add a drive in the Physical View or System Designer, the Drive Configuration wizard in Automation Studio creates a number of configuration objects in the Logical View. For each axis, a package is added to the Logical View with the same name as the axis reference. This package contains the configuration objects for the axis' NC object.

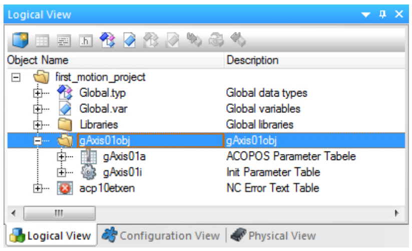


Figure 41: Package for axis reference "gAxis01"

Configuration objects are managed and accessed in the same way, regardless of the hardware being used. As a result, it makes little difference what type of drive technology is in use.

A drive configuration consists of the following components:

- 4.1 "Configuration modules"
- 4.2 "System settings"

The image below illustrates where individual configuration modules and system settings are stored during runtime, as well as where changes made to parameter values will have an effect.

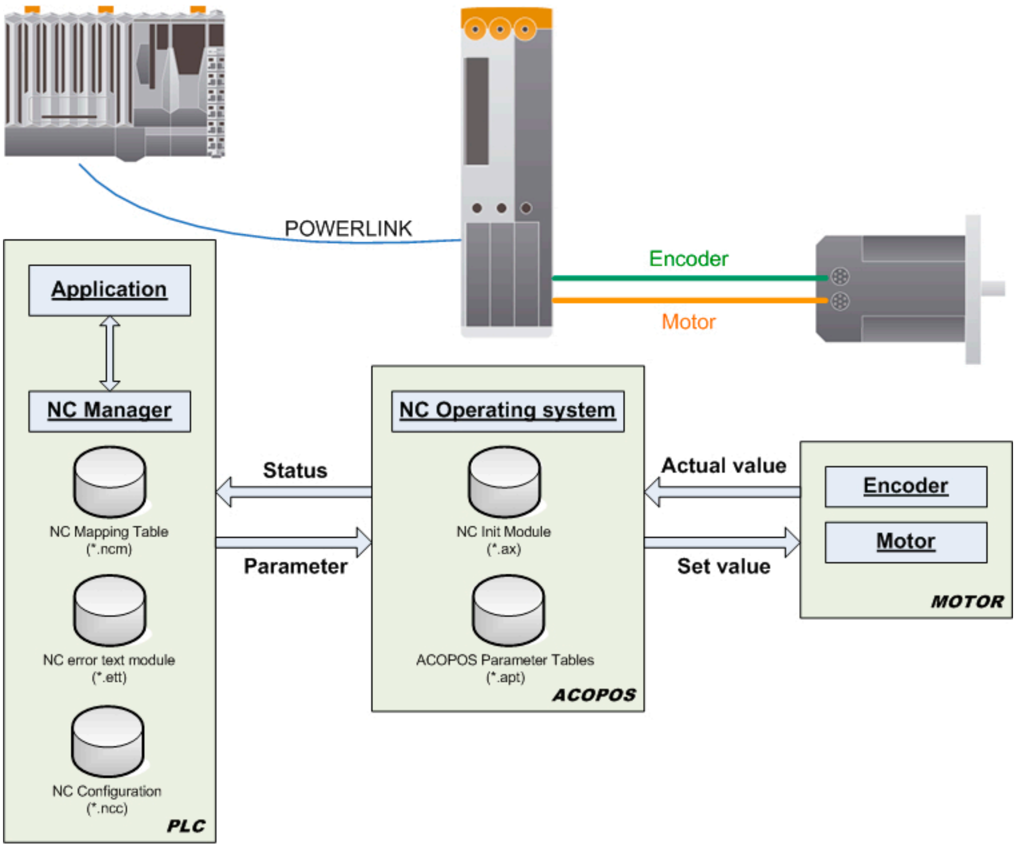


Figure 42: Overview of motion control system components

Motion \ Project development \ Motion control \ Configuration modules

4.1 Configuration modules

The modules created when adding a drive will be described in more detail in the following sections.

4.1.1 NC Init module

The NC Init module contains basic axis parameters used by NC Manager to initialize axis referencing when the controller is started.

The NC Init module is accessed from the Logical View. Double-clicking on the corresponding file with the extension "**.ax**" opens a window where the initialization parameters (e.g. speed/acceleration values) can be changed.

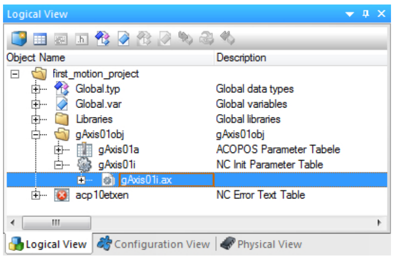


Figure 43: The NC Init module in the Logical View

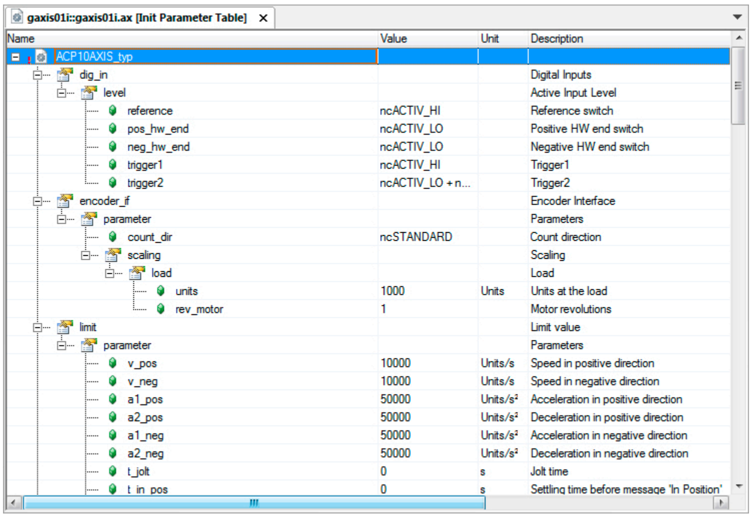


Figure 44: NC Init module

The structure of the NC Init module mirrors the data structure of the NC object. The parameters it contains are divided into groups. These groups and the description of the configuration options, as well as the corresponding data structure elements, are described in the Automation Studio help system. A more detailed description of the NC Init module can be found in one of the later sections (5.2 "Parameter management" on page 36).

Motion \ Project development \ Motion control \ Configuration modules \ NC Init module

Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS"

Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Overview \ Data type "ACP10AXIS_typ"

4.1.2 ACOPOS parameter table

Unlike the NC Init module, the ACOPOS parameter table allows you to set specific parameters (such as the nominal DC bus voltage). These parameters are also referred to as parameter IDs, or ParIDs.

The ACOPOS parameter table can be opened by double-clicking on it in the Logical View. This file can be recognized by its "**.apt**" extension.

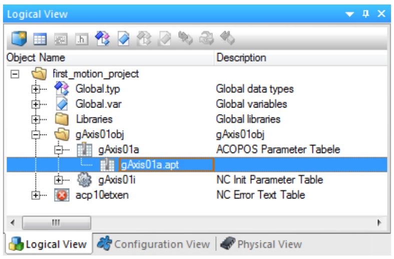
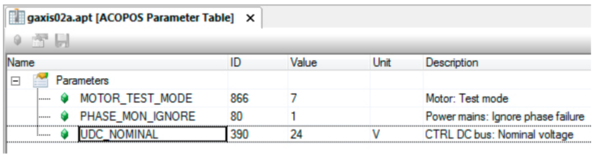


Figure 45: ACOPOS parameter table in the Logical View



gaxis02a.apt [ACOPOS Parameter Table]

Name	ID	Value	Unit	Description
Parameters				
MOTOR_TEST_MODE	866	7		Motor: Test mode
PHASE_MON_IGNORE	80	1		Power mains: Ignore phase failure
UDC_NOMINAL	390	24	V	CTRL DC bus: Nominal voltage

Figure 46: ACOPOS parameter table

Motion \ Project development \ Motion control \ Configuration modules \ ACOPOS parameter table

Motion \ Reference manual \ ACP10 \ ACOPOS parameter IDs

Exercise: ParIDs in the Automation Studio help system

The Automation Studio help system can be used to search for ParIDs.

Find the following parameters:

- Temperature of the temperature sensor, current motor temperature
- Current motor torque
- Voltage on the DC bus
- Temperature of the temperature sensor in the ACOPOS power unit

1) Open the Automation Studio help system.

2) In the help system, select the **Motion** section.

3) Select the "Parameter IDs / Drive functions" hotspot.

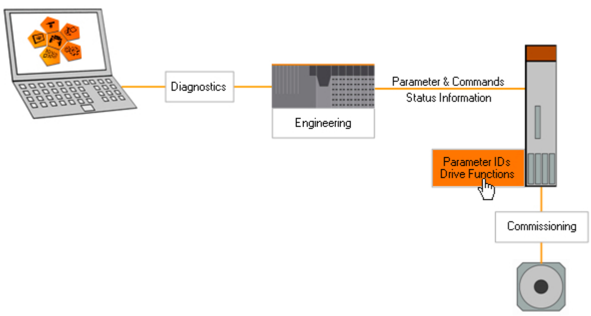


Figure 47: Possible selections in the Automation Studio help system

4) Search for the required parameters.

? Motion

Motion \ Reference manual \ ACP10 \ ACOPOS parameter IDs

4.1.3 NC error text table

The NC error text table contains all the texts that correspond to the drive's error numbers. When using the Drive Configuration wizard in Automation Studio, it is necessary to specify the name and language first.

The NC error text table can be opened by double-clicking on it in the Logical View. This file can be recognized by its "*.ett*" extension.

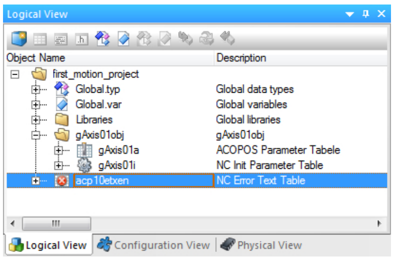


Figure 48: NC error text table in the Logical View

The ability to select a language for error texts allows the NC error text table to provide useful information about axis errors in addition to the actual error number, for example when reading axis errors using the PLCopen function block MC_ReadAxisError().

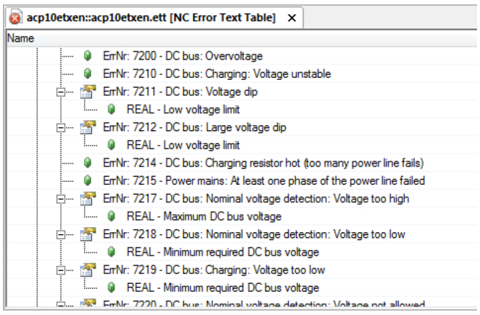


Figure 49: NC error text table

? Motion \ Project development \ Motion control \ NC error text table

4.1.4 NC Manager configuration

NC Manager is configured using the NC Manager configuration. These settings are separate from the actual hardware.

More information about this topic can be found in section 4.2 "System settings" on page 30.

? Motion \ Project development \ Motion control \ Configuration modules \ NC Manager configuration

4.1.5 NC mapping tables

A global variable is created each time a drive is configured using the wizard in Automation Studio.

The address of this variable is called the axis reference. It is used to form the link between the application and the physical drive.

In the NC mapping table, each drive can be assigned an NC Init module and an ACOPOS parameter table. You can also enter additional ACOPOS parameter modules manually by separating them with a comma (,). They will then be sent to the drive when the controller is restarted.

The NC mapping table assigns the following configuration modules to an axis:

- NC Init module
- ACOPOS parameter table
- Axis reference (gAxis01)

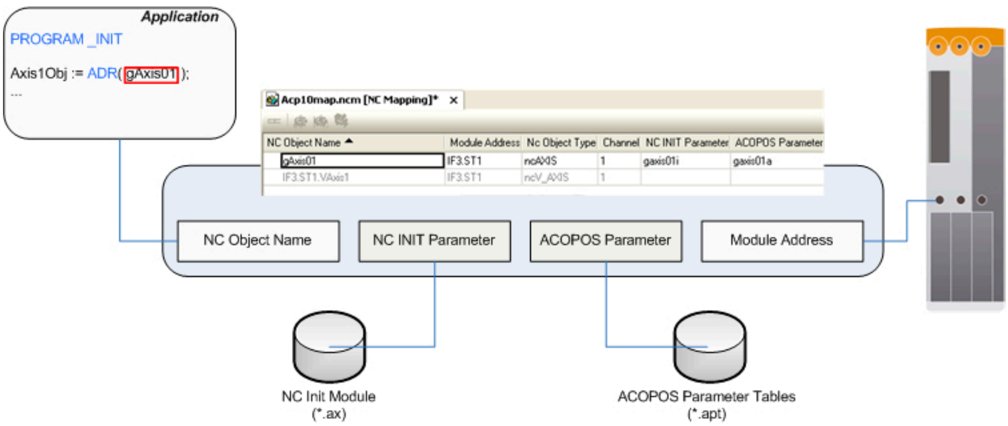


Figure 50: Linking the drive with an axis variable

The NC mapping table can be opened from the shortcut menu for the drive in the Physical View. It can also be opened by double-clicking on it in the Configuration View. This file can be recognized by its "**.ncm**" extension.

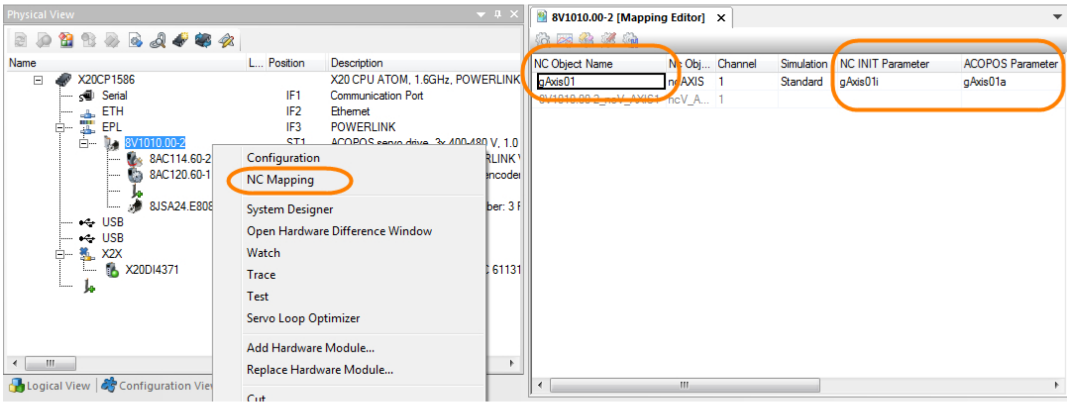


Figure 51: Opening the NC mapping table from the Physical View: assigning an axis reference, NC Init module and ACOPOS parameter table

Motion \ Project development \ Motion control \ Configuration modules \ NC mapping table

4.2 System settings

The NC Manager configuration can be opened by double-clicking on it in the Configuration View. This file can be recognized by its "*.ncc*" extension.

The system settings are used to configure the NC Manager. Some examples of some of these settings include the size of the buffer for the network command trace (5.4.2 "Network command trace" on page 45) and restart behavior after a power failure.

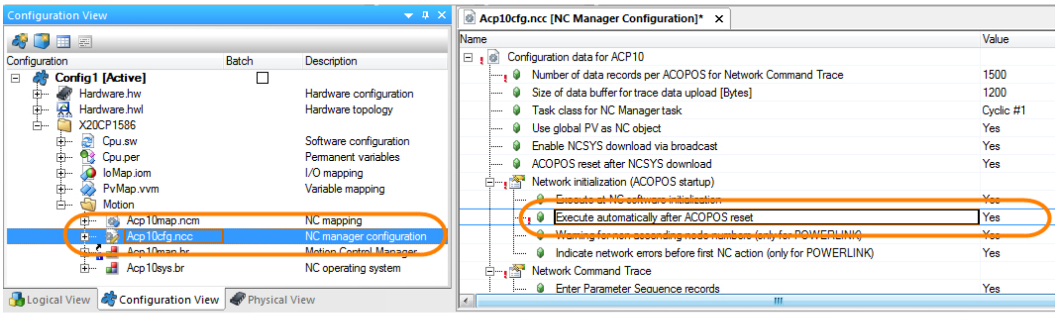


Figure 52: Opening the NC Manager configuration in the Configuration View: configuring the restart behavior after a power failure

Communication between the NC Manager and NC operating system stops when the drive's control voltage is lost. Change the setting "*Execute automatically after ACOPOS reset*" to "Yes" if communication should be reestablished automatically.

Motion \ Project development \ Motion control \ Configuration modules \ NC Manager configuration

5 COMMISSIONING AND DIAGNOSTICS

Two environments are available for quickly and easily commissioning and troubleshooting the drive configuration. The System Diagnostics Manager is accessed via the web server integrated in Automation Runtime. NC Test is part of Automation Studio. The main focus of the following chapter is to describe the functions and properties of the BC Test window.

Figure 53: Commissioning and diagnostics

System Diagnostics Manager – SDM

SDM can be used to evaluate drive status data such as current speed or limit switch positions. Other functions include starting trace recordings, uploading the resulting data and uploading the network command trace.

This tool can be accessed via the controller's IP address using any standard web browser.

Features of SDM:

- Read-only access to the controller
- Viewing the drive status
- Viewing axis errors
- Downloading trace configurations and uploading trace data
- Uploading a network command trace

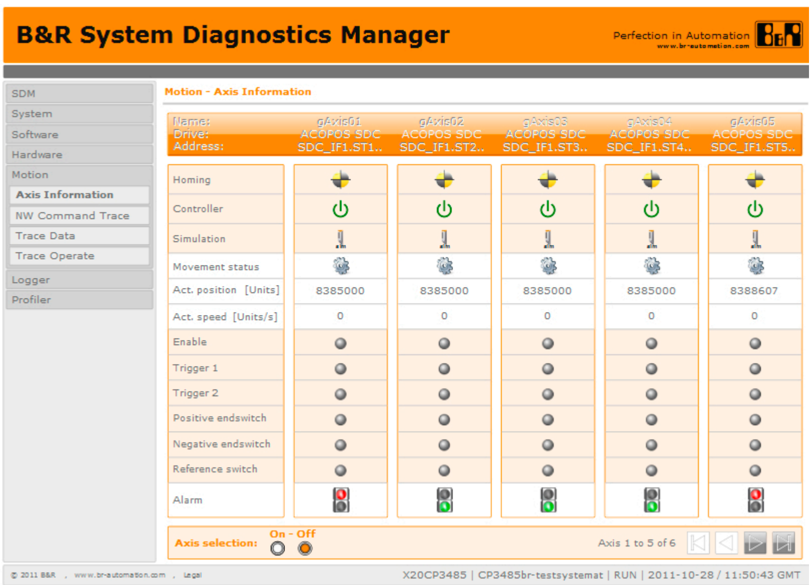


Figure 54: Troubleshooting with the System Diagnostics Manager

Diagnostics and service \ Diagnostics tools \ System Diagnostics Manager

NC Test

The NC Test window in Automation Studio offers a wide range of functions that are helpful when commissioning or troubleshooting a drive axis.

NC Test functions:

- Preparing a drive and issuing commands via the command interface
- Accessing and managing drive parameters
- Viewing drive status values
- Tracing parameters on the drive in real time
- Tracing network communication with the drive

A connection between Automation Studio, controller and drive is required in order to use NC Test.

Opening the NC Test window

Use the shortcut menu in the drive amplifier to select and open the NC Test window in the Physical View.

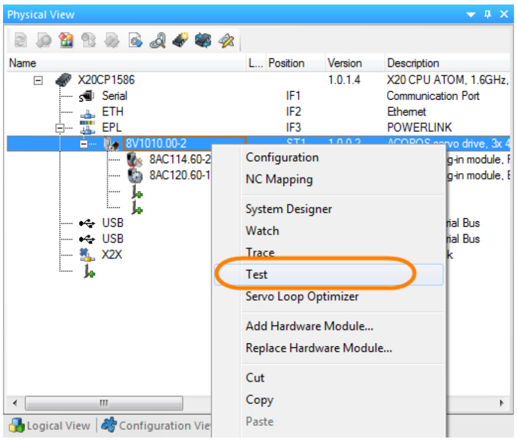


Figure 55: Opening the NC Test window from the Physical View

Since some B&R servo drives allow you to manage multiple axes, a dialog box opens to select the correct NC object. Select the "gAxis01" axis reference to continue.

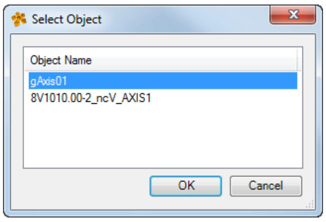


Figure 56: Selecting the NC object

Next, a dialog box opens where you can select the mode to be used when opening the NC Test window. You need to decide whether you want the NC Test window to have **parallel** or **exclusive** access to the axis. In exclusive mode, commands from the application are not passed on to the axis. Instead, operation takes place exclusively from the NC Test window.

Figure 57: NC Test mode

Appearance of the NC Test window

The NC Test window is divided into four sections.

NC Test window area	See section
1. Command interface	5.1 "Command interface" on page 34
2nd Parameter window	5.2 "Parameter management" on page 36
3. NC Watch	5.3 "Viewing axis information" on page 38
4. NC Trace	5.4 "Trace functions" on page 40

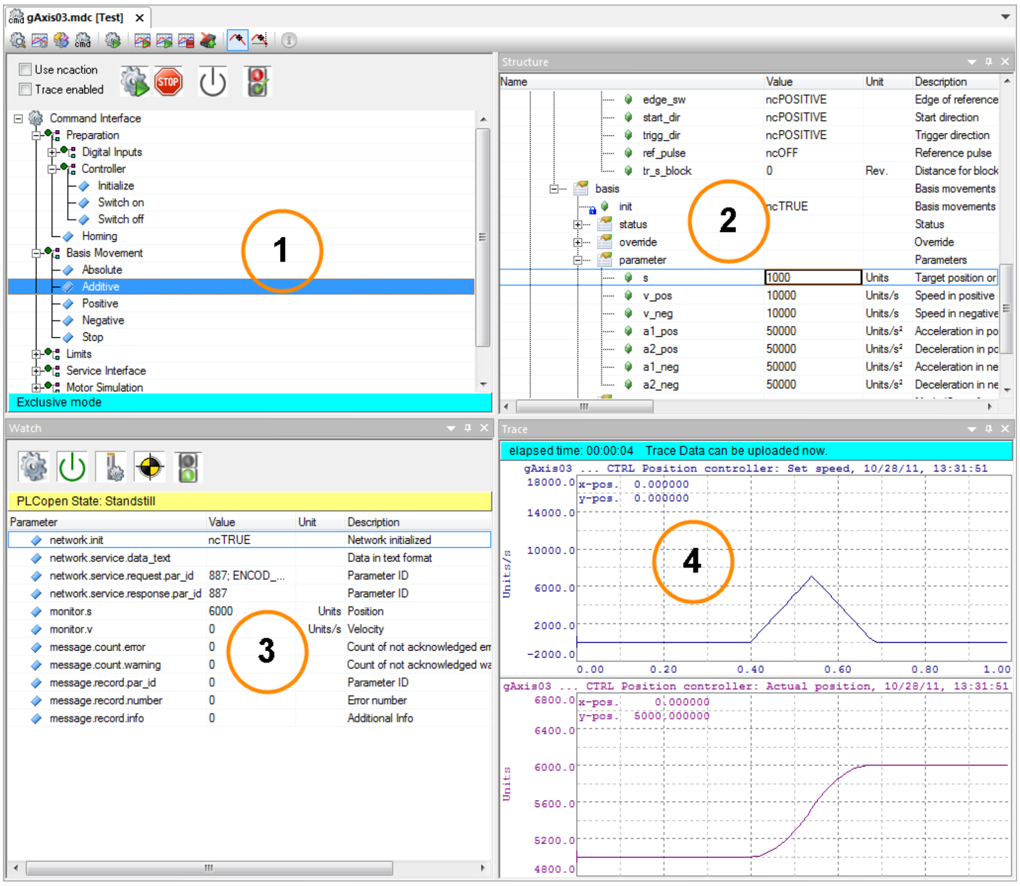


Figure 58: Structure of the NC Test window

If the NC Test window is closed during an active movement, then the user must choose whether the movement should be stopped or not.

Motion \ Diagnostics \ NC Test

5.1 Command interface

The command interface is used to control an axis.

The command interface is divided into the following sections:

- Preparation
- Basis movements
- Limit values
- Service interface
- Motor simulation
- Setup

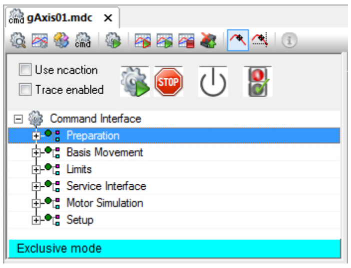


Figure 59: Command interface

Explanation	Symbol
Execute command: Executes the command selected in the command interface	
Stop movement: Stops the current movement	
Controller on/off: Switches the controller on and off	
Acknowledge axis error: The traffic light icon can be used to acknowledge axis errors in NC Watch.	

Table 5: Explanation of icons in the command interface

Preparation

Before you can move an axis, you must first activate the drive's controller using the **Switch on** command. The axis must be homed using the **Homing** command. It is then possible to perform basic movements.

The **Initialize** command can be used if parameters need to be initialized.

Digital inputs (limit switch, E-stop) can only be initialized when the controller is turned on. The controller parameters can also be initialized when the controller is turned on. A later section describes where the parameters are located and how to change them (5.2 "Parameter management").

Service interface

The service interface can be used to read individual parameters from the drive or write them to it.

Limit values

The ***Initialize*** command can be used for any limits that have been changed and must be initialized. This requires that the controller be switched off.

Motor simulation

If you are simulating a motor because it is not physically available or cannot be moved for tests, use the **"Switch on"** command to enable motor simulation.

When the simulation function is no longer needed, it can be disabled using **Switch off**.

Setup

The setup section is used to determine controller parameters using the integrated autotuning procedure.

Motion \ Diagnostics \ NC Test \ Command interface

Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS"

- Digital inputs
- Basis movements
- Limit values
- Simulation

Motion \ Commissioning \ Autotuning

Exercise: Reading ParIDs

Determine the following drive parameters:

- Motor ambient temperature (ParID: 668; TEMP_MOTOR_AMB)
- Actual position (ParID: 111; PCTRL_S_ACT)

1) Search for ParIDs in the Automation Studio help system.

Motion \ Reference manual \ ACP10 \ ACOPOS parameter IDs \ Overview

2) Reading the ParID via the service interface

3) Read out the values using NC Watch.

The values of the specified ParIDs are determined directly on the drive.

5.2 Parameter management

The drive parameters are managed in the Parameter window. The drive parameters are grouped by individual drive functions.

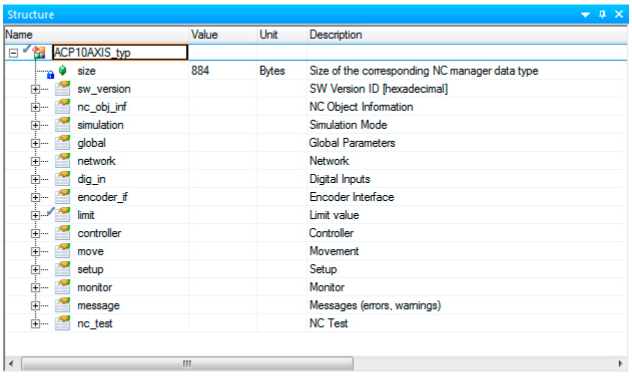


Figure 60: Parameter window

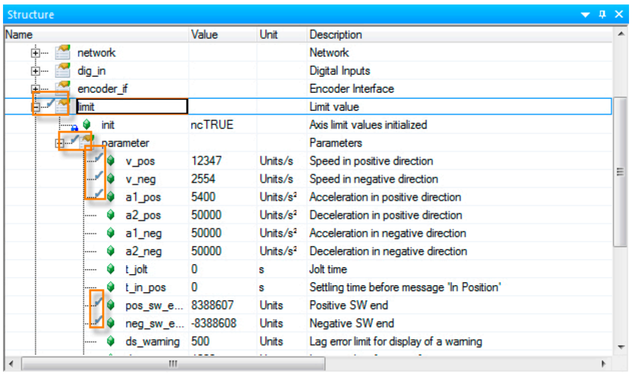


Figure 61: Changed parameters

The parameter structure contains parameters for the NC Init module.

Values that are changed manually are compared with the saved parameters and identified with a black check mark.

Any initial parameters that were changed can be applied to the NC Init module.

If you do not save your work explicitly, then a dialog box will appear when you close NC Test. This allows you to choose whether or not the changed INIT values will be saved to the NC Init module.

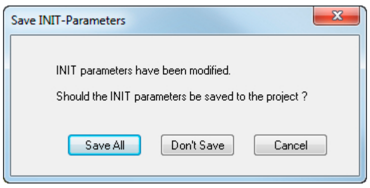


Figure 62: Closing the NC Test window

The parameters shown can be saved to the NC Init module. The data displayed matches the data structure of an NC object. Consult the Automation Studio help system for detailed descriptions of the grouped parameters and elements of the data structure.

> ?
> Motion \ Diagnostics \ NC Test \ Parameter window
>
> Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Overview \ Data type "ACP10AXIS_typ"

Exercise: Homing mode - data structure, NC Init module, parameter window

In the following exercise, you will change the homing mode of the motor from direct homing to homing to a limit switch. The corresponding elements of the data structure and different homing mode options are described in the Automation Studio help system. This information can be found by expanding the "Data structure" section at the top of the "Homing procedure" page in the help system.

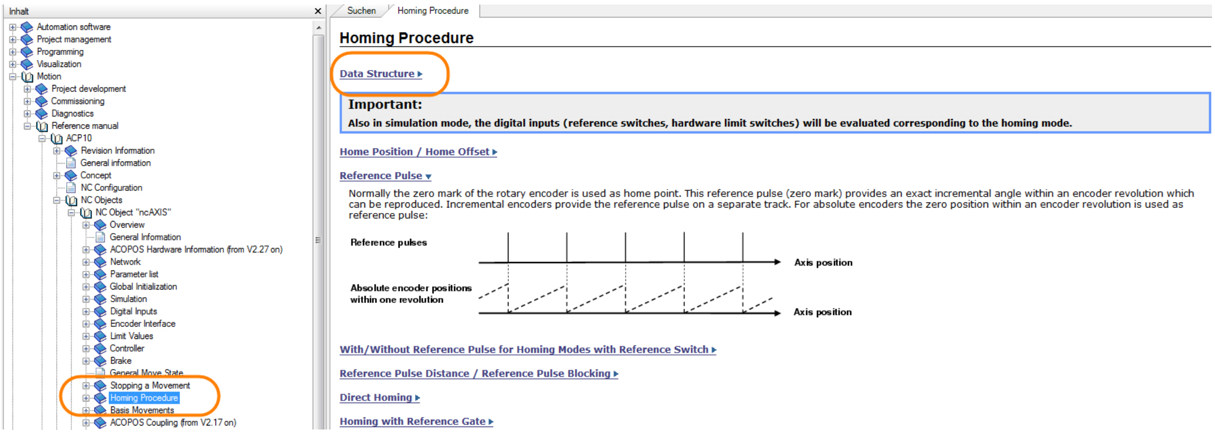


Figure 63: Description of the various homing modes and corresponding elements of the data structure in the Automation Studio help system

1) Find the description of the homing procedure in the Automation Studio help system.
2) Configure the homing mode according to the description in the help system.
3) Initialize the parameters using the command interface in NC Test.

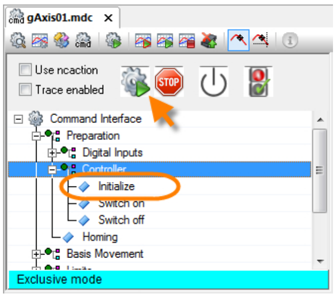


Figure 64: Initializing the parameters via the command interface

4) Switch on the controller.
5) Perform a homing procedure.

5.3 Viewing axis information

NC Watch shows the current status of an NC object. NC Watch is part of the NC Test window, but it can also be opened from the drive's shortcut menu in the Physical View.

The NC Watch window is divided into three sections:

1. Object states
   The icons indicate the state of a drive axis. A status message which provides information about the PLCopen status of the axis.

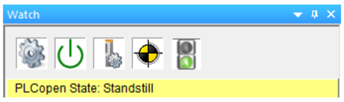


Figure 65: Object states in NC Watch

2. Object data
   This is where current values can be read from the drive. Use the shortcut menu to add additional values.
3. Error text
   The traffic light icon allows you to switch between viewing object data and error texts.

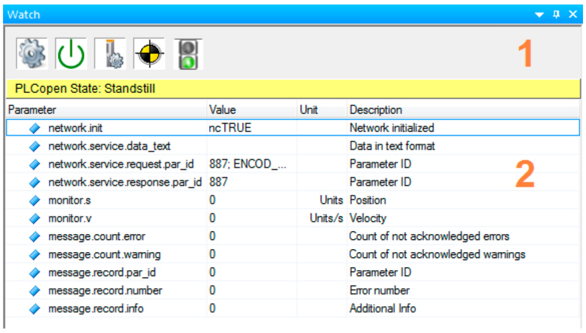


Figure 66: Viewing object data in NC Watch

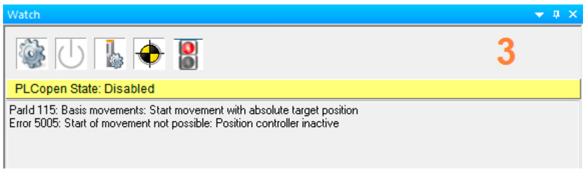


Figure 67: Viewing error texts in NC Watch

Motion \ Diagnostics \ NC Watch

5.3.1 Viewing object states and error texts

Icons in the toolbar display the most important drive states. Connection errors between an NC object and the controller are indicated by a red "X".

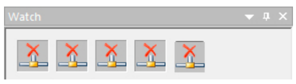


Figure 68: NC Watch: Problem with communication

If a connection is established, then the icons appear as shown in this image. The object state icons are shown in color or grayed out depending on whether the drive is switched on and homed.

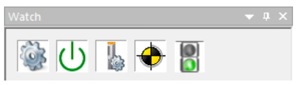


Figure 69: NC Watch: Status icons

The traffic light icon is used to toggle what is shown in the NC Watch window. It can be used to show the error text for a pending error.

No error pending on the axis:

Figure 70: Green light

Error pending on the axis:
Click on the icon to view the errors listed in the NC Watch window.

Figure 71: Red light

If there is an error, a detailed error description will appear below the icons.

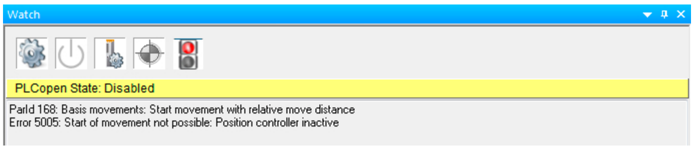


Figure 72: Error texts in NC Watch

Figure 73: Error acknowledgment using the icons in the command interface

A button in the command interface can be used to acknowledge one or more errors.

If an action causes multiple errors simultaneously, they will be acknowledged in order.

Motion \ Diagnostics \ NC Test \ NC Watch

5.3.2 Viewing object data

The window for object data in NC Watch contains the most important elements from the parameter structure by default. The user can modify the list to include other parameters. More values can be added using the shortcut menu.

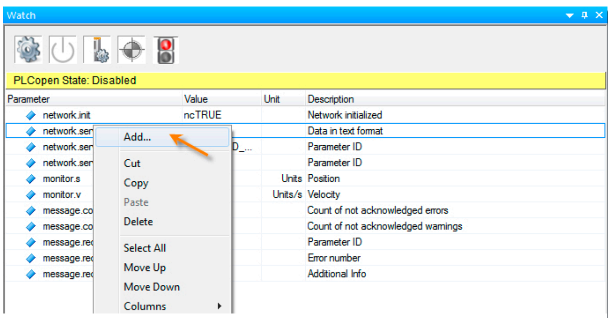


Figure 74: Adding values from the shortcut menu

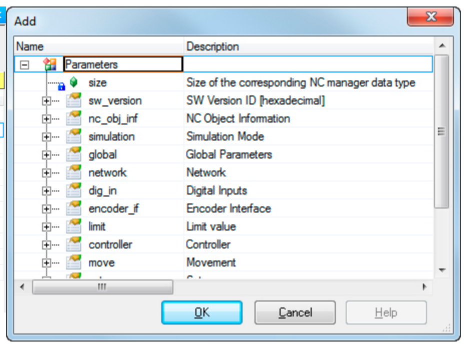


Figure 75: Selecting values from the parameter structure

> When you close NC Watch, the list of object data is saved. The same list will be shown the next time you open NC Watch.

5.4 Trace functions

The NC Test window contains two tools for data collection.

NC Trace can be used to record drive parameters on the drive in real time and display them in Automation Studio. The network command trace can be used to record the drive communication (i.e. the commands to the drive and status data).

5.4.1 NC Trace

Drive parameters are recorded in real time directly on the drive. The data collected from the ACOPOS servo drive is loaded into Automation Studio via the controller and displayed graphically.

Features of NC Trace:

- Recording speed, acceleration and current values
- Checking the motor load
- Checking the thermal load on the ACOPOS
- Checking the positioning sequence
- Multi-axis trace (synchronous recording of parameters from multiple axes)

The following image shows how the results are displayed in NC Trace. In this case, an additive movement was performed. This means the drive receives the command to move a specific relative distance in one direction. The first curve shows the position rising throughout the positioning sequence. The second curve shows the speed characteristic during positioning. The third curve shows the position lag error. This can be reduced by optimizing the controller settings.

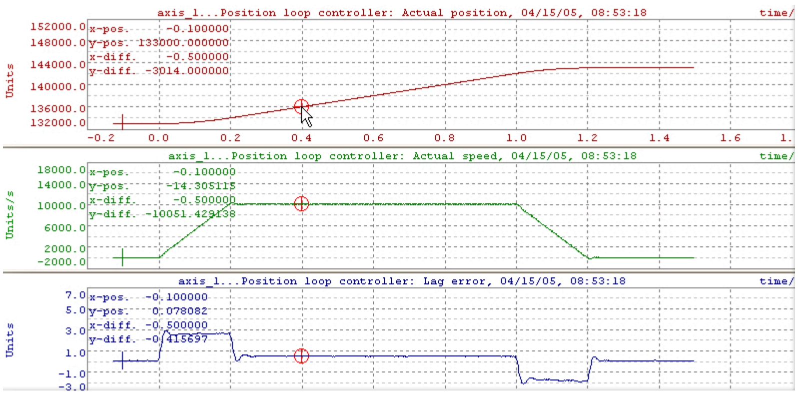


Figure 76: Example of a parameter trace (current position, speed, position lag error)

A number of different tools such as reference cursors and measurement cursors can be used for detailed analysis of this data.

> NC Trace can also be opened from the shortcut menu in the Physical View in a separate window from the NC Test window.

Enable the trace function.

Trace recording can be enabled in the command interface. This can be done, for example, by selecting an additive movement and checking the "Trace enabled" checkbox. The trace will start automatically when the command is executed.

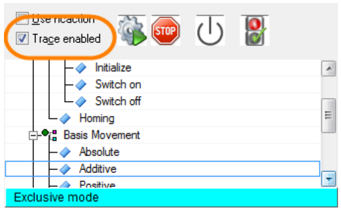


Figure 77: Enabling NC Trace

Saving trace data

Collected trace data can be saved to a file and used again later by **right-clicking** in the NC Trace window and selecting "Save chart data".

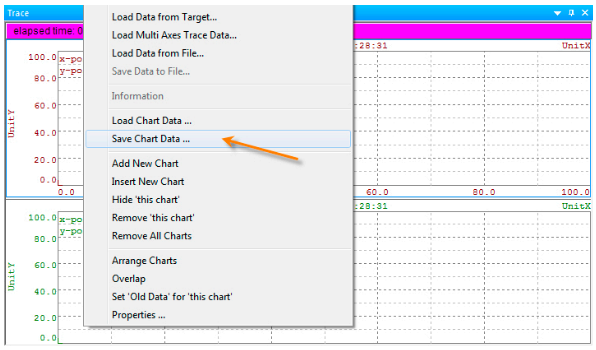


Figure 78: Saving trace data

Configuring a trace

Settings can be adjusted via the NC Trace shortcut menu. This allows you to configure the selection of drive parameters, how the trace recording is triggered and the length of recording.

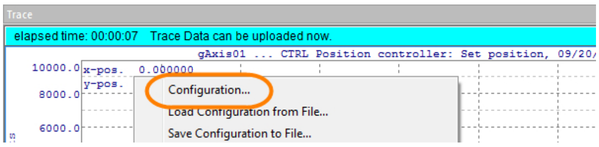


Figure 79: Opening the trace configuration

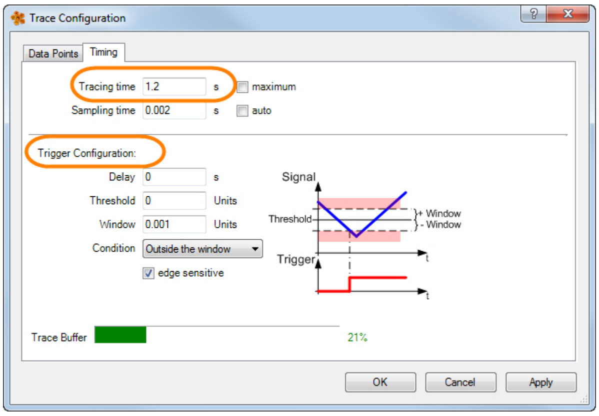


Figure 80: Setting the recording time and trigger behavior

Motion \ Diagnostics \ NC Trace

Motion \ Reference manual \ ACP10 \ ACOPOS parameter IDs

Exercise: Recording lag errors using the NC Trace function

A recording is to be set up for an additive movement using NC Trace.

For example, if the axis is blocked because it is being held in place, this will trigger a movement stop when the maximum lag error is exceeded[12]. An axis error will then be indicated in NC Watch. The maximum speed can be determined based on the recorded data.

As an example, check the following settings in the parameter window:

- s = 10000
- v = 10000

1) Prepare the drive (initialize, switch on the controller, perform a homing procedure)
2) Configure the NC Trace

   Add the parameters:

   - Actual speed / Set speed
   - Lag error

3) Enable the trace function.
4) Perform an additive movement.
5) After the trace is complete, set the measurement cursors for the current speed at the beginning and end.
6) Adjust the chart properties for the current speed curve to display the average speed.
7) Acknowledge any axis errors.
8) Set the average speed as the maximum speed.

The trace indicates that the lag error increases up to the configured maximum value before triggering a movement stop and pushing the drive into an error state.

The average value for an axis can be displayed in the area between the measurement cursors by making a simple change to the chart properties.

With the modified (reduced) maximum speed, the lag error remains within its limits so that the movement stop is no longer triggered.

Exercise: Tracing the deceleration ramp

Use a trigger to start recording the deceleration ramp.

The following parameters must be recorded:

- Actual speed / Set speed
- Lag error

The trigger should have the following conditions:

- The set speed is within the range v_max and v_max – 10%.
- The command for stopping the movement has been issued.

1) Set the trigger conditions.
2) Perform a movement in the positive direction.

[12] If the ACOPOS drive is supplied with 24 V for testing purposes, then excessive acceleration or speed will trigger the movement to be aborted when the maximum lag error is exceeded.

3) Stop the movement.

4) Evaluate the trace recording.

The drive's deceleration characteristics during braking can be seen in the trace recordings.

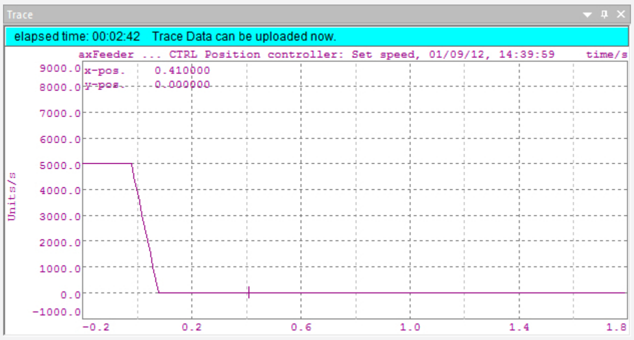


Figure 81: Deceleration ramp for the set speed in NC Trace

Properties of curves and curve calculation

NC Trace allows you to edit the display properties of trace recordings. It is possible to adjust how the curve is displayed, the colors used, the dimensions of the X and Y axes, visibility and the axis labels. The properties dialog box can be opened by selecting "Properties" from the NC Trace shortcut menu.

Trace curves can be used as a basis for calculations in NC Trace. For example, if you want to add the values from multiple trend curves, you can use a new diagram. In the NC Trace shortcut menu, use the "Add new chart" option to add a new blank diagram.

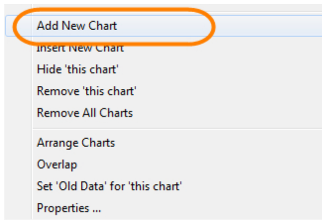


Figure 82: Adding a new diagram

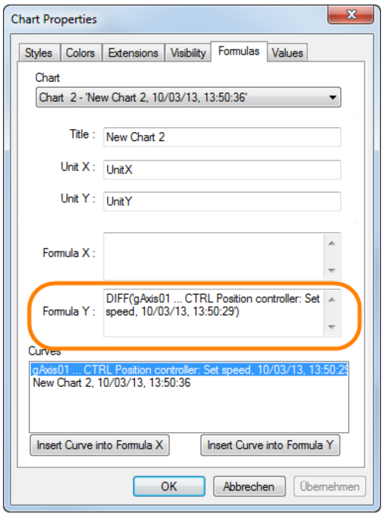


Figure 83: Calculating the first derivative of speed using the "DIFF" function

Different formulas can be entered in the properties window for the new diagram to perform calculations for the X and Y axes. In the image, the "DIFF" function is being used to calculate the first derivative of the axis speed. The result is the axis acceleration. The text fields in the dialog box can be used to modify the axis labels and diagram units.

A complete listing of the permitted functions, operators and syntax rules is provided in the Automation Studio help system.

These settings can be used to show two curves, one above the other. The upper curve shows the original drive speed recorded by the trace function. The recorded movement covered two motor revolutions. The newly added diagram shows the acceleration of the individual movement phases.

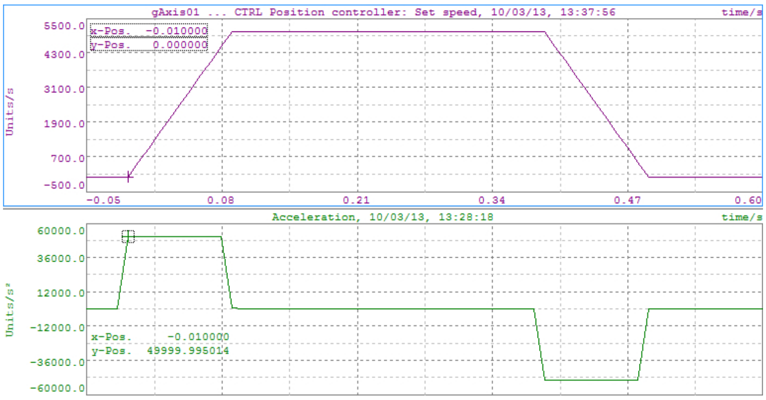


Figure 84: Above: Drive speed, bottom: Current acceleration

Motion \ Diagnostics \ NC Trace \ Curve properties
Motion \ Diagnostics \ NC Trace \ Curve calculations

Exercise: Configure the step time, record movement, determine acceleration

A jolt time can be configured for the axis limit values in order to preserve the mechanical drive components used for accelerating and decelerating the drive axis.

Two trace recordings will be performed. The first will record the current speed and display the acceleration in an additional diagram. The second takes place with a configured jolt time. The results of the recordings will then be compared.

1) Switch on the controller and perform a homing procedure.
2) Configure and enable the trace.
3) Perform an additive movement over two motor revolutions.
4) Add a new diagram and calculate the acceleration based on the speed.
5) Set the jolt time (t_jolt) to a value such as 70 ms.
6) Initialize the axis limit values.
7) Perform the same movement again.
8) Compare the results of the two recordings.

The top curve clearly shows the changed characteristic of the speed. The rounded corners are produced by using the jolt time. Likewise, jumps in the acceleration characteristic have also become smooth – they have become trapezoidal.

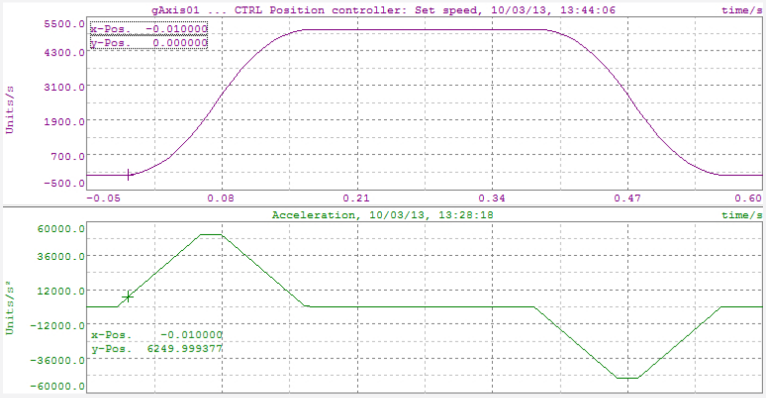


Figure 85: Above: Drive speed, bottom: Current acceleration

5.4.2 Network command trace

You can record communication between the controller and the drive using the Network Command Trace. This allows precise analysis of the time sequence of network communication between the controller and the drive.

This also means that any error responses to the commands between the drive and application can be viewed even after the error has been acknowledged.

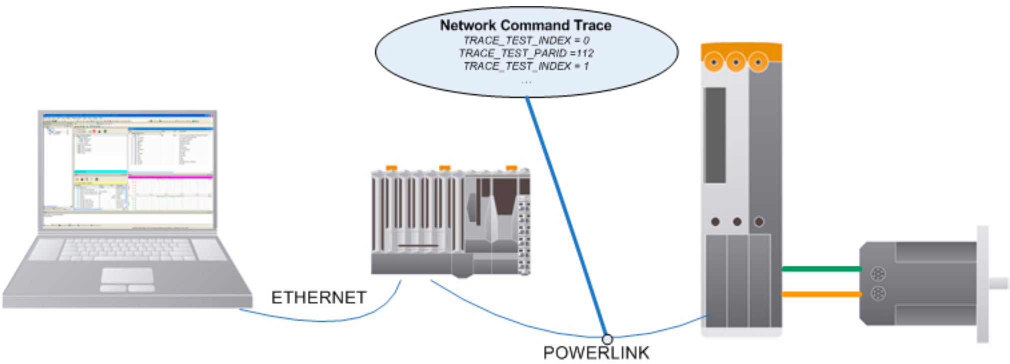


Figure 86: Network command trace

Network command trace is an extension of NC Trace and can be opened from the NC Trace shortcut menu Use the shortcut menu to switch to the cyclic Trace.

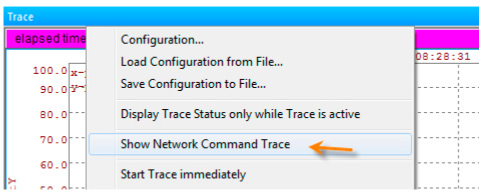


Figure 87: Opening network command trace

Network command trace is always enabled. The size of the configured recording buffer determines how many entries can be viewed. Records can be loaded using the shortcut menu.

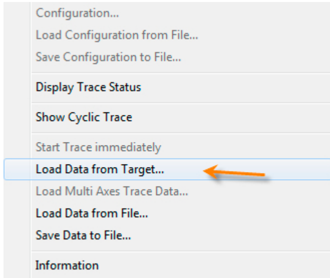
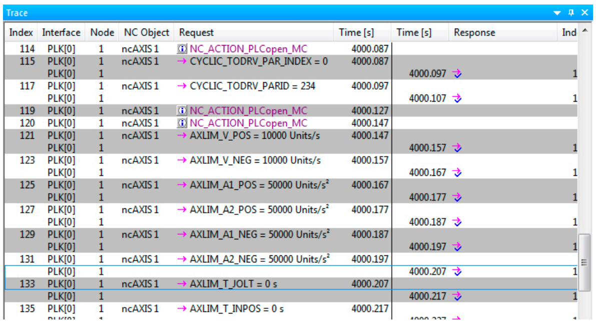


Figure 88: Loading the data



Trace

Index	Interface	Node	NC Object	Request	Time [s]	Time [s]	Response	Ind
114	PLK[0]	1	ncAXIS 1	NC_ACTION_PLCopen_MC	4000.087			
115	PLK[0]	1	ncAXIS 1	→ CYCLIC_TODRV_PAR_INDEX = 0	4000.087			
	PLK[0]	1				4000.097		1
117	PLK[0]	1	ncAXIS 1	→ CYCLIC_TODRV_PARID = 234	4000.097			
	PLK[0]	1				4000.107		1
119	PLK[0]	1	ncAXIS 1	NC_ACTION_PLCopen_MC	4000.127			
120	PLK[0]	1	ncAXIS 1	NC_ACTION_PLCopen_MC	4000.147			
121	PLK[0]	1	ncAXIS 1	→ AXLIM_V_POS = 10000 Units/s	4000.147			
	PLK[0]	1				4000.157		1
123	PLK[0]	1	ncAXIS 1	→ AXLIM_V_NEG = 10000 Units/s	4000.157			
	PLK[0]	1				4000.167		1
125	PLK[0]	1	ncAXIS 1	→ AXLIM_A1_POS = 50000 Units/s²	4000.167			
	PLK[0]	1				4000.177		1
127	PLK[0]	1	ncAXIS 1	→ AXLIM_A2_POS = 50000 Units/s²	4000.177			
	PLK[0]	1				4000.187		1
129	PLK[0]	1	ncAXIS 1	→ AXLIM_A1_NEG = 50000 Units/s²	4000.187			
	PLK[0]	1				4000.197		1
131	PLK[0]	1	ncAXIS 1	→ AXLIM_A2_NEG = 50000 Units/s²	4000.197			
	PLK[0]	1				4000.207		1
133	PLK[0]	1	ncAXIS 1	→ AXLIM_T_JOLT = 0 s	4000.207			
	PLK[0]	1				4000.217		1
135	PLK[0]	1	ncAXIS 1	→ AXLIM_T_INPOS = 0 s	4000.217			

Figure 89: Example of a network command trace

The Network Command Trace shows a list of entries. This provides information about each parameter that the NC Manager sent to the drive. Information about the drive's corresponding response is also provided. This data can be saved using the shortcut menu or loaded using the System Diagnostics Manager (SDM).

Motion \ Diagnostics \ Network command trace

5.5 Determining control settings using autotuning

B&R drive software is based on a cascaded control concept. A position setpoint is provided to the position controller by a setpoint generator that calculates a path profile upon receiving a positioning command. To achieve this position setpoint, the position controller specifies a speed profile. The task of the speed controller is to maintain the speed setpoint as closely as possible.

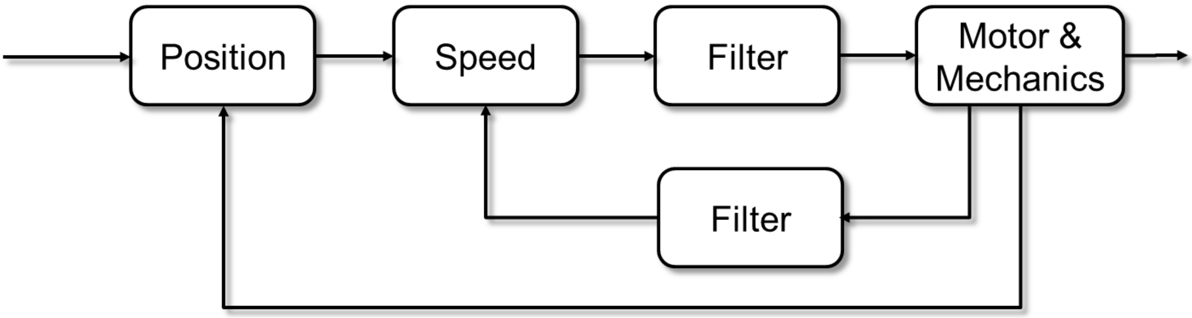


Figure 90: Simplified illustration of the cascaded control concept

The integrated autotuning procedure makes it possible to calculate the control parameters automatically. It is recommended that the parameters for closed-loop control be calculated in the following order: speed controller, followed by position controller. The control settings should then be tested before determining the parameters for the feed forward.

Preparing autotuning

The drive must be operational before autotuning can be carried out. The functionality of the holding brake must then be checked. It is also necessary to check the measured direction of rotation and distance of the encoder. If any deviations are observed, or if another malfunction of the encoder is detected, the encoder must be checked both mechanically and electrically. The encoder should then be phased [13]. The tuning parameters can now be entered:

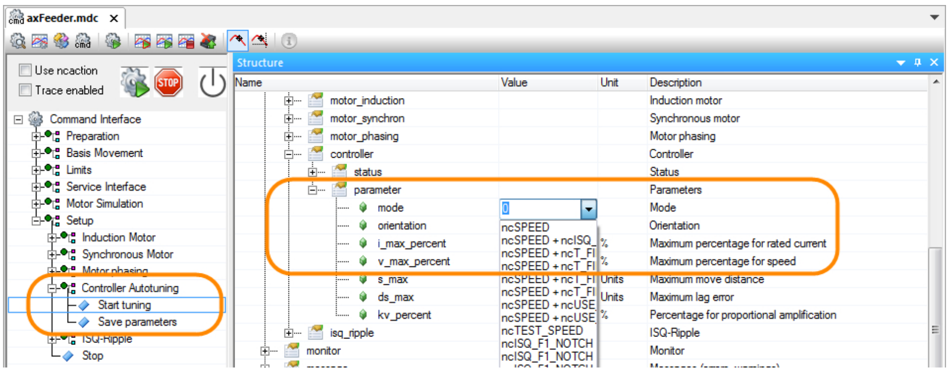


Figure 91: commands for autotuning in the command interface; tuning parameters in the parameter window of the NC Test window.

[13] Phasing is not normally required for B&R motors. Phasing is necessary, for example, if the encoder has been installed at a later time. Motor commissioning is described in detail in "TM460 – Initial Commissioning of Motors".

A suitable NC Trace configuration for analyzing each phase of the autotuning procedure is available in the "Motion \ Commissioning \ Autotuning" section of the Automation Studio help system.

Tune the cooling output

The speed controller's job is to determine the difference between the manipulated variable of the position controller (to which it is subordinate) and the measured speed. This calculates a manipulated variable for the subordinate current controller that works against a deviation in the speed by accelerating.

Selecting the "ncSPEED" autotuning mode[14] and restarting the tuning procedure from the command interface will determine the parameters for the speed controller.

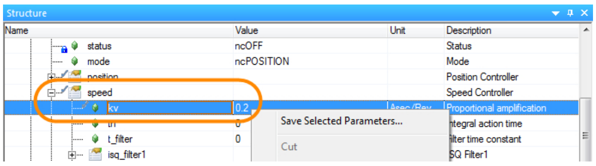


Figure 92: Viewing and saving the parameters in the parameter window

These parameters are displayed in the parameter window in the settings for the speed controller. They can be saved by selecting **<Save selected parameters>** from the shortcut menu.

Tuning the position controller

The purpose of the position controller is to compare the position provided by the setpoint generator to the actual position and to generate a manipulated variable for the subordinate speed controller that works against a position change by changing the speed.

Selecting the "ncPOSITION" autotuning mode and restarting the tuning procedure from the command interface will determine the parameters for the position controller.

This requires that the underlying speed controller is stable.

These parameters are displayed in the parameter window in the settings for the position controller. They can be saved by selecting **<Save selected parameters>** from the shortcut menu.

Testing the controller settings

Before a movement is executed with the new controller parameters, the control loop should be checked for stability. For this purpose, the system has the option of applying a short disturbance signal to the control loop ("ncTEST" autotuning mode). If the controller parameters are set correctly, the disturbance will decay. The images shown are only a guideline. The key factor is that the current or the speed should exhibit decay.

[14] Various filters (e.g. "ncSPEED + ncISQ_F1_NOTCH") are available when tuning the speed controller to stabilize the system.

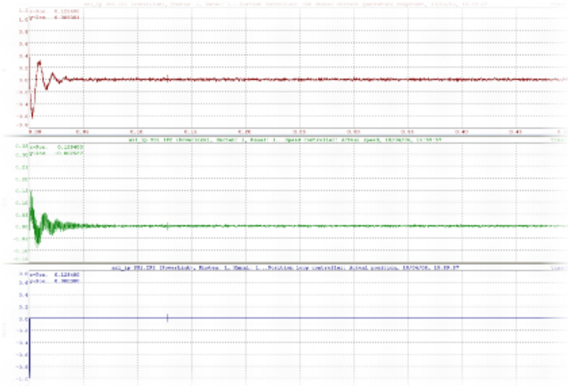


Figure 93: Example: Satisfactory controller parameters

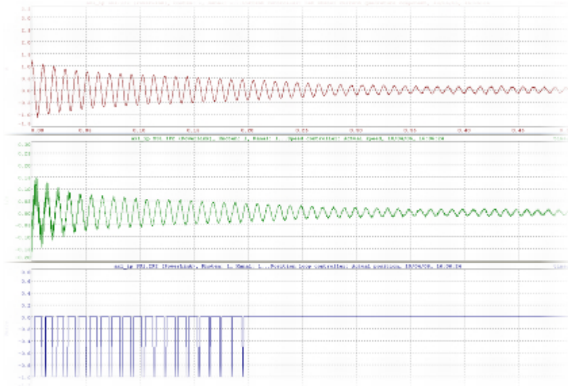


Figure 94: Example: Unsatisfactory controller parameters

It is a good idea to record several parameters to test the controller parameters. More information regarding how this can be configured is available in the Automation Studio help system.

Traced	Trigger	Description	ID
x		CTRL current: Stator current setpoint of the quadrature component	213
x		CTRL speed: Actual speed	251
x		CTRL position controller: Actual position	111
	x	Status: Controller	465

Table 6: Overview of parameters recorded in "ncTEST" tuning mode

Motion \ Reference manual \ ACP10 \

- NC objects \ NC object "ncAXIS" \ Setup (from 1.24 on) \ Setup for controller (auto-tuning) \ Mode "ncTEST" (controller test)
- Overview of ACOPOS parameter IDs

Feed-forward components

The purpose of the feed-forward component is to reduce the load on the controller when the speed changes. The values used by the feed-forward component take the system's moment of inertia into consideration and are determined during auto-tuning.

This requires that the underlying speed and position controllers are stable. To do so, the axis is put into motion and must be referenced.

Motion \ Commissioning

- Testing the holding brake
- Encoder phasing

Motion \ Commissioning \ Autotuning

- Preparing autotuning
- Speed controller
- Position controller
- Feed-forward components
- Testing controller settings

Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Setup (from V1.24 on) \ Setup for controller (autotuning)

- Function
- Data structure
- Mode "ncSPEED" (speed controller)
- Mode "ncPOSITION" (position controller)
- Mode "ncTEST" (controller test)
- Mode "ncFF" (feed forward)
- Example: Setup for controller in NC Test

Exercise: Determining control parameters using autotuning

Use the autotuning procedure to determine the controller parameters for an axis. To do so, proceed as follows:

1) Open the NC Test window
2) Check the holding brake and encoder signal.
3) Perform autotuning for the speed controller.
4) Perform autotuning for the position controller.
5) Test the controller parameters.

5.6 Commissioning checklist

This section provides step-by-step instructions for commissioning a motion control system. Pay special attention to the following:

Safety

It is particularly important to test the safety features. This includes the emergency stop and the limit switches installed on the machine.

Figure 95: Emergency stop

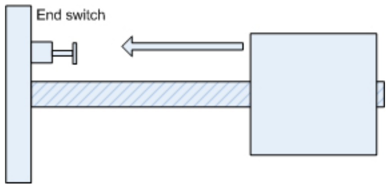


Figure 96: Limit switches

The ACOPOS user's manual and Automation Studio help system show how to correctly wire the E-stop and limit switch inputs.

Hardware \ Motion control \

- ACOPOS \ Safety technology
- ACOPOSmulti \ Safety technology
- ACOPOSmulti with SafeMC \ Safety technology

Digital inputs

It is important to check whether the servo drive's digital inputs have been wired according to the configured parameters. NC Watch can also be used to enter states for the inputs on the ACOPOS servo drive.

Figure 97: ACOPOS input

Checking the holding brake

The service interface in the command interface of the NC Test window can be used to verify that the holding brake is wired correctly.

Motion \ Commissioning \ Testing the holding brake

Units and movement parameters

The following configured units and settings must be checked:

- Encoder resolution and units per motor revolution
- Maximum lag error

- Software limits
- Maximum acceleration / speed values
- Stop functions
- Direction of motor rotation

Jolt time

Using jolt time is recommended to prevent placing an unnecessary load on the mechanical components. This is set up using the parameter t_jolt.

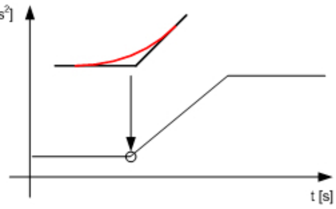


Figure 98: Jolt time

Homing

To ensure that positioning is accurate, a homing procedure must first be executed on the drive. There are a few different ways to do this.

> Programming \ Libraries \ Motion libraries \ ACP10_MC \ Function blocks \ Drive preparation \ MC_Home
>
> Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Homing procedure

Controller settings and autotuning

Controller parameters must be fine-tuned in order to adjust the control loop settings to the exact mechanical requirements. Automation Studio offers an integrated autotuning process for determining closed-loop control parameters. More information about these controller settings can be found in the "TM450 - ACOPOS: Control Concept and Configuration" training module.

> Motion \ Commissioning \ Autotuning
>
> Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Setup (V1.24 and higher) \ Setup for controller (autotuning)

Using third-party motors

A detailed description about commissioning motors can be found in "TM460 - Initial Commissioning of Motors".

> Motion \ Commissioning \ Encoder phasing
>
> Motion \ Commissioning \ Identifying motor parameters

Exercise: The effects of commissioning parameters

Change the following parameters and check their functionality.

Parameters:

- Jolt time (t_jolt)
- Lag error
- Direction of motor rotation (count_dir)
- Homing mode (homing)

1) Find and change the parameters in the parameter window.
2) Save and transfer the changed values.
3) Switch on the controller and perform a homing procedure.
4) Start the movement and trace.
5) Evaluate the trace data.
6) Perform autotuning.
7) Start the movement and trace.
8) Evaluate the trace data.
9) Work with the different homing modes.

   Explore the different homing modes using the help system as a guide. Conclude by performing a homing procedure to a limit switch.

6 SIMULATION OPTIONS

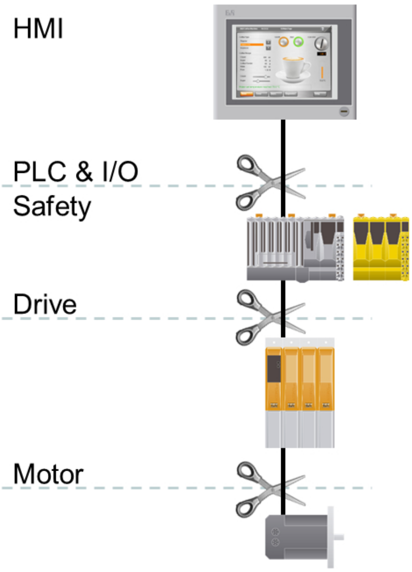


Figure 99: Complete simulation at every level

Automation Studio provides complete simulation for the controller, HMI, drive controller and motors.

In essence, all components of an integrated automation solution from B&R can be simulated. This makes it possible to simulate a motor, for example, if the actual motor shouldn't be running on the machine. Motion profiles can still be carried out on the controller or PC as well if the entire drive system is not yet available. The integrated WinIO interface makes it possible to fully simulate I/O points. The platform-independent Automation Runtime system allows control programs to be created and tested directly on the PC. This function is only available for the safety application. Controller applications can be executed in slow motion or time lapse in order to hone in on different periods in the machine's life cycle. Integrated VNC server functionality makes it possible to operate HMI applications not just remotely, but also directly on the PC.

6.1 Simulation of controller and drive

Simulation of a controller can be started by selecting the simulation icon in Automation Studio. All control programs run directly on the PC. This means that all of the software functions in the control application can be configured and tested independently of the hardware. When switching to simulation mode, the project is rebuilt, the simulation environment is automatically started and an online connection to the simulator is established.

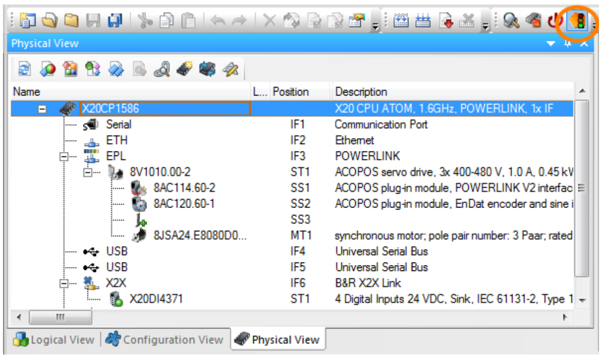


Figure 100: Enabling controller simulation from the Automation Studio toolbar

Active controller simulation is indicated in the Automation Studio status bar.

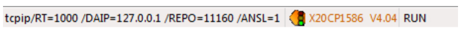


Figure 101: Active simulated indicated in the Automation Studio status bar

The drive is also automatically simulated when controller simulation is active.

Drive simulation

Drive simulation can also be activated separately in order to simulate the drive on the actual controller. The simulation mode for the drive can be selected in the NC mapping table.

Depending on the mode, either a complete simulation is performed or only setpoint generation without controller cascading. After changing the simulation mode, the project must be built and sent to the controller.

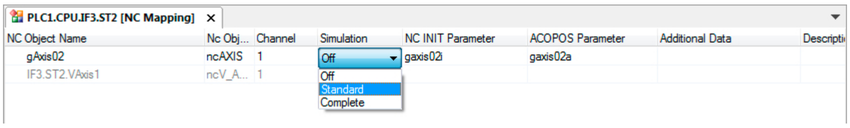


Figure 102: Enabling simulation in the NC mapping table

Project management \ Simulation \ CPU simulation Motion \ Reference manual \ ACOPOS drive functions \ ACOPOS simulation

6.2 Motor simulation in the NC Test window

Motor simulation

It is also possible to simulate a motor that is not connected to the drive or cannot be moved for whatever reason. Simulation mode is enabled in NC Test using the command interface under "Motor simulation". Simulation is started and stopped using the "**switch on**" and "**switch off**" commands. motor simulation can also be enabled from the ACOPOS parameter table.

Motion \ Reference manual \ ACOPOS drive functions \ ACOPOS simulation Motion \ Reference manual \ ACP10 \ NC objects \ NC object "ncAXIS" \ Simulation

7 SUMMARY

Drives are added in Automation Studio in the Physical View or System Designer. The Drive Configuration wizard guides you through the process of creating the motion control system. Configuration files for the drive configuration are stored in the Logical View. Diagnostic tools can be opened from the drive's shortcut menu in the Physical View and used to troubleshoot and put the drive into operation. NC Watch, NC Trace and the command interface can be used to view and control drive parameters and drive functions.

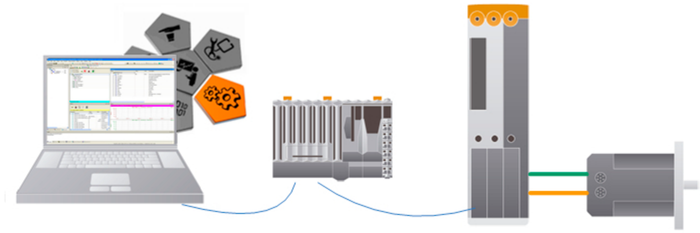


Figure 103: The integrated drive concept in Automation Studio

Network command trace can be used to monitor drive communication. The integrated autotuning function allows controller settings to be determined very quickly. System Diagnostics Manager is used to read basic information about the control and drive system, even without Automation Studio. The comprehensive Automation Studio help system provides assistance for drive configuration, troubleshooting and the installation of drive components.

The powerful simulation environment provided by Automation Studio allows the implementation and testing of drive applications right at the workstation, further helping to reduce the amount of time needed to get everything up and running.

TRAINING MODULES

TM210 – Working with Automation Studio
TM213 – Automation Runtime
TM223 – Automation Studio Diagnostics
TM230 – Structured Software Development
TM240 – Ladder Diagram (LD)
TM241 – Function Block Diagram (FBD)
TM242 – Sequential Function Chart (SFC)
TM246 – Structured Text (ST)
TM250 – Memory Management and Data Storage
TM400 – Introduction to Motion Control
TM410 – Working with Integrated Motion Control
TM440 – Motion Control: Basic Functions
TM441 – Motion Control: Multi-axis Functions
TM450 – ACOPOS Control Concept and Adjustment
TM460 – Initial Commissioning of Motors
TM500 – Introduction to Integrated Safety
TM510 – Working with SafeDESIGNER
TM540 – Integrated Safe Motion Control
TM600 – Introduction to Visualization
TM610 – Working with Integrated Visualization
TM630 – Visualization Programming Guide
TM640 – Alarms, Trends and Diagnostics
TM670 – Advanced Visual Components
TM800 – APROL System Concept
TM811 – APROL Runtime System
TM812 – APROL Operator Management
TM813 – APROL XML Queries and Audit Trail
TM830 – APROL Project Engineering
TM890 – The Basics of LINUX
TM920 – Diagnostics and service
TM923 – Diagnostics and Service with Automation Studio
TM950 – POWERLINK Configuration and Diagnostics

TM1010 – B&R CNC System (ARNC0)
TM1110 – Integrated Motion Control (Axis Groups)
TM1111 – Integrated Motion Control (Path Controlled Movements)
TM261 – Closed Loop Control with LOOPCONR
TM280 – Condition Monitoring for Vibration Measurement
TM480 – The Basics of Hydraulics
TM481 – Valve-based Hydraulic Drives
TM482 – Hydraulic Servo Pump Drives

TM410TRE.40-ENG