

Rexroth IndraMotion MLC 12VRS Functional Description

R911333848
Edition 04

Application Manual



Title	Rexroth IndraMotion MLC 12VRS Functional Description
Type of Documentation	Application Manual
Document Typecode	DOK-MLC***-FUNC****V12-AP04-EN-P
Internal File Reference	RS-df7e369c8fcfc63b0a6846a5017f988b-1-en-US-2

Purpose of Documentation	The documentation describes the • Wizards • Context menus • Dialogs and frequently required procedures • Device configuration • Function modules • Functionalities of the IndraMotion MLC 12VRS.

Record of Revision	Edition	Release Date	Notes
	Edition 01	04.2011	First edition for 12VRS
	Edition 02	08.2011	Supplements, corrections
	Edition 03	02.2012	Supplements, corrections
	Edition 04	10.2012	Open Loop, supplements, corrections

Copyright © Bosch Rexroth AG 2012

This document, as well as the data, specifications and other information set forth in it, are the exclusive property of Bosch Rexroth AG. It may not be reproduced or given to third parties without its consent.

Liability The specified data is intended for product description purposes only and shall not be deemed to be a guaranteed characteristic unless expressly stipulated in the contract. All rights are reserved with respect to the content of this documentation and the availability of the product.

Published by Bosch Rexroth AG
Electric Drives and Controls
P.O. Box 13 57
97803 Lohr, Germany
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr, Germany
Phone +49 9352 18 0
Fax +49 9352 18 8400

<http://www.boschrexroth.com/electrics>

System Development Automation Motion Logic Control, HaBu (KaWa/MePe)

Note This document has been printed on chlorine-free bleached paper.

Table of Contents

	Page
1 About this Documentation.....	13
1.1 Validity of the Documentation.....	13
1.2 Documentation Structure.....	13
1.3 Required and Supplementing Documentations.....	14
1.4 Information Representation.....	18
1.4.1 Safety Instructions.....	18
1.4.2 Symbols Used.....	18
1.4.3 Names and Abbreviations.....	19
2 Important Instructions on Use.....	21
2.1 Intended Use	21
2.1.1 Introduction.....	21
2.1.2 Areas of Use and Application.....	21
2.2 Unintended Use.....	22
3 Safety Instructions for Electric Drives and Controls.....	23
3.1 Definitions of Terms.....	23
3.2 General Information.....	24
3.2.1 Using the Safety Instructions and Passing Them on to Others.....	24
3.2.2 Requirements for Safe Use.....	24
3.2.3 Hazards by Improper Use.....	25
3.3 Instructions with Regard to Specific Dangers.....	26
3.3.1 Protection Against Contact With Electrical Parts and Housings.....	26
3.3.2 Protective Extra-Low Voltage as Protection Against Electric Shock	27
3.3.3 Protection Against Dangerous Movements.....	28
3.3.4 Protection Against Magnetic and Electromagnetic Fields During Operation and Mounting.....	29
3.3.5 Protection Against Contact With Hot Parts.....	30
3.3.6 Protection During Handling and Mounting.....	30
3.3.7 Battery Safety.....	30
3.3.8 Protection Against Pressurized Systems.....	31
3.4 Explanation of Signal Words and the Safety Alert Symbol.....	31
4 System Overview.....	33
4.1 Hardware.....	33
4.2 Firmware.....	36
4.3 Software.....	37
5 Commissioning and Operation.....	39
5.1 General Information.....	39
5.2 IndraMotion MLC – Basic Sequences, General Information.....	39
5.3 IndraMotion MLC L25/L45/L65 – Commissioning.....	39
5.3.1 Connecting the Control.....	39
5.3.2 Operating Control Display.....	41

Table of Contents

	Page
5.3.3 Control Memory Deletion.....	43
5.3.4 Setting the IP Address.....	44
5.3.5 Enabling the sercos Bus.....	47
5.3.6 Replacing Hardware.....	47
5.4 Installing and Starting IndraWorks.....	48
5.5 Overview on the IndraWorks Interface.....	50
5.6 Creating a Project.....	51
5.7 Creating a Control.....	53
5.8 Creating a Function Module.....	60
5.9 I/Os and Field Buses.....	62
5.9.1 Creating Onboard I/Os.....	62
5.9.2 Creating Inline I/Os.....	64
5.9.3 Creating Profibus I/Os.....	71
5.9.4 Creating a Profinet IO Controller.....	71
5.9.5 Creating sercos III I/Os.....	72
5.10 IP Communication	76
5.10.1 Overview.....	76
5.10.2 Basics.....	77
5.10.3 Default Setting.....	78
5.10.4 IndraWorks Commissioning Dialogs.....	82
5.10.5 Automatic IP Address Generation.....	84
5.10.6 Configuring the Bridge.....	85
5.10.7 Involved IP Communication Parameters.....	85
5.10.8 Ethernet Ports and Firewall.....	86
5.10.9 More Notes on the Use of Ethernet-Based Interfaces.....	87
5.11 Axes.....	87
5.12 Creating a Kinematics.....	88
6 Control.....	93
6.1 Project Structure.....	93
6.2 Toolbar of the Control.....	94
6.3 Control - Nodes.....	95
6.3.1 Node Overview.....	95
6.3.2 Logic.....	96
"Logic" Subfolder.....	96
"Logic" Context Menu.....	97
Application – Toolbar.....	98
"Application" Context Menu.....	99
6.3.3 Motion.....	100
General Information.....	100
"Motion" Context Menu.....	101
6.3.4 Technology.....	104
Introduction.....	104
Context Menu.....	105
"Options" Menu.....	106
Dialogs.....	107

Table of Contents

	Page
6.3.5 Robot.....	115
General Information.....	115
Robot - Context Menu.....	115
6.3.6 Onboard I/Os.....	116
6.3.7 Inline I/Os.....	116
6.3.8 Profibus/M.....	116
6.3.9 Profinet IO-Controller.....	116
6.3.10 Function Modules.....	117
6.3.11 sercos.....	117
General Information.....	117
sercos - Context Menu.....	118
sercos - Toolbar.....	119
6.4 Control – Context Menu.....	119
6.4.1 Overview on Menu Items.....	119
6.4.2 Device Editor.....	122
Device Editor - General Information.....	122
Applications	122
Log	123
PLC Settings.....	124
Information	125
6.4.3 Communication.....	125
TCP Settings.....	125
Interfaces.....	126
6.4.4 Language Settings.....	126
6.4.5 Cam Explorer.....	127
6.4.6 Parameters.....	133
Parameter Editor.....	133
Parameter Group.....	135
Saving Parameters in Drives.....	136
Parameters - Export.....	136
Parameters - Import.....	139
6.4.7 Diagnostics.....	141
Device Status.....	141
Advanced Properties.....	143
Clearing Errors.....	144
Error/Diagnostic Memory.....	144
Invalid Parameters.....	147
Task List/Configuration.....	149
Task Execution Viewer.....	150
6.4.8 Multi-Device.....	150
Introduction.....	150
Brief Description of the Multi-Device Table.....	151
Multi-device Functionality in Detail.....	152
Configuration File to Deactivate the Profibus DP Slaves.....	158
6.4.9 Firmware Management.....	161
6.4.10 Device Data.....	163

Table of Contents

	Page
Archiving.....	163
Restoring.....	167
Explorer.....	170
6.4.11 Simulation.....	172
6.4.12 Exporting.....	172
6.4.13 Importing.....	174
6.4.14 Comparing/Merging.....	176
General Information.....	176
Control.....	177
Axes.....	177
Robot.....	177
Function Modules.....	178
6.4.15 Runtime Licenses.....	178
6.4.16 Print Preview.....	178
6.4.17 Printing.....	180
6.5 Version Control of the Control.....	182
6.5.1 Introduction.....	182
6.5.2 Special Features.....	182
Project Management.....	182
Working with Controls.....	183
6.6 Complete Data Backup.....	183
6.6.1 General Information.....	183
6.6.2 Complete Backup.....	184
6.6.3 Restoring a Project.....	189
7 PLC Programming.....	193
7.1 General Information.....	193
7.2 Creating PLC Program.....	193
7.2.1 Opening "Plc_Main Object".....	193
7.2.2 Declaring Variables.....	193
7.2.3 Editing Instruction.....	194
7.2.4 Compiling Program.....	195
7.2.5 Loading the PLC Program to the Control.....	195
7.3 Library Management.....	195
7.3.1 General Information.....	195
7.3.2 Library Overview.....	196
7.3.3 Installing on the System and Integrating into a Project	197
7.3.4 Referenced Libraries.....	198
7.3.5 Library Versions.....	199
7.3.6 Unique Access to Library Blocks (Namespace).....	199
7.3.7 Creating Libraries.....	200
7.3.8 Converting IndraLogic 1.x Libraries.....	201
7.3.9 Integrating External Code.....	201
7.4 Task System	201
7.4.1 General Information.....	201
7.4.2 Task Configuration.....	204

Table of Contents

	Page
General Information.....	204
Configuration Examples.....	205
7.4.3 Task Editor.....	207
7.4.4 "Properties Dialog"	208
7.4.5 "Configuration" dialog.....	209
7.4.6 Task List/Configuration.....	210
7.4.7 Task Execution Viewer.....	212
7.4.8 I/O Turn-Around Times.....	213
General Information.....	213
Profibus, Profinet and Inline I/Os.....	214
sercos III I/Os.....	215
Onboard and Fast I/Os.....	216
7.4.9 System Tasks.....	217
7.5 I/O Accesses	222
7.6 Memory Organization and Pointer Programming	223
7.6.1 Memory Alignment.....	223
7.6.2 Byte Order.....	224
7.6.3 Pointer Addressing.....	225
8 Axes.....	227
8.1 Introduction and Overview.....	227
8.2 Real Axis.....	231
8.2.1 General Information.....	231
8.2.2 Error Reaction of the Axis.....	234
8.2.3 Creating a Real Axis.....	234
8.2.4 Real Axis – Dialogs.....	238
Overview on Dialogs.....	238
8.2.5 Real Axis – Context Menu.....	241
Overview on Menu Items.....	241
8.3 Virtual axis.....	242
8.3.1 General Information.....	242
8.3.2 Creating a Virtual Axis (Virtual Master Axis).....	243
8.3.3 Virtual Axis - Dialogs.....	246
Overview on Dialogs.....	246
8.3.4 Virtual axis - Context menu.....	246
Overview on Menu Items.....	246
8.4 Controller axis.....	248
8.4.1 General Information.....	248
8.4.2 Creating a Controller Axis.....	248
Creating an Axis in the Project Explorer.....	248
Step 1: General Settings.....	249
Step 2: Configuration.....	250
Central Controller Step 3: Hardware Link.....	250
Central Controller Step 4: Signal Assignment.....	251
Last Step: Summary.....	252
8.4.3 Controller Axis - Dialogs.....	253

Table of Contents

	Page
Controller Axis - Overview on Dialogs.....	253
8.4.4 Controller Axis - Context Menu.....	254
Overview on Menu Items.....	254
Initial Commissioning.....	255
Diagnostics.....	255
8.5 Link axis.....	257
8.5.1 General Information.....	257
8.5.2 Creating a Link Axis.....	257
Settings on the Consumer Side.....	257
Settings on the Producer Side.....	260
8.5.3 Link Axis - Dialogs.....	260
Overview on Dialogs.....	260
8.5.4 Like Axis – Context Menu.....	260
Overview on Menu Items.....	260
8.6 Encoder axis.....	262
8.6.1 General Information.....	262
8.6.2 Creating an Encoder Axis (Real Master Axis).....	262
Creating an Axis in the Project Explorer.....	262
Configuration of the Measuring Encoder.....	265
8.6.3 Encoder Axis - Dialogs.....	266
Overview on Dialogs.....	266
8.6.4 Encoder axis - Context menu.....	267
Overview on Menu Items.....	267
8.7 Dialogs.....	268
8.7.1 Overview on Dialogs.....	268
8.7.2 Motor.....	272
8.7.3 Motor Temperature Monitoring.....	273
8.7.4 Brake.....	273
8.7.5 Measuring Systems.....	274
Settings, Motor Encoder.....	274
Settings of Optional Encoder.....	275
8.7.6 Settings: Scaling/Units.....	277
8.7.7 Scaling/Units Extended.....	278
8.7.8 Mechanical Gear.....	280
8.7.9 Closed-Loop Control.....	281
8.7.10 Settings of Status Messages.....	281
8.7.11 Drive Halt.....	283
8.7.12 Establishing Dimensional Reference.....	283
Motor Encoder.....	283
Optional Encoder.....	284
8.7.13 Error Reaction.....	285
Drive.....	285
Power Section.....	287
8.7.14 E-Stop Function.....	288
8.7.15 Motion Limit Values.....	289
8.7.16 Initial Values.....	291

Table of Contents

	Page
8.7.17 Compensation Functions/Corrections.....	293
Encoder Correction.....	293
Backlash on Reversal Correction.....	294
8.7.18 Drive-integrated Command Value Generator.....	294
8.7.19 Cam Table.....	295
General Information.....	295
Master Axis Position Offset.....	296
Gear Settings.....	297
Cam Table Phase Offset.....	298
Dynamic Phase Offset.....	298
Cam Table Selection.....	299
Dynamic Synchronization - Phase Synchronization.....	299
8.7.20 MotionProfile.....	300
General Information.....	300
Master Axis Position Offset.....	302
Gear Settings.....	303
Cam Table - Phase Offset.....	303
Dynamic Phase Offset.....	304
MotionProfile Overview.....	304
Dynamic Synchronization - Phase Synchronization.....	306
8.7.21 FlexProfile.....	307
General Information.....	307
Master Axis Position Offset.....	309
Gear Settings.....	309
Cam Table Phase Offset.....	310
FlexProfile Overview.....	311
Dynamic Synchronization - Phase Synchronization.....	313
8.7.22 Gear with Velocity Synchronization.....	314
General Information.....	314
Master Axis Position Offset.....	315
Gear Settings.....	316
Dynamic Synchronization.....	317
8.7.23 Gear with Phase Synchronization.....	317
General Information.....	317
Master Axis Position Offset.....	319
Gear Settings.....	319
Dynamic Synchronization.....	320
8.7.24 Axis Configuration.....	321
8.7.25 Drive-integrated Safety Engineering.....	323
8.7.26 Position Switching Point.....	323
8.7.27 Touch Probe.....	324
8.7.28 Jogging.....	327
8.7.29 Initial Commissioning.....	328
8.7.30 Exchanging the Drive/Firmware	329
8.7.31 Parameters.....	334
Loading Basic Parameters.....	334

Table of Contents

	Page
Drive Password - Change Password.....	335
Save Mode.....	335
Saving Parameters in the Drive.....	336
8.7.32 Communication.....	336
Cyclic sercos Data Channel.....	336
Signal Status Word.....	337
Signal Control Word.....	339
8.7.33 Diagnostics.....	341
Status.....	341
Clearing Errors.....	342
Error/Diagnostic Memory.....	342
Average Value Filter Display.....	343
9 Motion Functionality.....	345
9.1 General Information.....	345
9.2 Motion Mode.....	345
9.3 Parameters.....	345
9.4 Modes for Project Editing.....	346
9.4.1 Introduction.....	346
Offline Mode.....	346
Online Mode.....	346
Offline Parameterization.....	347
9.4.2 Transitions Between the Modes.....	347
Mode Switching.....	347
Adjustment of the Command/Actual Configuration.....	348
Scanning the Ring.....	352
Axis Adjustment Dialog.....	355
9.4.3 Loading Offline Parameters to the Device.....	356
9.5 Modes to Operate a Real Axis	359
9.5.1 Deactivated Axis.....	359
9.5.2 Parking Axis.....	359
9.5.3 Notes on How to Use Deactivated and Parking Axes.....	360
9.5.4 Command Value Decoupling.....	360
Mode of Operation.....	360
Displaying the Command Value Decoupling.....	361
Axis State Dialogs for Real Axes.....	363
9.6 Single-Axis Modes.....	365
9.6.1 Overview.....	365
9.6.2 Operation Modes.....	366
9.7 Operation Modes for Synchronous Motions with Electronic Gear Function.....	372
9.8 Validating Motion Configuration.....	379
9.8.1 Overview.....	379
9.8.2 Error Messages.....	380
General Information.....	380
Axis Error Messages.....	381
Control Error Messages.....	383

Table of Contents

	Page
Error Messages of the Function Modules.....	384
Error Messages of the Kinematics.....	385
Error Messages of Cross Communication.....	390
Error Messages of sercos Devices.....	390
9.8.3 Information Messages.....	391
General Information.....	391
Non-Object-Related Information Messages.....	391
9.8.4 Warning Messages.....	391
General Information.....	391
Axis Warning Messages.....	391
Warning Messages of the Control.....	393
Warning Messages of Function Modules.....	393
Warning Messages of the Kinematics.....	393
Warning Messages of the Cross Communication.....	393
Warning Messages of sercos Devices.....	395
9.9 Electronic Cams.....	396
9.9.1 General Information.....	396
9.9.2 Cam Solutions.....	396
9.9.3 Electronic Cam.....	398
9.9.4 Fields of Application.....	398
9.9.5 Centralized and Decentralized Cam Concept.....	399
9.9.6 FlexProfile – Basic Functionalities	399
General Information.....	399
Basic Structure of Synchronous Operating States"FlexProfile".....	399
Profile Configuration.....	400
Time Master	401
Motion Laws.....	401
Switching and Synchronizing the FlexProfile	407
Event Control	414
Assigning MotionProfile and Axis.....	417
FlexProfile – Configuration.....	418
9.10 Touch Probe.....	424
9.10.1 Introduction.....	424
9.10.2 Features.....	424
9.10.3 Functional Description.....	425
9.10.4 Accuracy.....	427
9.10.5 Data Acquisition of the Touch Probe.....	427
Activation of Measured Value Acquisition.....	427
Measuring Signal Selection.....	428
Measured value acquisition mode.....	428
Saving Measured Values.....	428
Dead time compensation.....	428
Restarting or Disabling the Measured Value Acquisition.....	430
Measured Value Status.....	430
9.10.6 Touch Probe Expectation Window.....	430
Use of the Expectation Window.....	430

Table of Contents

	Page
Expectation Window Setting.....	431
Marker Detection.....	431
9.10.7 Overview on M-Parameters.....	431
M-0-0000, Dummy M-parameter.....	431
M-0-0001, Logical probe number.....	431
M-0-0002, Touch probe name.....	432
M-0-0005, Input selection.....	432
M-0-0006, Signal selection.....	432
M-0-0007, Control word.....	433
M-0-0009, Osci: list of all M-parameters that can be oscilloscoped.....	434
M-0-0010, List of all M parameters.....	434
M-0-0016, Project identification number.....	435
M-0-0021, Status word.....	435
M-0-0024, Command: Activation.....	436
M-0-0130, Measured value, positive edge.....	436
M-0-0131, Measured value, negative edge.....	436
M-0-0132, Difference value.....	437
M-0-0140, Time stamp, positive edge.....	437
M-0-0141, Time stamp, negative edge.....	437
M-0-0200, Dead-time compensation, positive edge.....	438
M-0-0201, Dead-time compensation, negative edge.....	438
M-0-0202, Starting position of the expectation window, positive edge.....	438
M-0-0203, End position of the expectation window, positive edge.....	438
M-0-0204, Starting position of the expectation window, negative edge.....	439
M-0-0205, End position of the expectation window, negative edge.....	439
M-0-0206, Maximum number of marker failures.....	439
M-0-0224, Number of marker failures, positive edge.....	439
M-0-0225, Number of marker failures, negative edge.....	440
M-0-0401, Input status.....	440
M-0-0405, Enabling.....	440
M-0-0409, Counter of measured values, positive edge.....	441
M-0-0410, Counter of measured values, negative edge.....	441
M-0-0411, Counter of difference values.....	441
9.10.8 Function Blocks for Touch Probes.....	442
9.11 Programmable Limit Switch.....	442
10 Kinematics.....	445
10.1 General Information.....	445
10.2 Menu Structure of the Kinematics.....	446
10.2.1 Menu Item: Robot.....	446
10.2.2 Menu item: Kinematics.....	446
10.2.3 Menu Item: Programs.....	447
10.2.4 Menu item of a Program.....	447
10.3 Kinematics - Context menu.....	448
10.3.1 Overview.....	448
10.3.2 Setup.....	448

Table of Contents

	Page
10.3.3 Diagnostics.....	449
10.4 Kinematics – Dialogs.....	450
10.4.1 Overview on dialogs.....	450
10.4.2 Axis Lengths and Coupling.....	450
10.4.3 Motion Limit Values.....	452
10.4.4 Belt Coupling Factors.....	452
10.4.5 Coordinate Offset.....	453
10.4.6 Jogging.....	454
10.4.7 Safe Zones.....	456
10.4.8 Workpiece Coordinate Assignment.....	458
10.5 RCL Editor.....	459
10.5.1 General Information.....	459
10.5.2 Characteristics.....	459
10.5.3 Creating a New RCL Program.....	459
10.5.4 Working with the Editor.....	460
10.6 Compiler.....	463
10.7 Debugger.....	465
10.7.1 General Information.....	465
10.7.2 Setting and Deleting Breakpoints.....	465
10.7.3 Debugging a Program.....	466
10.7.4 Monitoring Variables.....	467
10.8 Settable Options for RoCo.....	467
10.8.1 General Information.....	467
10.8.2 RoCo Compiler.....	467
10.8.3 RoCo Online Settings.....	469
10.8.4 RoCo-Editor.....	469
Text Editor.....	469
Text Area.....	471
Page Margins.....	474
Keywords.....	476
10.9 Creating a RoCo Project.....	477
11 CamBuilder Connection.....	491
11.1 CamBuilder Functionality.....	491
12 Oscilloscope Connection.....	493
12.1 Oscilloscope Functionality.....	493
13 User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts... 495	495
13.1 General Information.....	495
13.2 File Management.....	495
13.3 File Structure.....	495
13.4 Function.....	497
13.5 Node Names for Units.....	498

Table of Contents

	Page
13.6 Node Names for Menu Texts (Display Texts).....	499
14 Remote Access via Modem.....	501
14.1 General Information.....	501
14.1.1 Definition of Terms and Introduction.....	501
14.1.2 Connection via Internet.....	501
14.1.3 Point-to-Point Connection.....	502
14.1.4 Accessibility to the Control.....	502
14.2 Dial-Up Connection Between Service PC and Client PC.....	502
14.2.1 General Information.....	502
14.2.2 Required Equipment.....	503
14.2.3 Installation and Configuration.....	503
14.2.4 Possible Remote Maintenance Tasks.....	503
14.3 Dial-Up Connection Between Service PC and MoRoS.....	503
14.3.1 General Information.....	503
14.3.2 Required Equipment.....	504
14.3.3 Remote Maintenance Actions.....	504
14.3.4 Installation and Configuration.....	504
General Information.....	504
Initial MoRoS Configuration.....	504
Network Configuration.....	506
MoRoS Network Configuration.....	506
Network Configuration of the Controls.....	508
Configuration Service PC.....	508
15 Service and Support.....	513
Index.....	515

1 About this Documentation

1.1 Validity of the Documentation

The documentation describes the

- Wizards
- Context menus
- Dialogs
- Control commissioning
- Device configuration
- Functionalities

of the IndraMotion MLC 12VRS.

Target group

This documentation is intended for the commissioner of a system.

Application phases

This documentation supports the user during all phases

- Engineering
- Commissioning
- Operation and programming
- Maintenance
- Troubleshooting

1.2 Documentation Structure

The first part of the document provides important instructions on use and safety ([chapter 2 "Important Instructions on Use" on page 21](#) and [chapter 3 "Safety Instructions for Electric Drives and Controls" on page 23](#)).

The [chapter 4 "System Overview" on page 33](#) contains a system overview on the IndraMotion MLC.

The overview is structured as follows:

- Hardware
- Firmware
- Software

The [chapter 5 "Commissioning and Operation" on page 39](#) and [chapter 6 "Control" on page 93](#) contain descriptions on the commissioning and operation as well as on the IndraWorks user interface.

The [chapter 7 "PLC Programming" on page 193](#) provides an overview on the PLC programming.

The [chapter 8 "Axes" on page 227](#) and [chapter 9 "Motion Functionality" on page 345](#) describe the functionalities of the

- real axis
- virtual axis
- Encoder axis
- Link axis
- Controller axis

as well as the Motion functionality.

The [chapter 10 "Kinematics" on page 445](#) provides an overview on the Robot functionality.

About this Documentation

The [chapter 11 "CamBuilder Connection" on page 491](#) and [chapter 12 "Oscilloscope Connection" on page 493](#) contain document specifications describing the respective function.

The [chapter 13 "User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts" on page 495](#) describes the use of user-defined languages.

The [chapter 14 "Remote Access via Modem" on page 501](#) describes the remote access to an IndraMotion MLC.

The [chapter 15 "Service and Support" on page 513](#) provides information on the customer service helpdesk.

1.3 Required and Supplementing Documentations

Documentation titles with type codes and parts numbers

IndraWorks		MLC	XLC
/36/	Rexroth IndraWorks 12VRS Software Installation DOK-IWORKS-SOFTINS*V12-COxx-EN-P, R911334396 This documentation describes the IndraWorks installation.	X	X
/5/	Rexroth IndraWorks 12VRS Engineering DOK-IWORKS-ENGINEE*V12-APxx-EN-P, R911334388 This documentation describes the application of IndraWorks in which the Rexroth Engineering tools are integrated. It includes instructions on how to work with IndraWorks and how to operate the oscilloscope function.	X	X
/20/	Rexroth IndraMotion MLC 12VRS Functional Description DOK-MLC***-FUNC****V12-APxx-EN-P, R911333848 This documentation describes wizards, context menus, dialogs, control commissioning, device configuration and functionalities of the IndraMotion MLC.	X	
/20/	Rexroth IndraLogic XLC 12VRS Functional Description DOK-XLC***-FUNC****V12-APxx-EN-P, R911333878 This documentation describes wizards, context menus, dialogs, control commissioning, device configuration and functionalities of the IndraLogic XLC.		X
/7/	Rexroth IndraWorks 12VRS CamBuilder DOK-IWORKS-CAMBUIL*V12-APxx-EN-P, R911333842 This documentation describes the basic principles and operation of the CamBuilder, the cam editing tool.	X	X
/37/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS Automation Interface DOK-XLCMLC-AUT*INT*V12-APxx-EN-P, R911334178 This documentation describes the script-based access to IndraWorks project data via the interface of the Automation Interface.	X	X
/38/	Rexroth IndraWorks 12VRS FDT Container DOK-IWORKS-FDT*CON*V12-APxx-EN-P, R911334398 This documentation describes the IndraWorks FDT Container functionality. It includes the activation of the functionality in the project and working with DTMs.	X	X

About this Documentation

/29/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS Project Conversion DOK-XLCMLC-PROCONV*V12-APxx-EN-P, R911334187 This documentation describes the project conversion of IndraLogic 04VRS and IndraMotion MLC04VRS on IndraWorks version 12 with IndraLogic 2G. Changes with regard to Motion and PLC are described in detail.	X	X
/28/	Rexroth IndraMotion MLC 12VRS Commissioning DOK-MLC***-STARTUP*V12-COxx-EN-P, R911333858 This documentation describes the steps required to commission and perform service on the IndraMotion MLC system. It includes checklists for tasks to be frequently performed and a detailed description of the steps.	X	

Motion		MLC	XLC
/23/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS PLCopen Libraries DOK-XLCMLC-FUNLIB**V12-LIxx-EN-P, R911334182 This documentation describes the function blocks, functions and data types of the RIL_CommonTypes, ML_Base and ML_PLCOpen libraries for the IndraLogic XLC/IndraMotion MLC. It also includes the error reactions of function blocks.	X	X
/27/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS Generic Application Template DOK-XLCMLC-TF*GAT**V12-APxx-EN-P, R911334191 This documentation provides a structured template to the IndraLogic PLC programmer. This template can be used to add and edit the PLC programming code. It includes the template, the template wizard and example applications.	X	X
/31/	Rexroth IndraMotion MLC 12VRS RCL Programming Instruction DOK-MLC***-RCL*PRO*V12-APxx-EN-P, R911333852 This documentation provides information on the RobotControl. The programming language RCL (RobotControl Language) is focused. The program structure, variables, functions, motion statements and the required system parameters are described.	X	
/21/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS Parameters DOK-XLCMLC-PARAM***V12-RExx-EN-P, R911334176 This documentation describes the parameters of the XLC/MLC systems as well as the interaction between parameterization and programming. It includes the axis parameters, control parameters, kinematic parameters, touch probe parameters and programmable limit switch parameters.	X	X
/10/	Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07 DOK-INDRV*-MP*-07VRS**-FKxx-EN-P, R911328670		
/11/	Rexroth IndraDrive MPx-16 Functions DOK-INDRV*-MP*-16VRS**-APxx-EN-P, R911326767		

Field buses		MLC	XLC
/39/	Rexroth IndraMotion MLC 11VRS PLCopen Field Bus DOK-IM*ML*-PLCFBUS*V11-APxx-EN-P, R911333896 This documentation describes the creation of field bus drives in an IndraWorks project, function blocks, functions and data types of the RIL_CommonTypes.library (excerpt for field bus drives), the RMB_PLCOpenFieldBus.library and the RIL_Utility.library (excerpt for field bus drives). It also includes the error reactions of function blocks.	X	X
/4/	Rexroth IndraWorks 12VRS Field Buses DOK-IWORKS-FB*****V12-APxx-EN-P, R911334394 This documentation describes the supported field buses and their diagnostic function blocks.	X	X

About this Documentation

HMI		MLC	XLC
/8/	Rexroth IndraWorks 12VRS HMI DOK-IWORKS-HMI****V12-APxx-EN-P, R911334392 This documentation describes the functions, configuration and operation of the user interfaces IndraWorks HMI Engineering and IndraWorks HMI Operation.	X	X
/6/	Rexroth IndraWorks 12VRS WinStudio DOK-IWORKS-WINSTUD*V12-APxx-EN-P, R911333844 This documentation describes the installation of the software, working with WinStudio and the creation and operation of applications.	X	X
/50/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS HMI Connection DOK-XLCMLC-HMI****V12-APxx-EN-P, R911334184 This documentation describes the visualization systems supported by the IndraLogic XLC and IndraMotion MLC and their connection.	X	X

PLC		MLC	XLC
/3/	Rexroth IndraWorks 12VRS IndraLogic 2G Programming Instruction DOK-IWORKS-IL2GPRO*V12-APxx-EN-P, R911334390 This documentation describes the PLC programming tool IndraLogic 2G and its use. It includes the basic use, first steps, visualization, menu items and editors.	X	X
/33/	Rexroth IndraWorks 12VRS, Basic Libraries, IndraLogic 2G DOK-IL*2G*-BASLIB**V12-LIxx-EN-P, R911333835 This documentation describes the system-comprehensive PLC libraries.	X	X

Technology			
/30/	Rexroth IndraMotion MLC 12VRS Technology Libraries DOK-MLC***-TF*LIB**V12-LIxx-EN-P, R911333868 This documentation describes the function blocks, functions and data types of the "ML_TechInterface.library", "ML_TechMotion.library", "RMB_TechCam.library" and "ML_TechBase.library". It also includes libraries for the winder functionality, register controller functionality and CrossCutter functionality.	X	
/60/	Rexroth IndraMotion MLC 12VRS RegisterControl (Library) DOK-MLC***-REGI*CO*V12-LIxx-EN-P, R911333856 This documentation describes the inputs and outputs of the individual function blocks and provides notes on their use.	X	
/62/	Rexroth IndraMotion MLC 12VRS RegisterControl (Application Manual) DOK-MLC***-REGI*CO*V12-APxx-EN-P, R911333854 This documentation describes the application of the integrated register control for a rotogravure printing machine. The components of the mark stream sensor, the HMI application and the error recovery options are described. This instruction provides information on how to operate the register control, react to errors and query diagnostics. This documentation is written for machine setters and machine operators.	X	
/49/	Rexroth IndraMotion MLC 12VRS Winder Function Application DOK-MLC***-TF*WIND*V12-APxx-EN-P, R911333870 This application-related system documentation describes the application of the winder technology functions.	X	

About this Documentation

Hardware		MLC	XLC
/1/	Rexroth IndraControl L45/L65 DOK-CONTRL-IC*L45*L65*-PRxx-EN-P, R911324661	X	X
/2/	Rexroth IndraControl L25 DOK-CONTRL-IC*L25*****-PRxx-EN-P, R911328474	X	X
/24/	Rexroth IndraControl Lxx 12VRS Function Modules DOK-CONTRL-FM*LXX**V12-APxx-EN-P, R911333830 This documentation describes all function modules of the Lxx controls including engineering and diagnostics.	X	X
/12/	Rexroth IndraDrive Drive Controllers MPx-02 to MPx-07 DOK-INDRV*-GEN-**VRS**-PAxx-EN-P, R911297317		
/13/	Rexroth IndraDrive MPx-02 to MPx-07 and HMV DOK-INDRV*-GEN-**VRS**-WAxx-EN-P, R911297319		
/35/	Rexroth IndraDrive Drive Controller Control Sections CSB01, CSH01, CDB01 DOK-INDRV*-CSH*****-PR08-EN-P, R911295012		

Diagnostics and service		MLC	XLC
/26/	Rexroth IndraMotion MLC/XLC 11VRS Service Tool DOK-IM*ML*-IMST***V11-RExx-EN-P, R911331940	X	X
/22/	Rexroth IndraLogic XLC IndraMotion MLC 12VRS Diagnostics DOK-XLCMLC-DIAG***V12-RExx-EN-P, R911334180 This documentation includes all control parameters implemented in the control systems IndraLogic XLC and IndraMotion MLC.	X	X

System Overview		MLC	XLC
/48/	Rexroth IndraMotion for Printing 12VRS System Overview DOK-IM*PR*-SYSTEM**V11-PRxx-EN-P, R911333840 This documentation describes the product IndraMotion for Packaging. It introduces the control systems, drive systems and I/O systems as well as the commissioning and programming.	X	
/48/	Rexroth IndraMotion for Packaging 12VRS System Overview DOK-IM*PA*-SYSTEM**V12-PRxx-EN-P, R911333838 This documentation describes the product IndraMotion for Packaging. It introduces the control systems, drive systems and I/O systems as well as the commissioning and programming.	X	
/9/	Rexroth IndraMotion MLC 12VRS System Overview DOK-MLC***-SYSTEM**V12-PRxx-EN-P, R911333860 This documentation provides an overview on the hardware/software components of the automation system IndraMotion MLC in the mentioned version. It helps assembling a system.	X	
/9/	Rexroth IndraLogic XLC 12VRS System Overview DOK-XLC***-SYSTEM**V12-PRxx-EN-P, R911333880 This documentation provides an overview on hardware/software components of the automation system IndraLogic XLC in the respective version. It helps assembling a system.		X

About this Documentation

First steps		MLC	XLC
/25/	Rexroth IndraMotion MLC 12VRS First Steps DOK-MLC***-F*STEP**V12-COxx-EN-P, R911333846 This documentation describes the first steps of the IndraMotion MLC and the RobotControl. It includes the hardware and software prerequisites as well as the creation of a project.	X	
/25/	Rexroth IndraLogic XLC 12VRS First Steps DOK-XLC***-F*STEP**V12-COxx-EN-P, R911333876 This documentation describes the first steps of the IndraLogic XLC. It includes the hardware and software prerequisites as well as the creation of a project.		X

xx Corresponding edition
Fig. 1-1: XCL/MLC documentation overview

1.4 Information Representation

1.4.1 Safety Instructions

If there are the safety instructions in the documentation, they contain certain signal words ("Danger", "Warning", "Caution", "Notice") and sometimes a safety alert symbol (according to ANSI Z535.6-2006).

The signal word draws the attention to the safety instruction and indicates the risk potential.

The safety alert symbol (triangular safety reflector with three exclamation marks), preceding the signal words "Danger", "Warning", "Caution" indicates hazards for persons.

The safety instructions are represented as follows in this documentation:

DANGER

In case of non-compliance with this safety instruction, death or serious injury **will** occur.

WARNING

In case of non-compliance with this safety instruction, death or serious injury **can** occur.

CAUTION

In case of non-compliance with this safety instruction, minor or moderate injury can occur.

NOTICE

In case of non-compliance with this safety instruction, material or property damage can occur.

1.4.2 Symbols Used

Representation The representation of different symbols can be found under chapter [3.4 Explanation of Signal Words and the Safety Alert Symbol](#), page 31.

1.4.3 Names and Abbreviations

Term	Explanation
AxisInterface	The AxisInterface pools and extends PLCopen motion function blocks and provides an easy-to-operate interface for the drive functionality.
Imc interface	The Imc interface provides an easy-to-operate interface for the control functionality. It includes control signals and parameters for the diagnostics and administration of the control.
IndraMotion MLC	Compact Motion Logic systems with Motion, Robot and Logic Control functionality
IndraWorks Engineering Framework	Project planning and commissioning tool of Bosch Rexroth
IndraDrive	Drive control section
HydraulicDrive	Hydraulic drive control section
Wizard	Wizard guiding the user through several dialogs for an ergonomic data input
Function block	Provides application-specific functionalities
IEC	International Electrotechnical Commission
IAC	Integrated axis controller
sercos	sercos (SErial Realtime COmmunication System) interface is a world-wide standardized interface for the communication between controls and drives
XML	Extensible Markup Language (XML). It is a markup language to represent hierarchical structured data as text data

Fig. 1-2: Names and abbreviations used

2 Important Instructions on Use

2.1 Intended Use

2.1.1 Introduction

Bosch Rexroth products are developed and manufactured according to the state-of-the-art. The products are tested prior to delivery to ensure operating safety and reliability.

WARNING

Personal injury and damage to property due to incorrect use of products!

The products have been developed for the use in an industrial environment and may only be used as intended. If they are not used as intended, situations occur that can result in damage to property and injury to persons.



Bosch Rexroth as manufacturer shall not assume any warranty, liability or payment of damages in case of damage resulting from a non-intended use of the products; the user shall solely bear all risks arising from unintended use of the products.

Before using the Bosch Rexroth products, the following requirements have to be met must to guarantee the intended use of the products:

- Anybody dealing with Bosch Rexroth Products in any way is obliged to read and consent to the relevant safety instructions and the intended use.
- Hardware products may not be altered and have to remain in their original state; i.e., no structural changes are permitted.
- The decompilation of software products or the alteration of source codes is not permitted
- Do not install or operate damaged or faulty products
- It has to be ensured that the products have been installed as described in the relevant documentation

2.1.2 Areas of Use and Application

The products IndraMotion MLC of Bosch Rexroth are intended for Motion/Logic applications.



The IndraMotion MLC must only be used together with the accessories and attachment parts given in this documentation. Components that are not expressly mentioned must neither be attached nor connected. The same applies for cables and lines.

The operation must only be carried out with the component configurations and combinations that were expressly mentioned and with the software and firmware indicated and specified in the respective functional description.

The IndraMotion MLC has been developed for Logic applications as well as for the use in single and multi-axes drive and control tasks.

To allow for application-specific requirements of the IndraMotion MLC, various device types with different performance and interfaces are provided.

Typical areas of application of the IndraMotion MLC are:

Important Instructions on Use

- Handling and mounting systems
- Packaging and food machines
- Printing and paper-processing machinery
- Machine tools
- Presses
- Foundry and rolling technology
- Conveying engineering
- Building automation and control
- Special-purpose machines
- Wood working machinery

The IndraMotion MLC may only be operated under the mounting and installation conditions, position and ambient conditions (temperature, degree of protection, moisture, EMC, etc.) specified in this documentation.

2.2 Unintended Use

Usage of the IndraMotion MLC in applications areas other than those specified or described in the documentation and technical data is considered as "unintended".

The IndraMotion MLC must not be used if

- they are subjected to operating conditions not corresponding to the specified ambient conditions. Operation under water, under extreme temperature fluctuations or under extreme maximum temperatures or in explosion-prone areas is prohibited.
- the intended applications have not expressly been released by Bosch Rexroth. Therefore, please read the information given in the general safety instructions!
- the machine or system is provided with an unsecure internet connection. The Ethernet interface of the IndraMotion MLC is not securely protected against unauthorized access. Thus, only use the control in secure networks (security). Otherwise, malfunctions or a control failure can be caused due to unauthorized accesses

The operation is prohibited according to the following standards and guidelines:

- Lifts (Directive 95/16/EC)
- ATEX (94/9/EC)
- Burner control systems (EN 298)
- Pressure equipment (EN 764-7:2002/AC)

Furthermore, the use as safety-relevant part of controls according to DIN EN ISO 13849 is considered as unintended use. The functional safety has to be ensured by additional safety-relevant certified components.

3 Safety Instructions for Electric Drives and Controls

3.1 Definitions of Terms

Application Documentation	Application documentation comprises the entire documentation used to inform the user of the product about the use and safety-relevant features for configuring, integrating, installing, mounting, commissioning, operating, maintaining, repairing and decommissioning the product. The following terms are also used for this kind of documentation: User Guide, Operation Manual, Commissioning Manual, Instruction Manual, Project Planning Manual, Application Manual, etc.
Component	A component is a combination of elements with a specified function, which are part of a piece of equipment, device or system. Components of the electric drive and control system are, for example, supply units, drive controllers, mains choke, mains filter, motors, cables, etc.
Control System	A control system comprises several interconnected control components placed on the market as a single functional unit.
Device	A device is a finished product with a defined function, intended for users and placed on the market as an individual piece of merchandise.
Electrical Equipment	Electrical equipment encompasses all devices used to generate, convert, transmit, distribute or apply electrical energy, such as electric motors, transformers, switching devices, cables, lines, power-consuming devices, circuit board assemblies, plug-in units, control cabinets, etc.
Electric Drive System	An electric drive system comprises all components from mains supply to motor shaft; this includes, for example, electric motor(s), motor encoder(s), supply units and drive controllers, as well as auxiliary and additional components, such as mains filter, mains choke and the corresponding lines and cables.
Installation	An installation consists of several devices or systems interconnected for a defined purpose and on a defined site which, however, are not intended to be placed on the market as a single functional unit.
Machine	A machine is the entirety of interconnected parts or units at least one of which is movable. Thus, a machine consists of the appropriate machine drive elements, as well as control and power circuits, which have been assembled for a specific application. A machine is, for example, intended for processing, treatment, movement or packaging of a material. The term "machine" also covers a combination of machines which are arranged and controlled in such a way that they function as a unified whole.
Manufacturer	The manufacturer is an individual or legal entity bearing responsibility for the design and manufacture of a product which is placed on the market in the individual's or legal entity's name. The manufacturer can use finished products, finished parts or finished elements, or contract out work to subcontractors. However, the manufacturer must always have overall control and possess the required authority to take responsibility for the product.
Product	Examples of a product: Device, component, part, system, software, firmware, among other things.
Project Planning Manual	A project planning manual is part of the application documentation used to support the sizing and planning of systems, machines or installations.
Qualified Persons	In terms of this application documentation, qualified persons are those persons who are familiar with the installation, mounting, commissioning and operation of the components of the electric drive and control system, as well as with the hazards this implies, and who possess the qualifications their work

Safety Instructions for Electric Drives and Controls

requires. To comply with these qualifications, it is necessary, among other things,

- 1) to be trained, instructed or authorized to switch electric circuits and devices safely on and off, to ground them and to mark them
- 2) to be trained or instructed to maintain and use adequate safety equipment
- 3) to attend a course of instruction in first aid

User A user is a person installing, commissioning or using a product which has been placed on the market.

3.2 General Information

3.2.1 Using the Safety Instructions and Passing Them on to Others

Do not attempt to install and operate the components of the electric drive and control system without first reading all documentation provided with the product. Read and understand these safety instructions and all user documentation prior to working with these components. If you do not have the user documentation for the components, contact your responsible Bosch Rexroth sales partner. Ask for these documents to be sent immediately to the person or persons responsible for the safe operation of the components.

If the component is resold, rented and/or passed on to others in any other form, these safety instructions must be delivered with the component in the official language of the user's country.

Improper use of these components, failure to follow the safety instructions in this document or tampering with the product, including disabling of safety devices, could result in property damage, injury, electric shock or even death.

3.2.2 Requirements for Safe Use

Read the following instructions before initial commissioning of the components of the electric drive and control system in order to eliminate the risk of injury and/or property damage. You must follow these safety instructions.

- Bosch Rexroth is not liable for damages resulting from failure to observe the safety instructions.
- Read the operating, maintenance and safety instructions in your language before commissioning. If you find that you cannot completely understand the application documentation in the available language, please ask your supplier to clarify.
- Proper and correct transport, storage, mounting and installation, as well as care in operation and maintenance, are prerequisites for optimal and safe operation of the component.
- Only qualified persons may work with components of the electric drive and control system or within its proximity.
- Only use accessories and spare parts approved by Bosch Rexroth.
- Follow the safety regulations and requirements of the country in which the components of the electric drive and control system are operated.
- Only use the components of the electric drive and control system in the manner that is defined as appropriate. See chapter "Appropriate Use".
- The ambient and operating conditions given in the available application documentation must be observed.
- Applications for functional safety are only allowed if clearly and explicitly specified in the application documentation "Integrated Safety Technolo-

Safety Instructions for Electric Drives and Controls

gy". If this is not the case, they are excluded. Functional safety is a safety concept in which measures of risk reduction for personal safety depend on electrical, electronic or programmable control systems.

- The information given in the application documentation with regard to the use of the delivered components contains only examples of applications and suggestions.

The machine and installation manufacturers must

- make sure that the delivered components are suited for their individual application and check the information given in this application documentation with regard to the use of the components,
- make sure that their individual application complies with the applicable safety regulations and standards and carry out the required measures, modifications and complements.
- Commissioning of the delivered components is only allowed once it is sure that the machine or installation in which the components are installed complies with the national regulations, safety specifications and standards of the application.
- Operation is only allowed if the national EMC regulations for the application are met.
- The instructions for installation in accordance with EMC requirements can be found in the section on EMC in the respective application documentation.

The machine or installation manufacturer is responsible for compliance with the limit values as prescribed in the national regulations.

- The technical data, connection and installation conditions of the components are specified in the respective application documentations and must be followed at all times.

National regulations which the user must take into account

- European countries: In accordance with European EN standards
- United States of America (USA):
 - National Electrical Code (NEC)
 - National Electrical Manufacturers Association (NEMA), as well as local engineering regulations
 - Regulations of the National Fire Protection Association (NFPA)
- Canada: Canadian Standards Association (CSA)
- Other countries:
 - International Organization for Standardization (ISO)
 - International Electrotechnical Commission (IEC)

3.2.3 Hazards by Improper Use

- High electrical voltage and high working current! Danger to life or serious injury by electric shock!
- High electrical voltage by incorrect connection! Danger to life or injury by electric shock!
- Dangerous movements! Danger to life, serious injury or property damage by unintended motor movements!
- Health hazard for persons with heart pacemakers, metal implants and hearing aids in proximity to electric drive systems!

Safety Instructions for Electric Drives and Controls

- Risk of burns by hot housing surfaces!
- Risk of injury by improper handling! Injury by crushing, shearing, cutting, hitting!
- Risk of injury by improper handling of batteries!
- Risk of injury by improper handling of pressurized lines!

3.3 Instructions with Regard to Specific Dangers

3.3.1 Protection Against Contact With Electrical Parts and Housings



This section concerns components of the electric drive and control system with voltages of **more than 50 volts**.

Contact with parts conducting voltages above 50 volts can cause personal danger and electric shock. When operating components of the electric drive and control system, it is unavoidable that some parts of these components conduct dangerous voltage.

High electrical voltage! Danger to life, risk of injury by electric shock or serious injury!

- Only qualified persons are allowed to operate, maintain and/or repair the components of the electric drive and control system.
- Follow the general installation and safety regulations when working on power installations.
- Before switching on, the equipment grounding conductor must have been permanently connected to all electric components in accordance with the connection diagram.
- Even for brief measurements or tests, operation is only allowed if the equipment grounding conductor has been permanently connected to the points of the components provided for this purpose.
- Before accessing electrical parts with voltage potentials higher than 50 V, you must disconnect electric components from the mains or from the power supply unit. Secure the electric component from reconnection.
- With electric components, observe the following aspects:
 - Always wait **30 minutes** after switching off power to allow live capacitors to discharge before accessing an electric component. Measure the electrical voltage of live parts before beginning to work to make sure that the equipment is safe to touch.
- Install the covers and guards provided for this purpose before switching on.
- Never touch electrical connection points of the components while power is turned on.
- Do not remove or plug in connectors when the component has been powered.
- Under specific conditions, electric drive systems can be operated at mains protected by residual-current-operated circuit-breakers sensitive to universal current (RCDs/RCMs).

Safety Instructions for Electric Drives and Controls

- Secure built-in devices from penetrating foreign objects and water, as well as from direct contact, by providing an external housing, for example a control cabinet.

High housing voltage and high leakage current! Danger to life, risk of injury by electric shock!

- Before switching on and before commissioning, ground or connect the components of the electric drive and control system to the equipment grounding conductor at the grounding points.
- Connect the equipment grounding conductor of the components of the electric drive and control system permanently to the main power supply at all times. The leakage current is greater than 3.5 mA.
- Establish an equipment grounding connection with a minimum cross section according to the table below. With an outer conductor cross section smaller than 10 mm² (8 AWG), the alternative connection of two equipment grounding conductors is allowed, each having the same cross section as the outer conductors.

Cross section outer conductor	Minimum cross section equipment grounding conductor	
	Leakage current ≥ 3.5 mA	
	1 equipment grounding conductor	2 equipment grounding conductors
1,5 mm ² (AWG 16)	10 mm ² (AWG 8)	2 × 1,5 mm ² (AWG 16)
2,5 mm ² (AWG 14)		2 × 2,5 mm ² (AWG 14)
4 mm ² (AWG 12)		2 × 4 mm ² (AWG 12)
6 mm ² (AWG 10)		2 × 6 mm ² (AWG 10)
10 mm ² (AWG 8)		-
16 mm ² (AWG 6)	16 mm ² (AWG 6)	-
25 mm ² (AWG 4)		-
35 mm ² (AWG 2)		-
50 mm ² (AWG 1/0)	25 mm ² (AWG 4)	-
70 mm ² (AWG 2/0)	35 mm ² (AWG 2)	-
...	...	...

Fig.3-1: Minimum Cross Section of the Equipment Grounding Connection

3.3.2 Protective Extra-Low Voltage as Protection Against Electric Shock

Protective extra-low voltage is used to allow connecting devices with basic insulation to extra-low voltage circuits.

On components of an electric drive and control system provided by Bosch Rexroth, all connections and terminals with voltages between 5 and 50 volts are PELV ("Protective Extra-Low Voltage") systems. It is allowed to connect devices equipped with basic insulation (such as programming devices, PCs, notebooks, display units) to these connections.

Safety Instructions for Electric Drives and Controls

Danger to life, risk of injury by electric shock! High electrical voltage by incorrect connection!

If extra-low voltage circuits of devices containing voltages and circuits of more than 50 volts (e.g., the mains connection) are connected to Bosch Rexroth products, the connected extra-low voltage circuits must comply with the requirements for PELV ("Protective Extra-Low Voltage").

3.3.3 Protection Against Dangerous Movements

Dangerous movements can be caused by faulty control of connected motors. Some common examples are:

- Improper or wrong wiring or cable connection
- Operator errors
- Wrong input of parameters before commissioning
- Malfunction of sensors and encoders
- Defective components
- Software or firmware errors

These errors can occur immediately after equipment is switched on or even after an unspecified time of trouble-free operation.

The monitoring functions in the components of the electric drive and control system will normally be sufficient to avoid malfunction in the connected drives. Regarding personal safety, especially the danger of injury and/or property damage, this alone cannot be relied upon to ensure complete safety. Until the integrated monitoring functions become effective, it must be assumed in any case that faulty drive movements will occur. The extent of faulty drive movements depends upon the type of control and the state of operation.

Dangerous movements! Danger to life, risk of injury, serious injury or property damage!

A **risk assessment** must be prepared for the installation or machine, with its specific conditions, in which the components of the electric drive and control system are installed.

As a result of the risk assessment, the user must provide for monitoring functions and higher-level measures on the installation side for personal safety. The safety regulations applicable to the installation or machine must be taken into consideration. Unintended machine movements or other malfunctions are possible if safety devices are disabled, bypassed or not activated.

To avoid accidents, injury and/or property damage:

- Keep free and clear of the machine's range of motion and moving machine parts. Prevent personnel from accidentally entering the machine's range of motion by using, for example:
 - Safety fences
 - Safety guards
 - Protective coverings
 - Light barriers
- Make sure the safety fences and protective coverings are strong enough to resist maximum possible kinetic energy.
- Mount emergency stopping switches in the immediate reach of the operator. Before commissioning, verify that the emergency stopping equip-

Safety Instructions for Electric Drives and Controls

ment works. Do not operate the machine if the emergency stopping switch is not working.

- Prevent unintended start-up. Isolate the drive power connection by means of OFF switches/OFF buttons or use a safe starting lockout.
- Make sure that the drives are brought to safe standstill before accessing or entering the danger zone.
- Additionally secure vertical axes against falling or dropping after switching off the motor power by, for example,
 - mechanically securing the vertical axes,
 - adding an external braking/arrester/clamping mechanism or
 - ensuring sufficient counterbalancing of the vertical axes.
- The standard equipment **motor holding brake** or an external holding brake controlled by the drive controller is **not sufficient to guarantee personal safety!**
- Disconnect electrical power to the components of the electric drive and control system using the master switch and secure them from reconnection ("lock out") for:
 - Maintenance and repair work
 - Cleaning of equipment
 - Long periods of discontinued equipment use
- Prevent the operation of high-frequency, remote control and radio equipment near components of the electric drive and control system and their supply leads. If the use of these devices cannot be avoided, check the machine or installation, at initial commissioning of the electric drive and control system, for possible malfunctions when operating such high-frequency, remote control and radio equipment in its possible positions of normal use. It might possibly be necessary to perform a special electromagnetic compatibility (EMC) test.

3.3.4 Protection Against Magnetic and Electromagnetic Fields During Operation and Mounting

Magnetic and electromagnetic fields generated by current-carrying conductors or permanent magnets of electric motors represent a serious danger to persons with heart pacemakers, metal implants and hearing aids.

Health hazard for persons with heart pacemakers, metal implants and hearing aids in proximity to electric components!

- Persons with heart pacemakers and metal implants are not allowed to enter the following areas:
 - Areas in which components of the electric drive and control systems are mounted, commissioned and operated.
 - Areas in which parts of motors with permanent magnets are stored, repaired or mounted.
- If it is necessary for somebody with a heart pacemaker to enter such an area, a doctor must be consulted prior to doing so. The noise immunity of implanted heart pacemakers differs so greatly that no general rules can be given.
- Those with metal implants or metal pieces, as well as with hearing aids, must consult a doctor before they enter the areas described above.

Safety Instructions for Electric Drives and Controls

3.3.5 Protection Against Contact With Hot Parts

Hot surfaces of components of the electric drive and control system. Risk of burns!

- Do not touch hot surfaces of, for example, braking resistors, heat sinks, supply units and drive controllers, motors, windings and laminated cores!
- According to the operating conditions, temperatures of the surfaces can be **higher than 60 °C (140 °F)** during or after operation.
- Before touching motors after having switched them off, let them cool down for a sufficient period of time. Cooling down can require **up to 140 minutes!** The time required for cooling down is approximately five times the thermal time constant specified in the technical data.
- After switching chokes, supply units and drive controllers off, wait **15 minutes** to allow them to cool down before touching them.
- Wear safety gloves or do not work at hot surfaces.
- For certain applications, and in accordance with the respective safety regulations, the manufacturer of the machine or installation must take measures to avoid injuries caused by burns in the final application. These measures can be, for example: Warnings at the machine or installation, guards (shieldings or barriers) or safety instructions in the application documentation.

3.3.6 Protection During Handling and Mounting

Risk of injury by improper handling! Injury by crushing, shearing, cutting, hitting!

- Observe the relevant statutory regulations of accident prevention.
- Use suitable equipment for mounting and transport.
- Avoid jamming and crushing by appropriate measures.
- Always use suitable tools. Use special tools if specified.
- Use lifting equipment and tools in the correct manner.
- Use suitable protective equipment (hard hat, safety goggles, safety shoes, safety gloves, for example).
- Do not stand under hanging loads.
- Immediately clean up any spilled liquids from the floor due to the risk of falling!

3.3.7 Battery Safety

Batteries consist of active chemicals in a solid housing. Therefore, improper handling can cause injury or property damage.

Risk of injury by improper handling!

- Do not attempt to reactivate low batteries by heating or other methods (risk of explosion and cauterization).
- Do not attempt to recharge the batteries as this may cause leakage or explosion.
- Do not throw batteries into open flames.
- Do not dismantle batteries.

Safety Instructions for Electric Drives and Controls

- When replacing the battery/batteries, do not damage the electrical parts installed in the devices.
- Only use the battery types specified for the product.



Environmental protection and disposal! The batteries contained in the product are considered dangerous goods during land, air, and sea transport (risk of explosion) in the sense of the legal regulations. Dispose of used batteries separately from other waste. Observe the national regulations of your country.

3.3.8 Protection Against Pressurized Systems

According to the information given in the Project Planning Manuals, motors and components cooled with liquids and compressed air can be partially supplied with externally fed, pressurized media, such as compressed air, hydraulics oil, cooling liquids and cooling lubricants. Improper handling of the connected supply systems, supply lines or connections can cause injuries or property damage.

Risk of injury by improper handling of pressurized lines!

- Do not attempt to disconnect, open or cut pressurized lines (risk of explosion).
- Observe the respective manufacturer's operating instructions.
- Before dismounting lines, relieve pressure and empty medium.
- Use suitable protective equipment (safety goggles, safety shoes, safety gloves, for example).
- Immediately clean up any spilled liquids from the floor due to the risk of falling!



Environmental protection and disposal! The agents (e.g., fluids) used to operate the product might not be environmentally friendly. Dispose of agents harmful to the environment separately from other waste. Observe the national regulations of your country.

3.4 Explanation of Signal Words and the Safety Alert Symbol

The Safety Instructions in the available application documentation contain specific signal words (DANGER, WARNING, CAUTION or NOTICE) and, where required, a safety alert symbol (in accordance with ANSI Z535.6-2011).

The signal word is meant to draw the reader's attention to the safety instruction and identifies the hazard severity.

The safety alert symbol (a triangle with an exclamation point), which precedes the signal words DANGER, WARNING and CAUTION, is used to alert the reader to personal injury hazards.



In case of non-compliance with this safety instruction, death or serious injury will occur.

Safety Instructions for Electric Drives and Controls

** WARNING**

In case of non-compliance with this safety instruction, death or serious injury **could** occur.

** CAUTION**

In case of non-compliance with this safety instruction, minor or moderate injury could occur.

NOTICE

In case of non-compliance with this safety instruction, property damage could occur.

4 System Overview

4.1 Hardware

Control hardware The IndraMotion MLC system consists of the control hardware of the IndraControl series of and of a corresponding firmware.

IndraMotion MLC L25/L45/L65 are modular and scalable controls. They combine the benefits of an embedded PC architecture with a standardized terminal-based technology I/O system. They can be used as a universal hardware platform for Motion applications.

The control systems contain the following components:

IndraMotion MLC L25:

- The main features of an embedded control hardware IndraControl L25 in the lower performance segment are:
 - Ethernet 10/100 MBit
 - Onboard sercos III interface (Ethernet)
 - Firmware on a 1 GB Compact Flash Card
 - Connection for Inline I/O terminals (possible I/O extensions on the right side of the control)

IndraMotion MLC L45:

- The main features of an embedded control hardware IndraControl L45 in the middle performance segment are:
 - Ethernet 10/100 MBit
 - Onboard I/O
 - Onboard Profibus DP master/slave
 - Onboard sercos III interface (Ethernet)
 - Profinet and Ethernet-IP adapter
 - Firmware on a 1 GB Compact Flash Card
 - Inline I/O terminals (possible I/O extensions on the right side of the control)

IndraMotion MLC L65:

- The main features of an embedded control hardware IndraControl L65 in the upper performance segment are:
 - Ethernet 10/100 MBit
 - Onboard I/O
 - Onboard Profibus DP master/slave
 - Onboard sercos III interface (Ethernet)
 - Profinet and Ethernet-IP adapter
 - Firmware on a 1 GB Compact Flash Card
 - Inline I/O terminals (possible I/O extensions on the right side of the control)

The following axis types are supported:

- Real axes (interpolation either in the control or in the drive)
- Virtual axes (interpolation only in the control)
- Encoder axes

System Overview

- Controller Axes
- Link Axes

Technical Data Comparison of technical data of the IndraMotion MLC hardware:

IndraControl	CML25.1	CML45.1	CML65.1
Processor (compatible)	Renesas SH 7785/576 MHz	AMD LX800/500 MHz	Intel Celeron M/1 GHz
RAM	128 MB	256 MB	512 MB
Remanent memory	256 KB		
S-RAM on board, battery-buffered	–	8 MB, optional	8 MB, optional
Removable storage (Compact Flash card)	1 GB		
Ready contact	•	•	•
Interfaces			
Ethernet 10/100 MBit	•	•	•
sercos III			
Real-time Ethernet bus	•	•	•
Max. number of participants	48	99	99
Max. number of drives	16	32	64
Max. number of I/Os	32	99	99
C2C sercos III functionality on board	–	–	–
Profinet I/Os	○	•	•
Profibus DP (Master/Slave)	○	•	•
Ethernet IP			
Master (scanner)	–	–	–
Slave (adapter)	•	•	•
USB	–	–	–
Inputs/outputs	–	8DI+8DO	8DI+8DO
I/O extension (Inline terminals)	Up to 64 bytes		
Connection for function modules	• Up to two modules can be connected	• Up to four modules can be connected	
Other			
Max. number of axes	16	32	64
Operating system	VxWorks		

System Overview

IndraControl	CML25.1	CML45.1	CML65.1
Display	Diagnostics Display		
Supports Robot Control The battery-buffered S-RAM module (CFL01.1-Y1) might be required to save the RoCo programs	Yes		

- Default function
- Option
- Not available

Fig. 4-1: Technical data of the IndraMotion MLC



For further information on the IndraMotion MLC system, refer to /9/ "Rexroth IndraMotion MLC12VRS System Overview" (see [chapter 1.3 "Required and Supplementing Documentations" on page 14](#)).

Function modules

Connecting a function module to the CML25.1, CML45.1 and CML65.1 extends the functionality of the control. Currently, the following function modules are available:

- CrossComm sercos II (CFL01.1-Q2); to establish an MLC cross link via sercos fiber optic cable rings (single or double ring) to the IndraMotion MLC L25/L45/L65 equipped with onboard sercos III interface
- CrossComm sercos III (CFL01.1-R3) to establish an MLC cross link link (C2C) via sercos III
- sercos II (CFL01.1-Q2); to connect sercos II drives to the IndraMotion MLC equipped with onboard sercos III interface
- High-performance programmable limit switch PLS (CFL01.1-N1) with a sample rate of 125 µs and 16 outputs
- Fast I/O (CFL01.1-E2); 8 inputs, 8 outputs, 8 freely programmable I/Os
- S-RAM module (CFL01.1-Y1) 8 MByte memory extension, battery-buffered, to save RoCo projects
- Real-time Ethernet (CFL01.1-TP); Real-time Ethernet- und Profibus DP master/slave functionality



For the configuration of the function modules, refer to [Creating a Function Module, page 60](#).

For more information on function modules, refer to /24/ "Rexroth IndraControl Lxx 13VRS Function Modules12VRS" (see [chapter 1.3 "Required and Supplementing Documentations" on page 14](#)).

I/O periphery

The following bus systems are provided for the I/O extension of the IndraMotion MLC:

- Inline, Profibus*, sercos III*, Profinet*

* - IndraMotion MLC MLC L25 is provided with sercos III; Profibus and Profinet are possible via function module.

The configuration of this hardware is described in

- Inline I/O - [Creating Inline I/Os, page 64](#)

System Overview

Visualization devices

- Profibus I/O - [Creating Profibus I/Os, page 71](#)
- sercos III I/O - [Creating sercos III I/Os, page 72](#)
- Profinet I/O - [Creating a Profinet IO Controller, page 71](#)

A variety of HMI devices are available for visualization.

The IndraMotion MLC L25/L45/L65 support the following devices:

- IndraControl VCP - protocol: BRC symbolic via
 - Ethernet
 - Profibus
- IndraControl VEP - Ethernet via "WinStudio" to configure with HMI
- IndraControl VSP
- IndraControl VPP



While Motion and visualization are running, starting and using all other programs and operation system components is not supported (or else Motion can be interrupted).

4.2 Firmware

Compact Flash card To operate the control IndraMotion MLC, a corresponding Compact Flash cards with the system firmware (Runtime) IndraMotion MLC 12VRS is required.

The Compact Flash card in an IndraControl CML L25.1/CML L45.1/CML L65.1 is divided into three partitions:

Partition	Capacity	Description
USER partition	665 Mbytes	User data (recipes, process data etc.) is stored in this partition. The USER partition is accessed from the user program via the respective library functions (SysLibFile.lib, SysLibFileAsync.lib). The partition is visible in Windows using standard card reader devices.
SYSTEM partition	128 Mbytes	The system firmware (operating system, IndraLogic Runtime) is stored in this partition. The SYSTEM partition may not be modified by the user.
OEM partition	128 Mbytes	All application data is stored in this partition. The data is accessed via the "Load PLC program" function (refer to chapter 7.2 "Creating PLC Program" on page 193)

Fig.4-2: Partitions on the Compact Flash card

IndraControl	Retain data (KB)	Retain data PLC (KB)	Retain data PLC persistent (KB)	Phys. available RAM (MB)	User code + data + %I + %Q + constants including on-line change per application (MB)	Flag (KB)	Total RAM requirement per IEC application (MB)	Compact Flash (MB)	User partition (MB)	System partition (MB)	OEM partition (MB)
CML L25.1	~128 (130816)	~64 (65512)	~63 (64488)	~123	12	64	~12	1GB	665	128	128
CML L45.1	~128 (130816)	~64 (65512)	~63 (64488)	~335	24	64	~24	1GB	665	128	128
CML L65.1	~128 (130816)	~64 (65512)	~63 (64488)	~346	36	64	~36	1GB	665	128	128

Fig. 4-3: Memory distribution/Memory requirement of the IndraMotion MLC

Update options

Within a version, the system firmware can be updated via a download function in IndraWorks (refer to [chapter 6.4.9 "Firmware Management" on page 161](#)) (new release).

4.3 Software

IndraWorks

The "IndraWorks Engineering" software is used to commission and configure the IndraMotion MLC. It consists of the following components:

- IndraWorks: Project planning, configuration
- IndraLogic: PLC programming
- IndraWorks HMI: Visualization and user interface
- IndraWorks WinStudio: Configuration tool to create user masks for the IndraWorks HMI

All components are automatically installed. IndraWorks WinStudio is installed as "Lite" version with 500 variables.

PLC functionality

In addition to the standard IndraLogic libraries, libraries with PLCopen function blocks and technology functions are available:

- PLCopen function blocks
- The AxisInterface facilitates the access to axes. It provides a structure-based interface (library: ML_TechInterface.library)
- The Generic Application Template (GAT) enables the simple and fast development of clearly structured Logic and Motion applications

The GAT is available in three variants for the IndraMotion MLC target system:

- GAT^{compact}: For simple applications with few axes
- GAT with central operation mode SFC: For flexible modular machine concepts

System Overview

- GAT with decentral operation mode SFC: For flexible modular machine concepts
- Technology-based function blocks extend the basic functionality of the IndraMotion MLC target system and provide application-specific functionalities such as cam function blocks, programmable limit switch, PID controller and a secure key transmission (library: ML_TechBase.library)
- Technology function blocks extend the basic functionality and provide application-specific functionalities like e.g. FlyingShear, CrossCutter, RegisterController etc. (library: ML_TechMotion.library)
- Functions supporting hydraulic components provide the following tools:
 - Sequential programming for motion sequences
 - Force-controlled traversing and adjustment of several hydraulic cylinders

Additional licenses are provided for:

- IndraWorks CamBuilder to create and edit cams and MotionProfiles (FlexProfile)
- IndraWorks WinStudio with a higher number of usable variables and arrays
- Function for the register control with integrated HMI support



For further information on the IndraMotion MLC system, refer to /9/, [Required and Supplementing Documentations](#), page 14.

Free PLC libraries and PLC libraries by third-party providers:

- OSCAT ("Open Source Community for Automation Technology"): This open source library includes several useful functions in the fields of automation technology und building automation.

The OSCAT library can be found under www.oscat.de.



Note that these are not Bosch Rexroth products. Therefore, we are not liable for completeness, functionality and operating safety and we do not provide any support for these libraries.

5 Commissioning and Operation

5.1 General Information

This chapter describes the commissioning of the hardware, the basic functions of the IndraWorks engineering environment and general procedures.

Refer to /5/, [Required and Supplementing Documentations, page 14](#) for the PLC programming using IndraLogic.

5.2 IndraMotion MLC – Basic Sequences, General Information

Working with IndraWorks and an IndraMotion MLC requires a functioning hardware with compatible firmware components.

See also /25/, [Required and Supplementing Documentations, page 14](#).

Therefore, the following steps are recommended:

- Hardware commissioning
 - [Connecting the Control, page 39](#)
 - [Control Memory Deletion, page 43](#) if required
 - [Setting the IP Address, page 44](#)
 - [Enabling the sercos Bus, page 47](#)

Then prepare the IndraWorks interface for operation and perform the project creation steps in offline mode.

First, the offline activities:

- [Installing and Starting IndraWorks, page 48](#)
- [Creating a Project, page 51](#)
- [Creating a Control, page 53](#) and if required
 - [Creating Onboard I/Os, page 62](#)
 - [Creating Inline I/Os, page 64](#)
 - [Creating Profibus I/Os, page 71](#)
 - [Creating sercos III I/Os, page 72](#)
 - [Creating a Profinet IO Controller, page 71](#)
 - [Creating a Kinematics, page 88](#)
 - [Creating a Function Module, page 60](#)

Omit individual points if they are not necessary in the specific case.

Perform the transition to the online mode in IndraWorks with subsequent drive parameterization or to the offline parameterization with subsequent on-line adjustment.

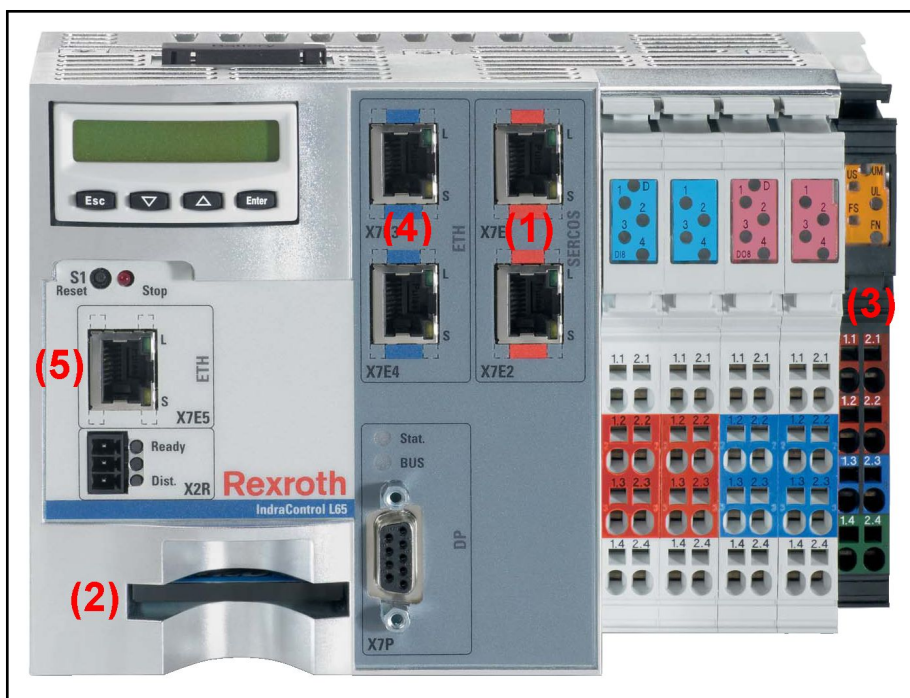
5.3 IndraMotion MLC L25/L45/L65 – Commissioning

5.3.1 Connecting the Control

This section describes the preparation of a control for the configuration using the IndraWorks user interface via Ethernet.

Commissioning and Operation

Ports



- ① sercos III connections
- ② Compact Flash card
- ③ Power supply
- ④ Connections for Ethernet IP, Profinet, Standard Ethernet (only with IndraMotion MLC L45/L65)
- ⑤ Standard Ethernet and Ready contact connection

Fig. 5-1: Example: CML65.1 hardware for IndraMotion MLC applications



Fig. 5-2: Example: CML25.1 hardware for IndraMotion MLC applications

Ready contact behavior

The Ready contact is opened on the software when

- Switching to CP0
- Fatal system error (F9)

The contact is closed when

- switching to CP4



The Ready contact is also opened in case of special hardware problems (e.g. falling below the thresholds of supply voltage or internal voltages etc.).

For more detailed information, refer to the project planning manual of the respective control.

Control startup

Requirements for the initial startup:

- Bridge the sercos III connections X7E1 and X7E2 ① with an Ethernet cable (elimination of possible errors in the sercos bus)
Omit this step with controls without sercos interface.
- Insert the Compact Flash card with the IndraMotion MLC firmware into ②
- Connect network cable with connection to workstation at ⑤
- Provide supply voltage at ③

After switching on the supply voltage, the control runs up to the diagnostic display **BB STOP**.

5.3.2 Operating Control Display

The control display is used to display the control status, configure the network settings and trigger single commands.

The control status is displayed after startup. If there is no error, the status of the PLC (STOP or RUN) and the status of the sercos bus (P0, P2 or BB) are displayed. If an error is pending, the error number is alternately shown with a brief error description. Press ESC on the display to acknowledge the error.

Press "Enter" on the display to reach the control menu. The buttons on the control panel have the following functions:

Enter	Selects the menu item currently shown or goes one level deeper into the menu.
ESC	Exits a menu item or goes one level up in the menu. Once the uppermost level of the menu is reached, exit the menu system by pressing ESC and the status display becomes active again.
Down arrow	Shows the next entry in the menu level.
Arrow up	Shows the previous entry of the menu level.

Commissioning and Operation

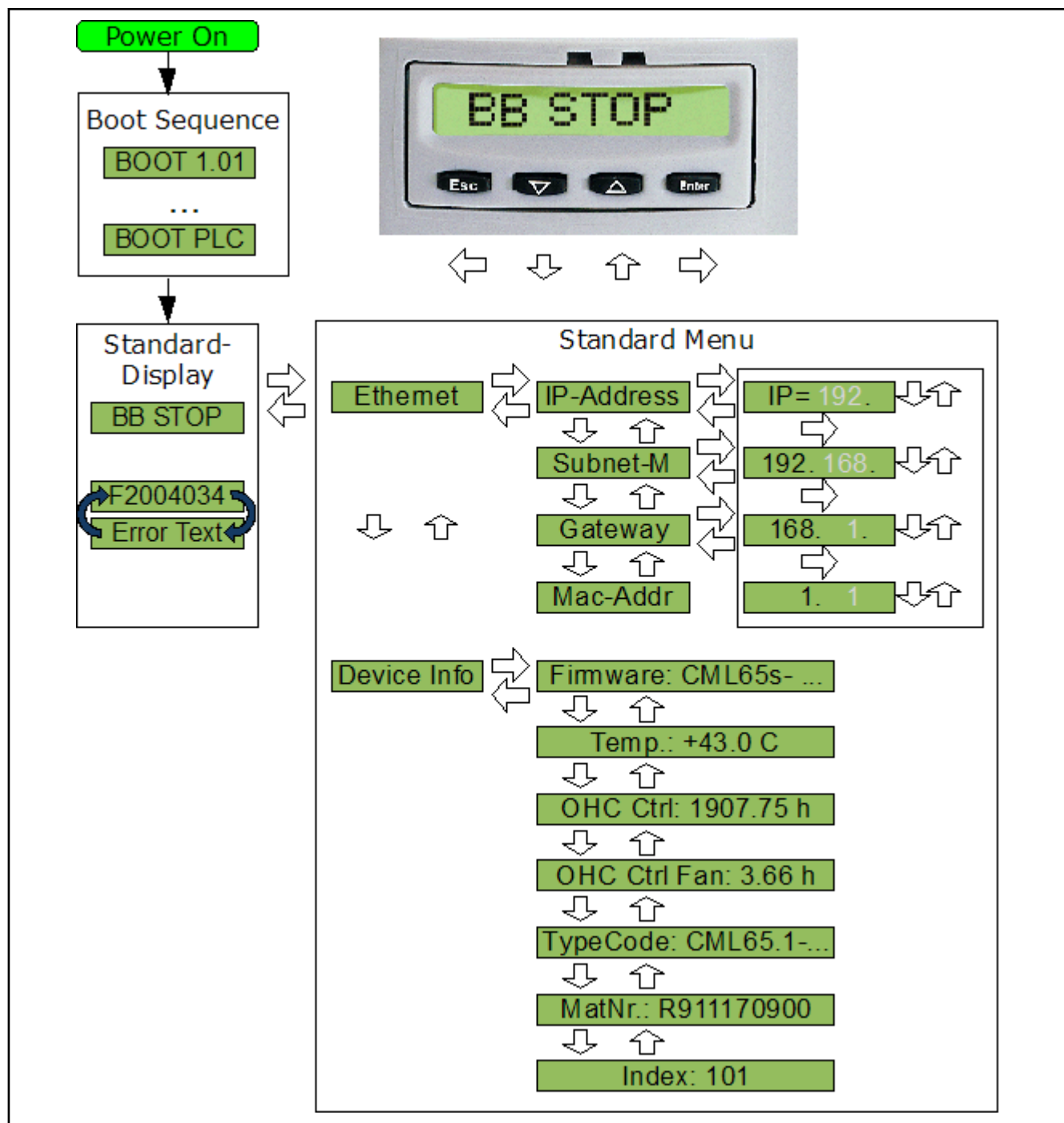


Fig.5-3: Diagnostics and operation of the IndraMotion MLC display

Press ESC and Enter simultaneously at the control panel of the control for eight seconds to switch to the advanced (extended) mode. This shortcut enables the advanced functionality of the control display and is to be used by trained staff only.

Commissioning and Operation

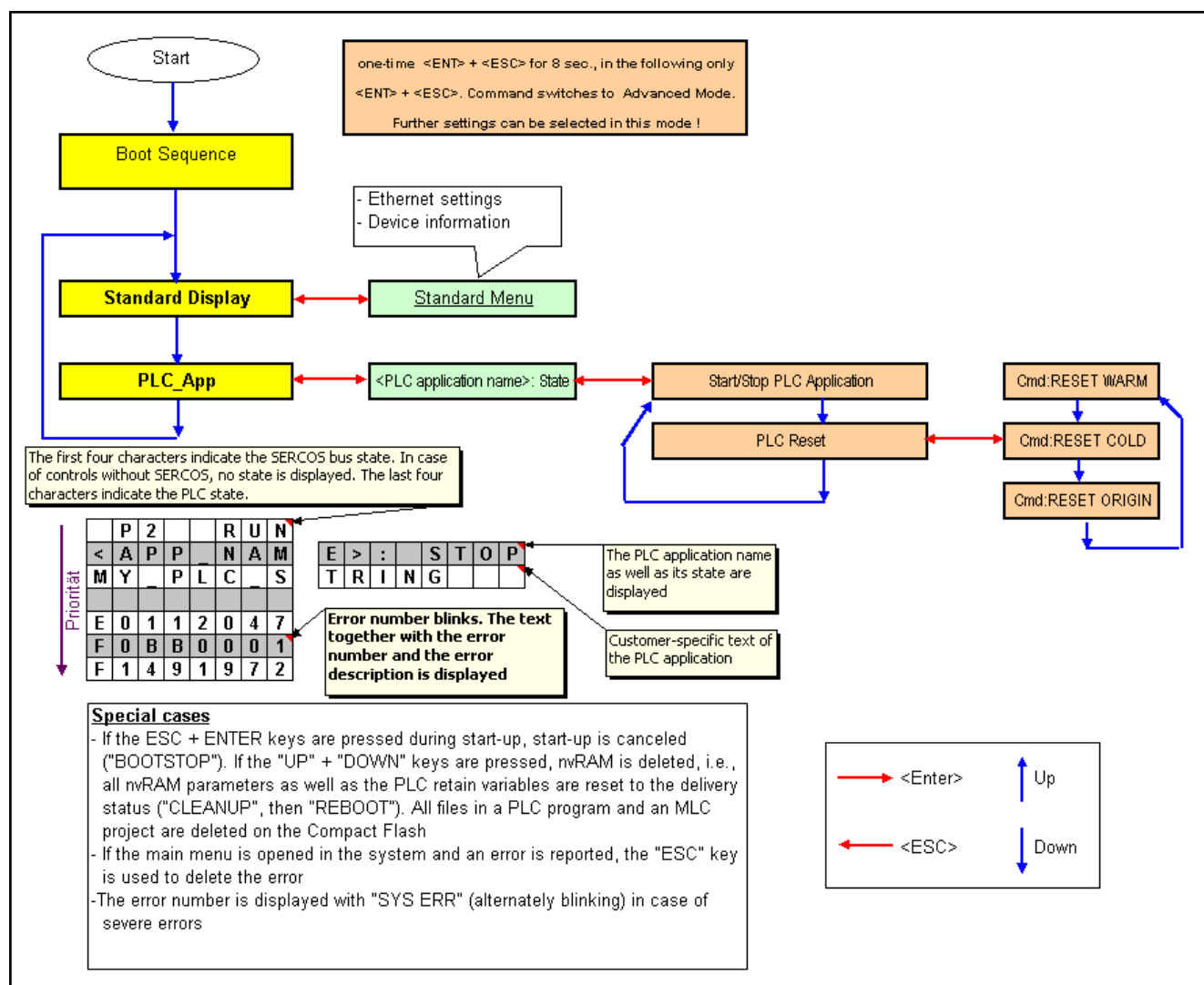


Fig.5-4: Menu in advanced (extended) mode

5.3.3 Control Memory Deletion

From previous use, the control can contain old program components, parameterizations and PLC data that are removed in this step by clearing the memory.



Memory deletion is also mandatory after each firmware change.

When using the function module as C2C interface (control link), consider the note on parameter C-0-0040.

With brand-new controls, this step can be omitted.

Commissioning and Operation

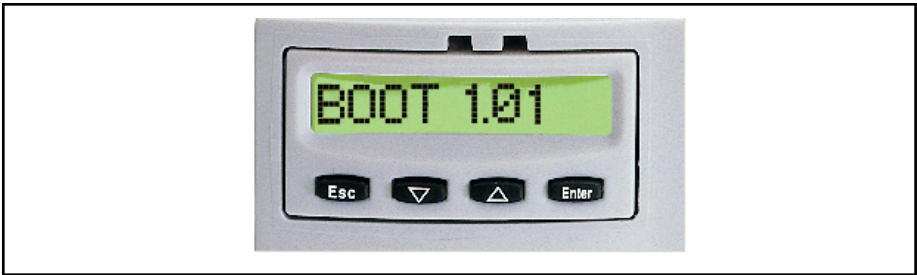


Fig.5-5: Rebooting the control

- 1. After switching on the control, if **BOOT 1.01** is displayed, press both the outer buttons below the display: <ESC> and <Enter>. **BOOTSTOP** is displayed.
- 2. Press both the inner buttons. **CLEANUP** is displayed and - in case of an error-free run - **REBOOT** is displayed. Release the buttons. If an error occurs, **CLEANERR** is displayed.

 If the memory deletion failed several times, inform the customer support, since the display does not provide any help regarding the cause of the failure.

- 3. The control runs until **BB STOP**, the memory is cleared, the Motion components are ready for operation and the PLC components in process are stopped.

5.3.4 Setting the IP Address

IP Address - Initial Setting Before the control can be connected via Ethernet to the programming device (e.g. notebook), it has to have an IP address that is valid for the network.

State of delivery In the state of delivery, the following addresses are set on the control:

Address	Initial value
IP address	192.168.001.001
Subnet mask	255.255.255.0
Standard gateway	192.168.001.001

Fig.5-6: Initial values

Set the IP and subnet mask of a control via the four keys of the control display (see [fig. 5-7 "Display with four operating keys \(standard display\)" on page 45](#)). Afterwards, connect the control to the PC via the company network or via cross-over cables. The cross-over connection requires no address change in the control. Only the programming device address has to be adjusted.

 If the control is connected to the company network, observe the instructions of the network administrator.

In "**BB STOP**" state, read out and set the IP address and the subnet address and determine the current firmware status.

Commissioning and Operation



ESC One level back; discard a change
 v Downward menu navigation or decrement value
 ^ Upward menu navigation or increment value
 Enter Open the next menu level or confirm the entry
Fig.5-7: Display with four operating keys (standard display)

The following figure shows the operating menu of the display based on the standard display:

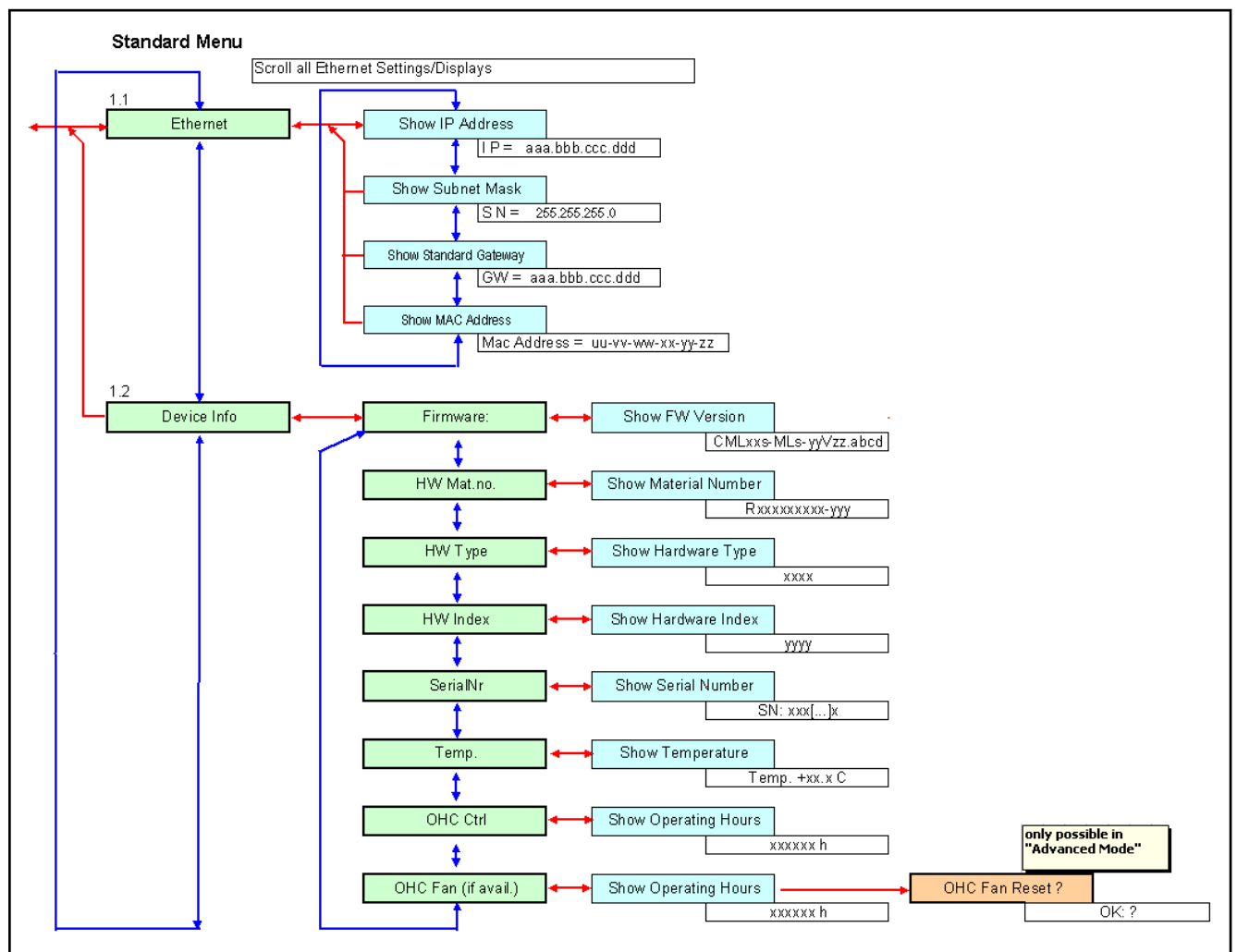


Fig.5-8: Standard menu overview to operate display and keyboard

1. Move with <Enter> from the standard display **BB STOP** to the **ETHERNET** display.
2. Press <Enter> again to reach the **IP address**.

Commissioning and Operation

3. Change the preset address block by block with <ENTER>. (Change numeric values with <DOWN> and <UP> and confirm and move forward with <Enter>). The individual blocks flash to visualize the change.
4. After the fourth block, press <Enter> to trigger the query: OK: ?; Confirm with <Enter>, the change is applied and the **ETHERNET** display opens. If <ESC> is pressed during the change, the change is discarded!
5. Change the subnet mask and, if necessary, the gateway address analogously.

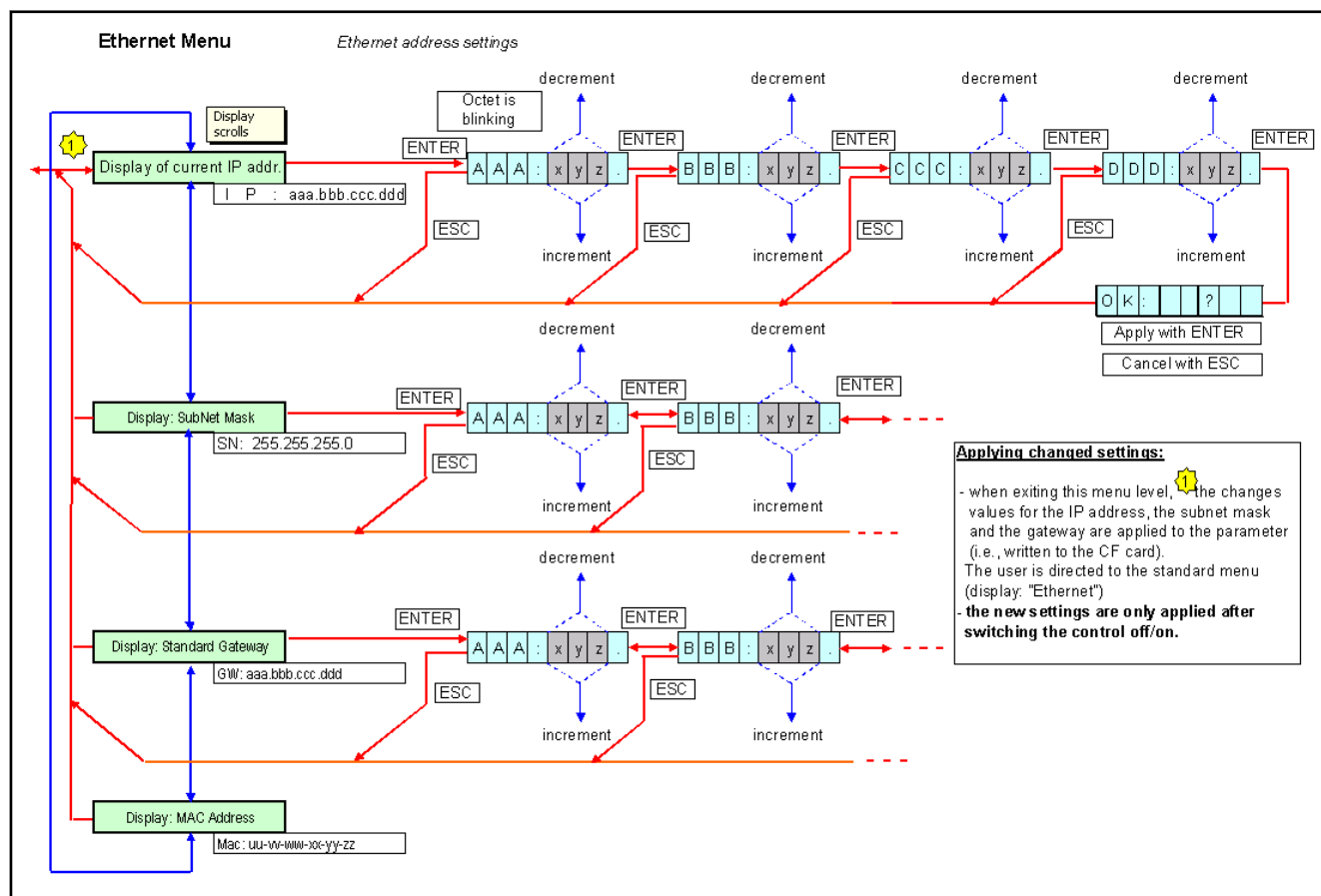


Fig.5-9: Ethernet menu



An address change becomes only effective after the voltage has been switched off and on again (boot). Up to then, the new Ethernet settings are displayed although the previous values are still active.

Identify the MAC address according to the menu overview. Do not modify the MAC address.



After setting the IP address and the subnet mask, they are to correspond to the requirements for future use.

Network settings via the configuration file

Some network settings (IP address of control, Subnet Mask and Standard Gateway) can be changed via a configuration file.

The configuration file must be in an ASCII format and have the file name "Config.txt".

Commissioning and Operation

The "Config.txt" file is located in the USER partition of the Compact Flash card. The Compact Flash card can be accessed with a standard card reader. The configuration file "Config.txt" can be edited with a text editor.

The entries for IP address, Subnet Mask and Standard Gateway can be modified. Use the "Config.txt" template and change the values desired there.

Do not change the spelling of the key words! The syntax of the values must correspond to conventions.

The control applies the changed values following a restart. If the data is not syntactically correct, the control does not apply the modified values but accesses the latest data saved.

Network parameters when delivered

```
[ethernet]
ip_address 192.168.0.65
subnet_mask 255.255.255.0
gateway 0.0.0.0
```



The Ethernet settings (ip_address, subnet_mask und gateway) must not contain leading zeroes.

Cross-over connection

A cross-over cable provides a special coupling between an IndraMotion MLC and a PC. In this case, adjust the IP address of the PC and its subnet mask to the IndraMotion MLC according to the operating system of the PC (e.g. for Windows XP: **Start ► Settings ► Network Connections ► LAN Connection ► Properties ► Internet Protocol TCP/IP ► General**).

If the subnet mask is the same, set an (adjacent) IP address is on the PC:

Address	Control	Notebook (XP) before	Notebook adapted
IP address	11.105.74.212	IP address	11.105.74.213
Subnet mask	255.255.255.128	obtain	255.255.255.128
Standard gateway	Neutral	IP address	Not set

Fig. 5-10: Setting for connection using cross-over

5.3.5 Enabling the sercos Bus

The bridge suggested in section [Ports, page 40](#)., which was used instead of the sercos bus, is now replaced by the bus.

It is necessary that

- every single drive in the ring has the same current firmware (MPH/MPB, "Synchronous operation" function package or HNC3x, "Servo package" function package);
- Each address is only specified once
- the wiring was correctly executed.

The individual drives are automatically synchronized with the baud rate of the master (of the control).

After switching the control off and back on, it is to be possible for the control to reach **BB Stop** and $\square\square\square$ – without power on – for the drives to reach **bb**.

5.3.6 Replacing Hardware

- Goal** The IndraMotion MLC hardware is to be replaced. The project data and firmware are to be transferred from one control hardware to another using a Compact Flash card.

Commissioning and Operation

- Procedure** It is assumed that the control runs in **BB RUN** with active axes. The following steps are required:
1. Switch off the control, replace, wire up the new control. Apply any function modules present and the Inline bus.
 2. Insert the Compact Flash card of the old control into the new control.
 3. Switch on the control.
- After switch-on, the control hardware detects the new Compact Flash card.
The display of the control shows "Hardware replaced; NVRAM restored; A0200017".
The control runs until **BB Run**. That depends on the setting of "C-0-0450, Motion, startup target mode".

5.4 Installing and Starting IndraWorks

- Installing "IndraWorks Engineering"** Before the first use, install the IndraWorks program system of the IndraWorks Engineering Suite.
- For more information on the installation, refer to the manual Rexroth "IndraWorks 12VRS Software Installation" (see /36/, [Required and Supplementing Documentations, page 14](#)).
- In order to ensure executability, the following conditions have to be met:
- Pentium 3 PC, 1 GB RAM (4 GB RAM recommended), 5 GB hard disk capacity
 - Operating systems:
 - Windows XP Professional Service Pack 3
 - Windows Server 2003 Service Pack 2
 - Windows 7 Professional 32 bits
 - Windows 7 Professional 64 bits (recommended)
- Windows 7 with 64 bits is recommended for bigger projects. With this operating system, up to 4 GB application memory are provided for IndraWorks.
-
-  On the installation CD or – in case of an installation from the network – there is an installation manual with notes on any service packages.
-
- IndraWorks – Startup** After the initial start of IndraWorks Engineering (or after **View ► Startup**), the following screen is displayed:

Commissioning and Operation

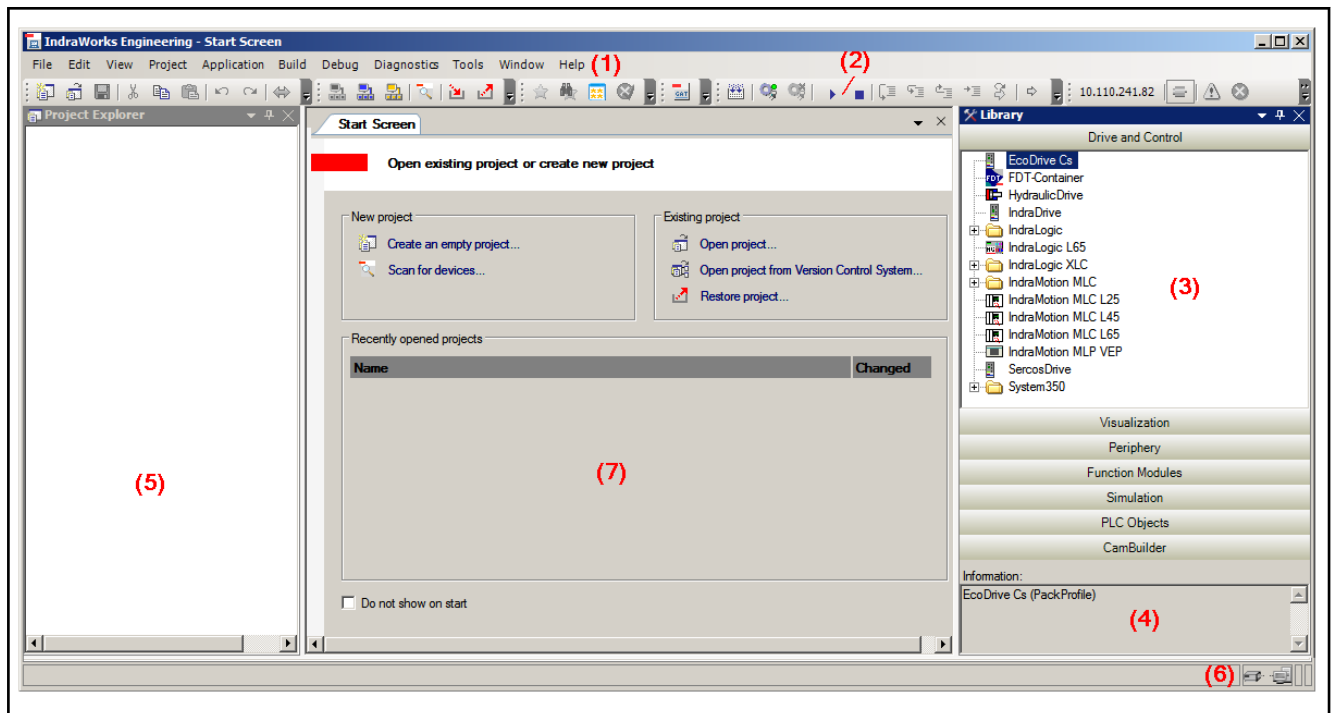


Fig.5-11: Engineering Suite – Startup after the initial start

The figure illustrates the menu ① of the Engineering Suite. Supplement or change the menu items while working in the Suite. Menus that cannot be used are grayed out.

Display/hide ("View" menu item) and configure the toolbars ② below the menu.

The library ③ contains the elements which can be used for the configuration in the Suite. Its elements are stored in the following folders (for help, right-click on the element):

- **Drive and Control:**
 - EcoDrive CS
 - FDT Container
 - HydraulicDrive
 - IndraDrive
 - IndraLogic folder
 - IndraLogic L65
 - IndraLogic XLC folder
 - IndraMotion MLC folder
 - IndraMotion MLC L25
 - IndraMotion MLC L45
 - IndraMotion MLC L65
 - SercosDrive
 - etc.
- **Visualization:**
 - VCP/VCH
 - VEH/VEP

Commissioning and Operation

– VPx/VSx

If an IndraMotion MLC or IndraLogic XLC control is created, the library groups are extended by the following folders:

- Periphery
- Function modules
- PLC objects
- CamBuilder

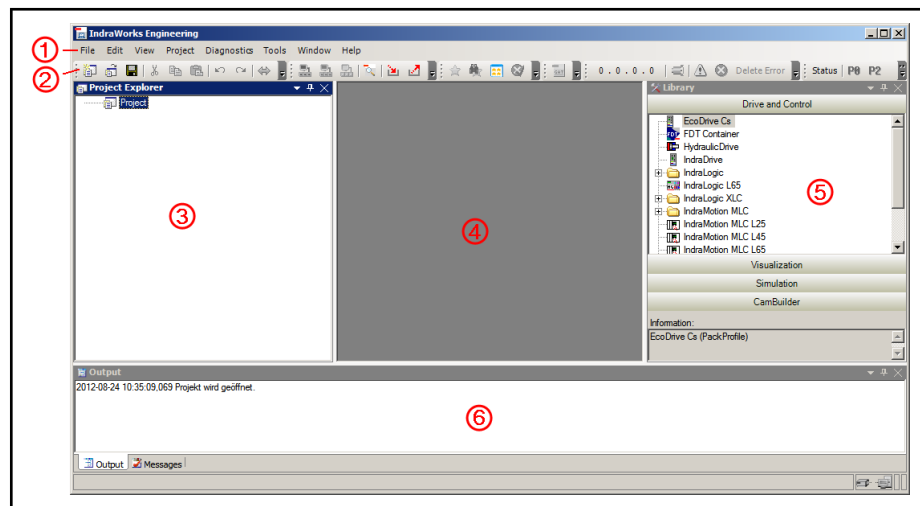
The current library element is described in more detail in Information ④.

Area ⑤ is reserved for the Project Explorer. In the Project Explorer, the library elements are arranged according to the preferences of the user. Working with the Project Explorer is described in the following chapters.

The status bar ⑥ contains information on the current project (online/offline).

As an alternative to the startup, the central area ⑦ provides space for the parameterization and configuration dialogs opened by the user.

5.5 Overview on the IndraWorks Interface



- ① Menu bar
- ② Toolbar
- ③ Project Explorer
- ④ Workspace and document window
- ⑤ Device libraries
- ⑥ Output window

Fig.5-12: IndraWorks interface with default settings

Menu bar ① The menu bar ① and menu items can be freely configured. Move, delete, add and rename complete menus or menu items. To edit the menu items, select **Tools ► Customize...** (or <Alt> + <x> and <a>). All menu items are stored in the "Commands" tab of the "Customize" dialog. From there, press the mouse button to store the individual menu items in any menu of the menu bar. As long as the "Customize" dialog is open, it is possible to call a context menu for each menu and menu item in the menu bar with a right click. Rename or delete (among other functions) entries using the context menu.

Toolbars ② The toolbar ② can be freely configured. Move, delete or add icons. To edit the icons, select **Tools ► Customize...** (or <Alt> + <x> and <a>). In the "Customize" dialog, select and deselect predefined and self-created toolbars. In the "Commands" tab of the same dialog, all menu items are also stored as icons. From there, store the individual icons in any toolbar with the mouse

Commissioning and Operation

button pressed. As long as the "Customize" dialog is open, it is possible to call a context menu for each icon in the toolbar with a right click. Rename or delete (among other functions) entries using the context menu.

Move the individual toolbars with . If the toolbar is drawn over a defined area, it becomes a separate window. In order to dock the toolbar again to the IndraWorks interface, move the window with the toolbar to the required position in the IndraWorks interface.

Project Explorer ③

The Project Explorer is preset on the left of the workspace. The Project Explorer is a structured representation of projects and their components. Right click on a context menu for each item in the Project Explorer.

Workspace and document window ④

The workspace shows document and tool windows to edit different objects of the IndraWorks project.

Device libraries ⑤

The devices available for an IndraWorks project are stored in the device libraries. The devices are classified into different groups:

- Controls (Drive and Controls)
- Visualization devices
- Inline I/O modules
- Field bus modules

Output window ⑥

Status and diagnostic information on the current project are output in the output window.

Windows

Move and dock the individual windows as desired, such as the "workspace" or "library", in IndraWorks. Arrows show possible positions while moving. Drag a window directly to the desired arrow with the mouse button pressed, see [fig. 5-13 "Docking arrows" on page 51](#).

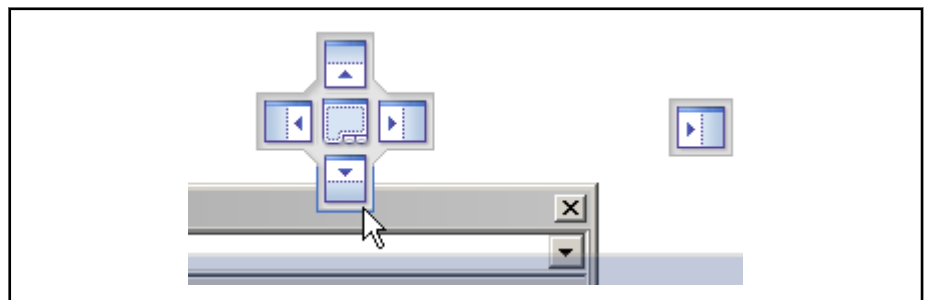


Fig.5-13: Docking arrows

Each window can also be positioned outside the IndraWorks interface. Further settings can be found in the context menu of the window. Go to the context menu of a window via the title bar.

Resetting interface layout

Restore the default interface in the **Window ► Reset window layout** menu.



For more detailed information on the user interface and the handling of IndraWorks, refer to the "Rexroth IndraWorks Engineering; Operating and Programming Instructions" documentation (see /5/, [Required and Supplementing Documentations, page 14](#)).

5.6 Creating a Project

A new project is created via the [IndraWorks – Startup, page 48](#)., or via the **File ► New ► Project** menu item.

Via the following dialog box, determine the project name, the filing directory and the project language.

Commissioning and Operation

It is also possible to specify the fonts for the project via these windows.

Default settings:

- Proportional font: Arial
- Non-proportional font: Courier New

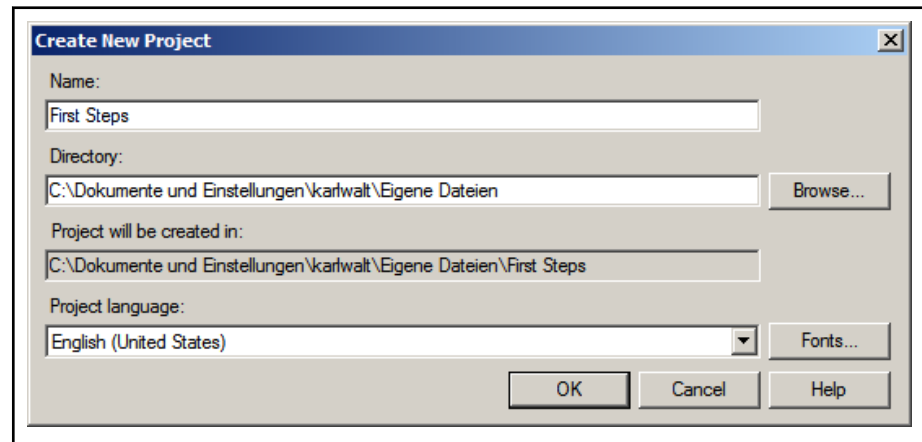


Fig. 5-14: Creating a new project – Specifying the name



When specifying the project name, the file folder name is specified at the same time.

The suggested default name can be replaced by the user.

Context menu on the project

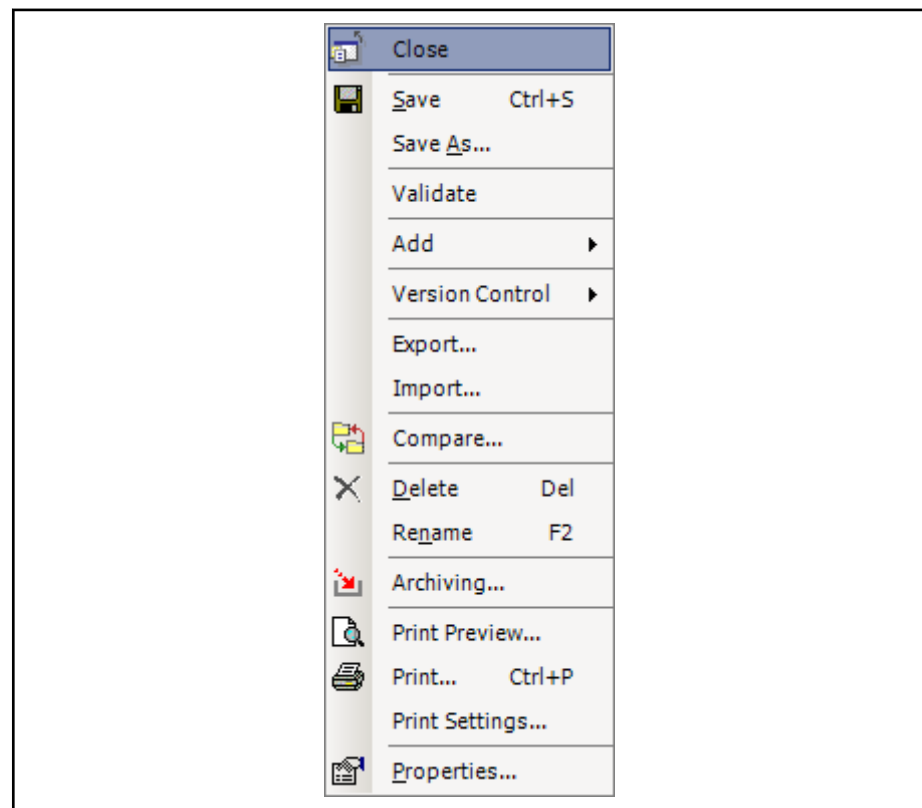


Fig. 5-15: Context menu on an empty project

Right click on the project to open the context menu (or press <Shift> + <F10> on the project).

The context menu of an empty project contains the following menu items:

- **Close** – The current project is closed and saved
- **Save** – The current project state is manually saved
- **Save As...** – The current project state is saved manually with the same content, but under a different name and in a different folder
- **Validate** – IndraWorks checks for configuration errors in the Motion configuration, see [chapter 9.8 "Validating Motion Configuration" on page 379](#)
- **Add** – Enables a new folder, a new file or a new element to be added to the project
- **Version Control** – Enables changes to documents or files to be recorded, see [chapter 6.5 "Version Control of the Control" on page 182](#)
- **Export...** – Saves elements from the current project in a file, see [chapter 6.4.12 "Exporting" on page 172](#)
- **Import...** – Imports elements into the current project from a file, see [chapter 6.4.13 "Importing" on page 174](#)
- **Compare...** – Enables objects to be compared and merged, see [chapter 6.4.14 "Comparing/Merging" on page 176](#)
- **Delete** – Deletes the created project
- **Rename** – Enables the project name to be changed
- **Archiving...** – The current project state is saved as password-protected (optional) storage in the file system or on the device, see [chapter 6.6 "Complete Data Backup" on page 183](#)
- **Print preview...** – Provides a preview of the project and the device data to be printed
- **Print...** – Prints the project data and device data
- **Print settings...** – Selects the font type and size for printing
- **Properties...** – Shows the name and path of the project



By creating controls and axes etc., the context menu is extended by further menu items (e.g. **Switch Devices Online**, **Start Offline Parameterization**).



The changes are saved automatically in a process controlled by the Suite. However, the user can also initiate saving instead.

If the user has a "CamBuilder" license, a "Cam Pool" folder can be created to store cam data via the library ("CamBuilder" tab).

See also /7/, [Required and Supplementing Documentations, page 14](#).

Operating strategy

There are two fundamental options to operate the Engineering Suite:

- Drag elements from the library and drop them in the Project Explorer at the desired position (this option is used with physically existing elements (e.g. control or real axis), see also [Creating a Control, page 53](#)).
- Right click to define the element in the Project Explorer at the desired position (this option is used with logical elements, e.g. virtual axis, see also [Creating a Virtual Axis \(Virtual Master Axis\), page 243](#)).

5.7 Creating a Control

There are two options to create a control in the project:

Commissioning and Operation

Creating a control by dragging and dropping

Create a control in a project by dragging the library element from the respective subfolder (IndraLogic XLC, IndraMotion MLC) of the "Drive and Control" library folder to the project folder. The properties of this element are given in the information window.

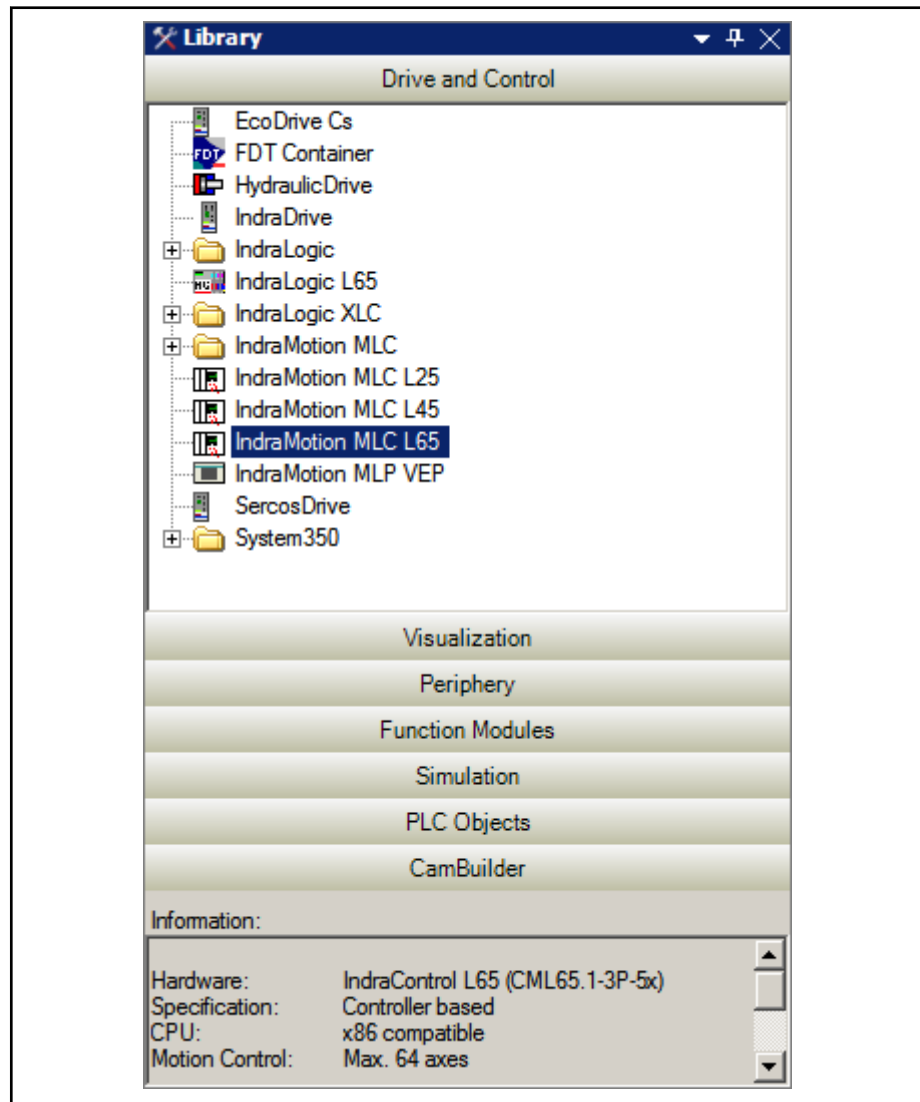


Fig.5-16: Creating a control (as shown for the IndraMotion MLC L65) - dragging from the library

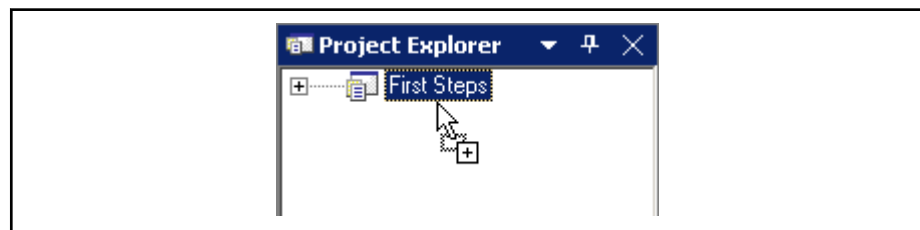


Fig.5-17: Creating a control - dropping it to the Project Explorer

Creating a control using the context menu

Right-click on the project node to open the context menu for the project. Then select the desired control under the menu item "Add".

Commissioning and Operation

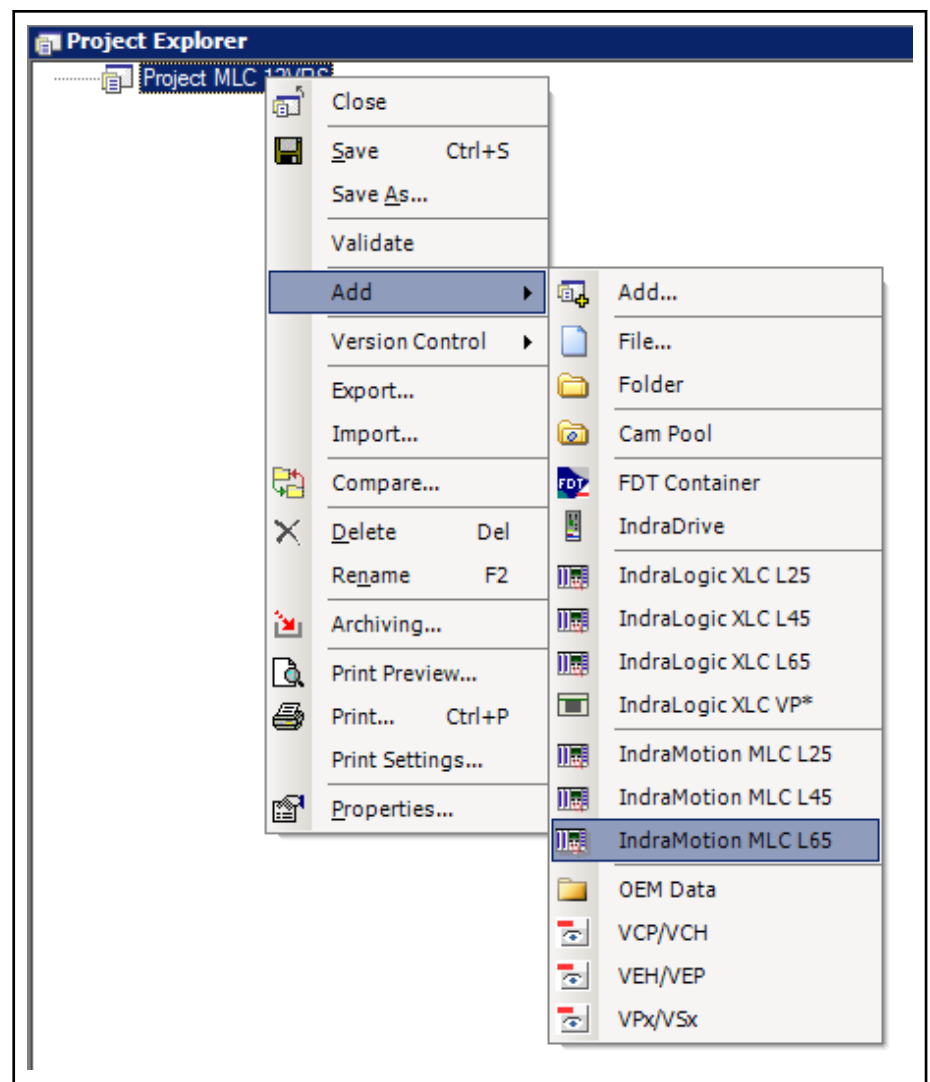


Fig.5-18: Creating an IndraMotion MLC in the project

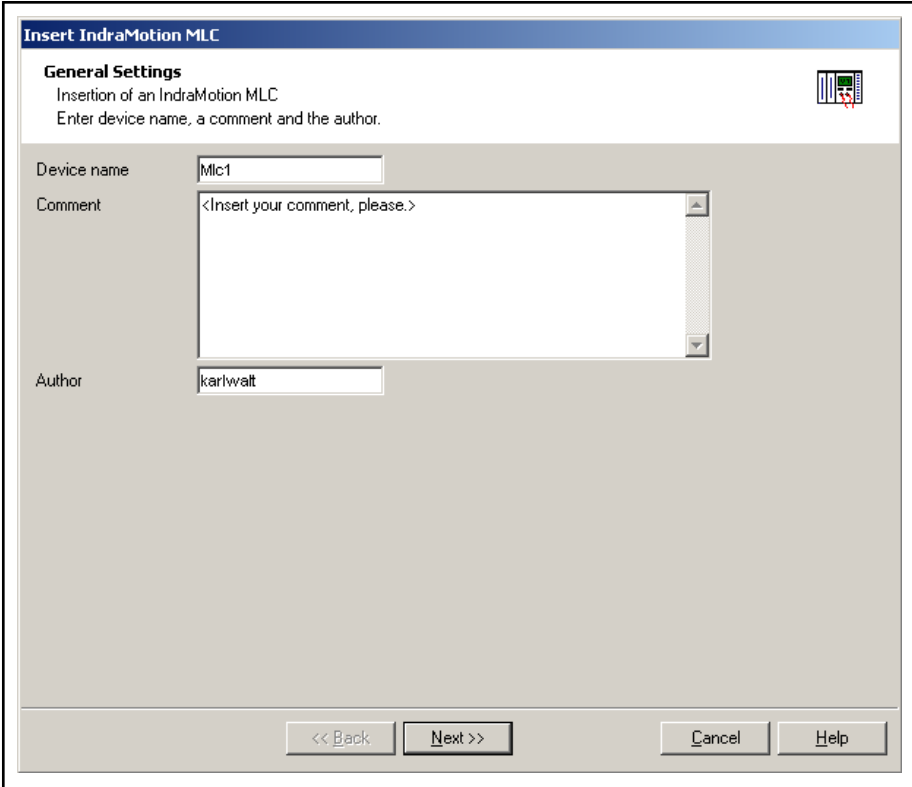
Creating a control by copying

A control can also be added by "copying and pasting". A control that already exists in the project is copied and inserted at the project node.

Dialogs to create a control

Creating a control is supported by a wizard with three dialog windows.

Commissioning and Operation



The screenshot shows a software window titled "Insert IndraMotion MLC". Inside, there is a section labeled "General Settings" with the instruction "Insertion of an IndraMotion MLC" and "Enter device name, a comment and the author." Below this, there are three input fields: "Device name" with the value "Mlc1", "Comment" with the placeholder text "<Insert your comment, please.>", and "Author" with the value "karlwallt". At the bottom of the window, there are four buttons: "<< Back", "Next >>", "Cancel", and "Help".

Fig.5-19: Creating a control - General Settings

In the "General Settings" step, the device name, a comment on this device and the author are entered.

Accept the preset values (comment is optional) or change them.

The name of the author is derived from the user's Windows login.

Commissioning and Operation

Fig.5-20: Creating a control (shown for the IndraMotion MLC L65) - Configuration

Device type The system and the control hardware is set in the "Device type" setting. The control hardware can be read on the type plate of the control.

The following types of hardware are available for the IndraMotion MLC:

Type code	IndraControl	sercos III	Profibus/Profinet
CML25.1-3N	L25	x	-
CML45.1-3P	L45	x	x
CML65.1-3P	L65	x	x

Fig.5-21: IndraMotion MLC hardware variants

The selected hardware platform can be subsequently modified via the "Properties" dialog of the device (calling via the context menu of the device).

Commissioning and Operation

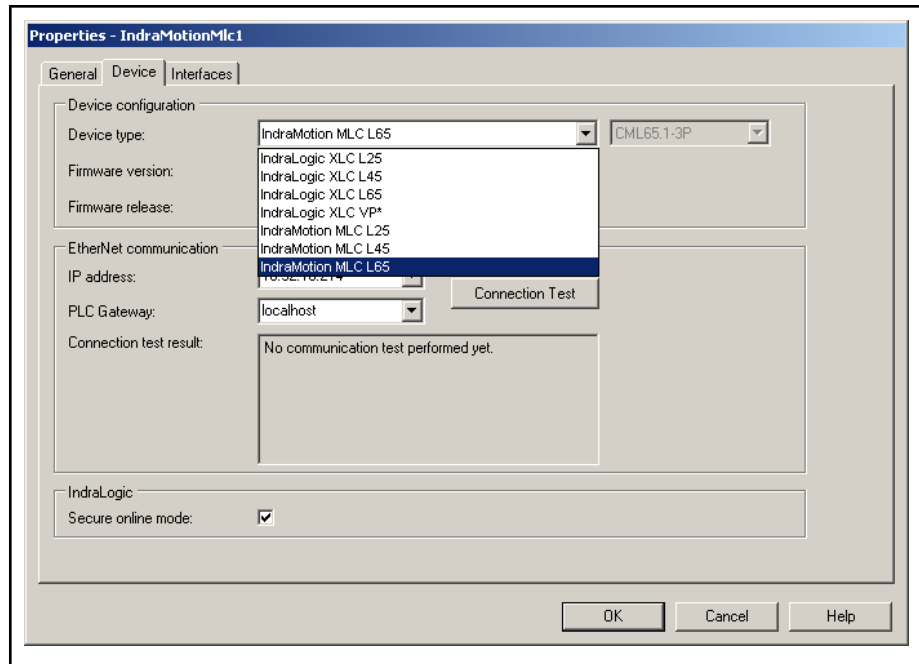


Fig.5-22: Changing the device type



The device types (product family and hardware) differ in their functional scope.

If the new device type set does not support certain features of the source device type, these special configurations are deleted during the conversion (e.g. sercos III available/not available). Features only supported in low numbers are maintained but generate a warning before switching online (e.g. various axis numbers).

Before the configuration of the "Properties" dialog is applied with **OK**, it is checked whether the device type has changed. If there was a change, an information dialog with a change preview is displayed providing information on the additional or unsupported features of the target device. The device conversion can still be cancelled. The device type is only changed when the change preview is confirmed with **OK**.

Firmware version	The firmware version generally corresponds to the IndraWorks Suite version.
Firmware release	The firmware release indicates the firmware state in the version. It can later on be updated considering the set IndraLogic project version. That means the IndraLogic project version has to be updated if necessary.
IP address	The IP address of the control "has to" be set in the Configuration step. Otherwise, communication with the control is not possible. The IP address corresponds to the value set in the Setting the IP Address section, page 44 .
PLC Gateway	The PLC Gateway is generally located on the same computer in which IndraWorks is running. Generally, this setting can thus left unchanged.
Connection test	The connection test checks whether a control with this IP address exists (ping) and responds with the device name of the control and the name of the current user (author). Both can differ from the device name and the author listed in the previous figure.

Commissioning and Operation

Secure online mode The "Secure Online Mode" switch enables additional requests for starting and stopping the PLC and forcing values in the PLC monitoring. It is enabled by default for security reasons.

PLC programming language The PLC programming language only specifies the language for the automatically created standard programs "MotionProg" and "PlcProg". For new programs (PRG), the programming language can be selected when the program is created.

In the third dialog of the wizard, the desired function modules (only for the IndraControl Lxx) have to be preset in their actual order and the use of the onboard interfaces Profibus DP and Ethernet/IP slave has to be defined.

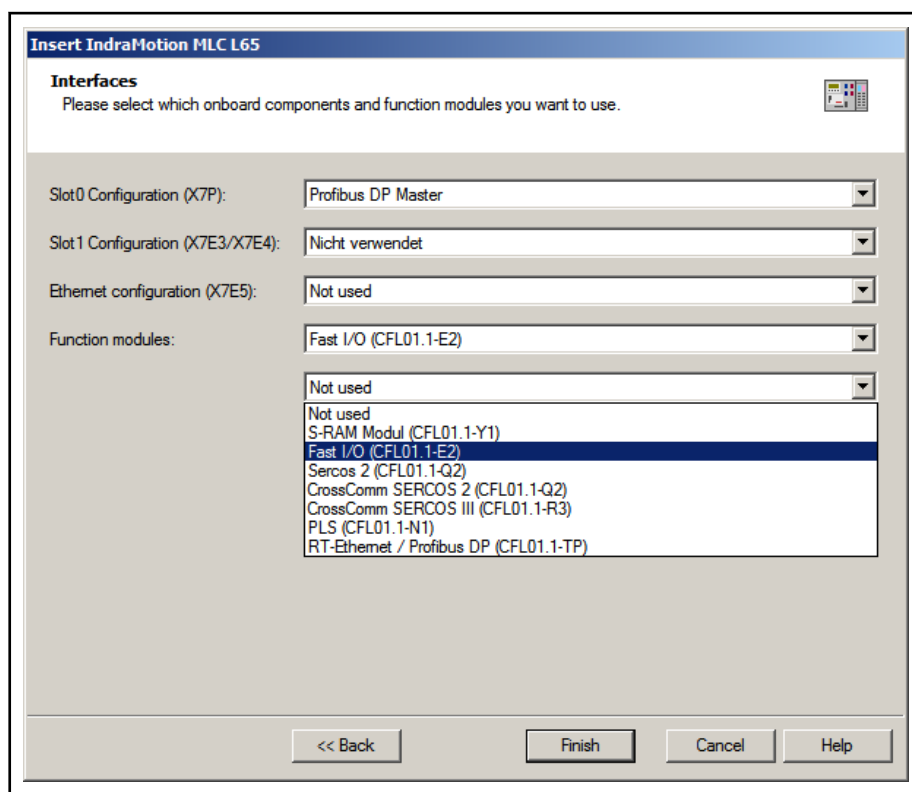


Fig.5-23: Creating a control (illustrated by the example of the IndraMotion MLC L65) - specification of the function modules connected, in this case "Fast I/O (CFL01.1-E2)"



A function module can also be entered into the IndraMotion MLC if it is not yet connected to the control.

While going online, the absence is acknowledged by an error message.

A subsequently connected module is detected at next control startup.

Then click **Finish**.

In the following waiting time, the PLC program belonging to the project is automatically created in the background.

The device (in the example "Mlc1") is thus added to the current project.

Overview in the Project Explorer

The project structure is explained in [chapter 6.1 "Project Structure" on page 93](#).

Commissioning and Operation

Context menu The project structure is explained in [chapter 6.4 "Control – Context Menu"](#) on [page 119](#).

5.8 **Creating a Function Module**

Specification of function module when configuring the control

To extend the functionality of the control, connect up to four (up to two with the IndraMotion MLC L25) function modules at the left of the control.

It is possible to specify a function module when configuring the control (see following figure), even if it is not yet connected to the control.

The module is already integrated into the right location in the Project Explorer.

While going online, the absence is acknowledged by an error message.

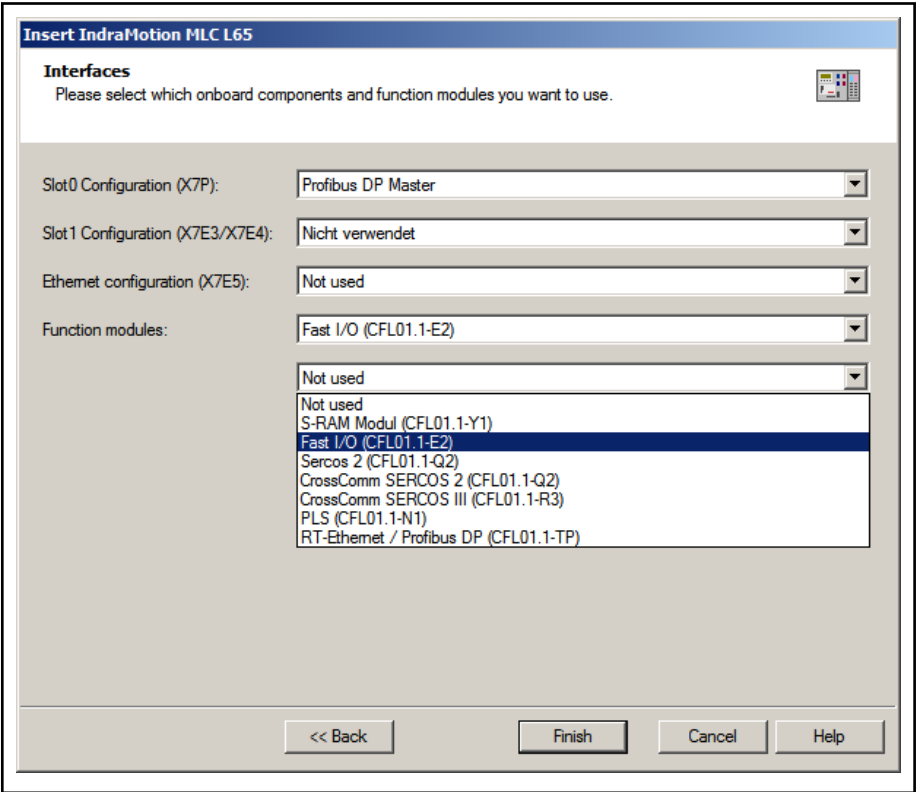


Fig.5-24: Adding a Fast I/O function module via a wizard (for an IndraMotion MLC for example)

Inserting a module from the device library via drag&drop

Function modules to be subsequently connected have to be selected from the library via drag&drop and stored on the control in the Project Explorer.

Procedure:

Drag the function module from the "Function modules" group of the library to the control

Commissioning and Operation

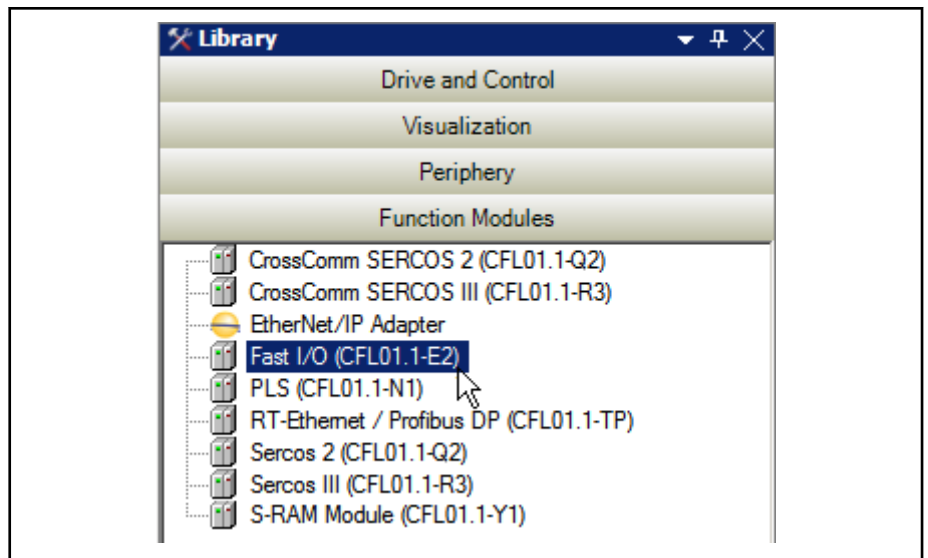


Fig.5-25: Dragging a module from the library
.... and "dropping" it on the control or above the sercos folder.

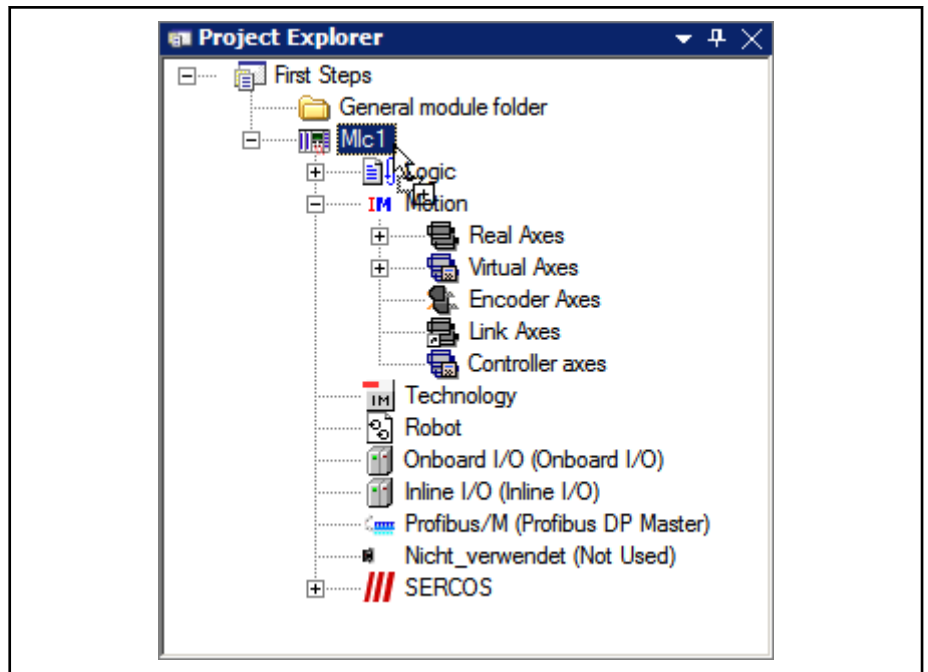


Fig.5-26: Inserting a module



A subsequently connected module is not detected before the next control startup.

Configuring the module

Each module, except for the S-RAM module (CFL01.1-Y1), has to be subsequently configured via corresponding dialogs.

The function modules currently available for the IndraMotion MLC as well as their functionality and configuration are described in /24/, [Required and Supplementing Documentations](#), page 14.

Commissioning and Operation

5.9 I/Os and Field Buses

5.9.1 Creating Onboard I/Os

The "IndraMotion MLC L45" and "IndraMotion MLC L65" provide the user with 1 byte (fast) of inputs/outputs. There is no onboard I/O available on other controls, such as the IndraMotion MLC L25.

The connections can be used byte by byte or bit by bit.



The response time (turnaround time) of the bus from writing to the output to reading the modified input value is

- for 1 ms cycle time – approx. 1 ms
- for 2 ms cycle time – approx. 2 ms

Double-click on the "Onboard_I/O (onboard I/O)" folder in the Project Explorer to open the window.

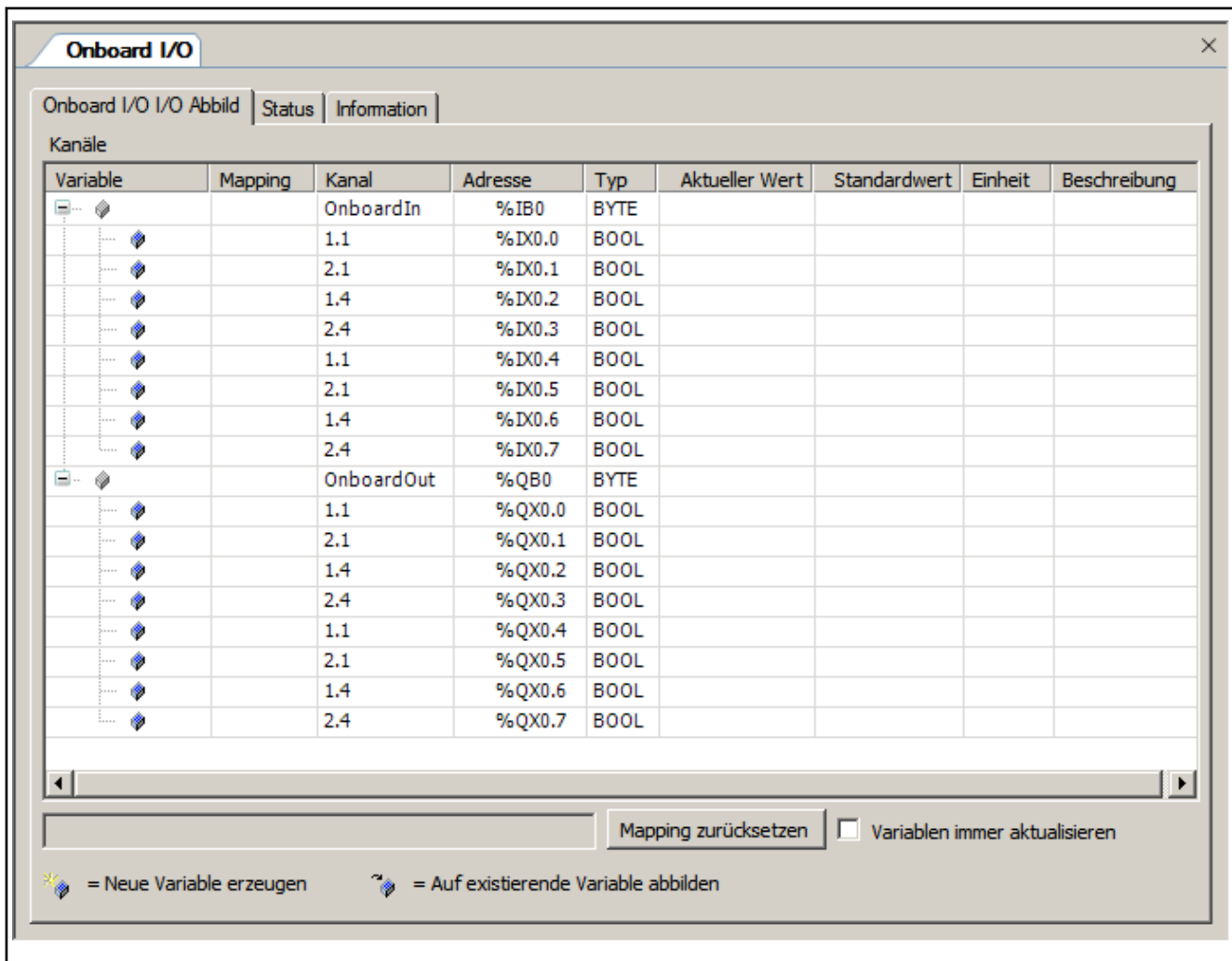


Fig.5-27: Creating a local I/O – Declaration of the inputs and outputs

The starting addresses for the input and output bytes are assigned automatically but can subsequently be changed.



Keep the starting addresses %IB0 or %QB0 for onboard I/Os.

The starting addresses of the Inline I/Os can be set to %IB2 or %QB2 and for Profibus I/O e.g. to %IB500 or %QB500.

"OnboardMaster I/O Image" tab

Assign the I/O ranges of the onboard I/Os to the physical control addresses (I/O addresses of the PLC).

- **Variable:** This column shows input and output modules. Click on the plus or minus icon to switch between byte view and bit view.
Also assign a symbolic address to each absolute address (double-click on the respective field). After it has been entered, the symbolic address is automatically created as global variable in the PLC project.
- **Mapping:** If a new variable is defined, only one variable name has to be entered in the "Variable" column, for example: "bVar1". In this case, the symbol  is inserted in the **Mapping** column. Clicking on the icon with the mouse automatically **declares the variable internally as global variable** and the icon changes to . After this point in time, the variable is globally available in the application.
- **Channel:** Symbolic name of the channel
- **Address:** I/O Address. Enter the desired I/O address as byte address (e.g. %IB10). Entries in italics are for display purposes only and cannot be edited.
- **Type:** Byte addresses are identified with "BYTE" and bit addresses with "BOOL".
- **Current value:** Physical state of the input/output. The state is only displayed in diagnostic mode during communication between IndraWorks and the control.
- **Default value:** Pre-assignment of the variable (TRUE or FALSE)
- **Unit:** Unit for the parameter value, e.g. "ms" for milliseconds
- **Description:** Enter any comment on an address.
- **Reset mapping:** Delete all variable assignments.
- **Always update variable:**
 - Checkmark not set:
Only the variables used in the PLC program are updated.
 - Checkmark set:
All variables are updated.

"Status" tab

Click on the "Status" tab to display the list.

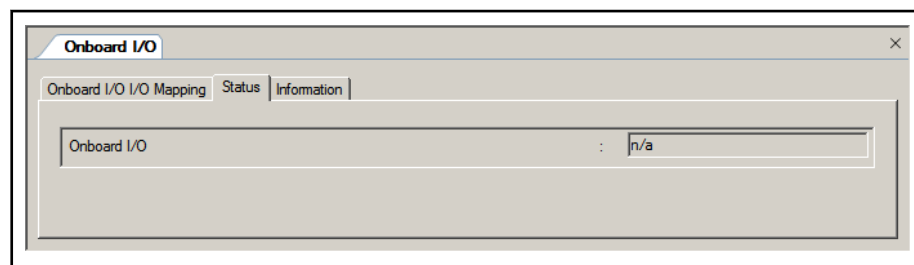


Fig. 5-28: Onboard status workspace

- **OnboardMaster:** This dialog displays the status information (e.g. "Running", "Not running (n/a)" for the respective device.

"Information" tab

Click on the "Information" tab to display the list.

Commissioning and Operation

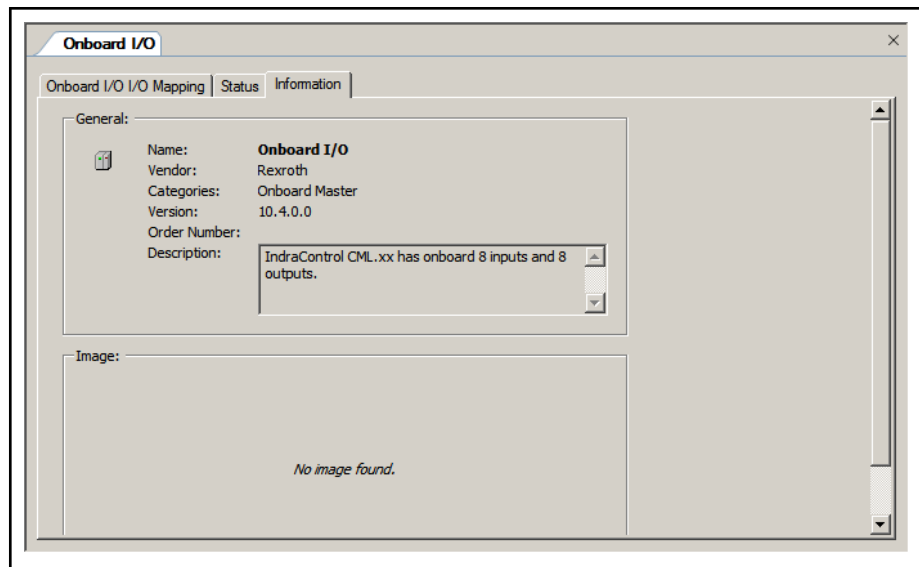


Fig.5-29: Onboard information workspace

This subdialog of the device dialog (Device Editor) displays some general information taken from the device description file regarding the device currently selected from the device tree:

- Name
- Vendor
- Categories
- Version
- Ordering number
- Description
- Image, if available

5.9.2 Creating Inline I/Os

Inline modules are connected at the right of the IndraMotion MLC control.

For details, see /1/, [Required and Supplementing Documentations](#), page 14.



The response time (turn-around time) of the bus for writing to the output to reading the modified input value is

- for 1 ms cycle time – approx. 5 ms
- for 2 ms cycle time – approx. 6 ms



Check that there are no address overlaps, for example between onboard I/Os and Inline I/Os.



Keep the starting addresses %IB0 or %QB0 for onboard I/Os.

The starting addresses of the Inline I/O can then be set to %IB1 or %QB1 and for Profibus I/O e.g. to %IB500 or %QB500.

Creating Inline I/O modules

Insert the Inline modules from the library stock under **Periphery ► Inline** and add them via **drag&drop** or via the context menu/right click **Add ► Module** to the Project Explorer.

Commissioning and Operation

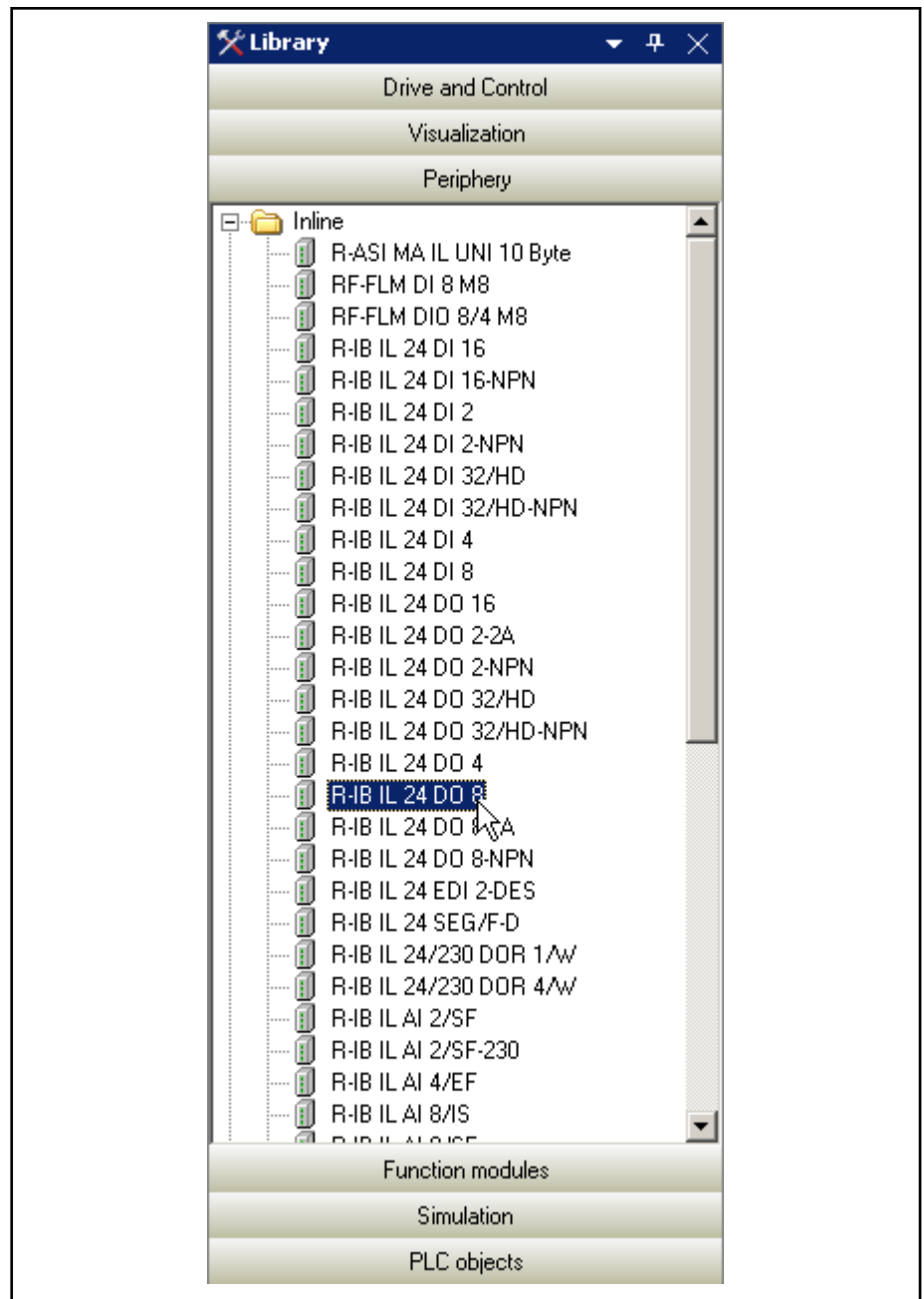


Fig.5-30: Creating Inline I/Os – Selecting and dragging the module from the library

Drag the respective Inline module to the Project Explorer and drop it on to the "Inline_I/O (Inline I/O)" folder there.



Observe the assembly sequence during creation!

Commissioning and Operation

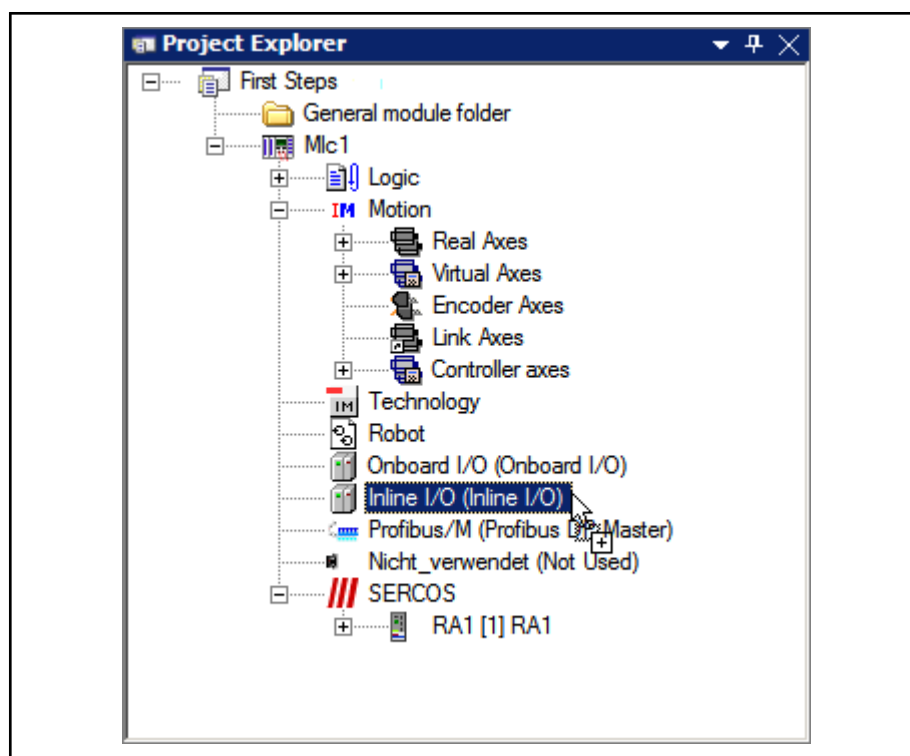


Fig.5-31: Creating Inline I/Os – Filing in the Project Explorer
or

Commissioning and Operation

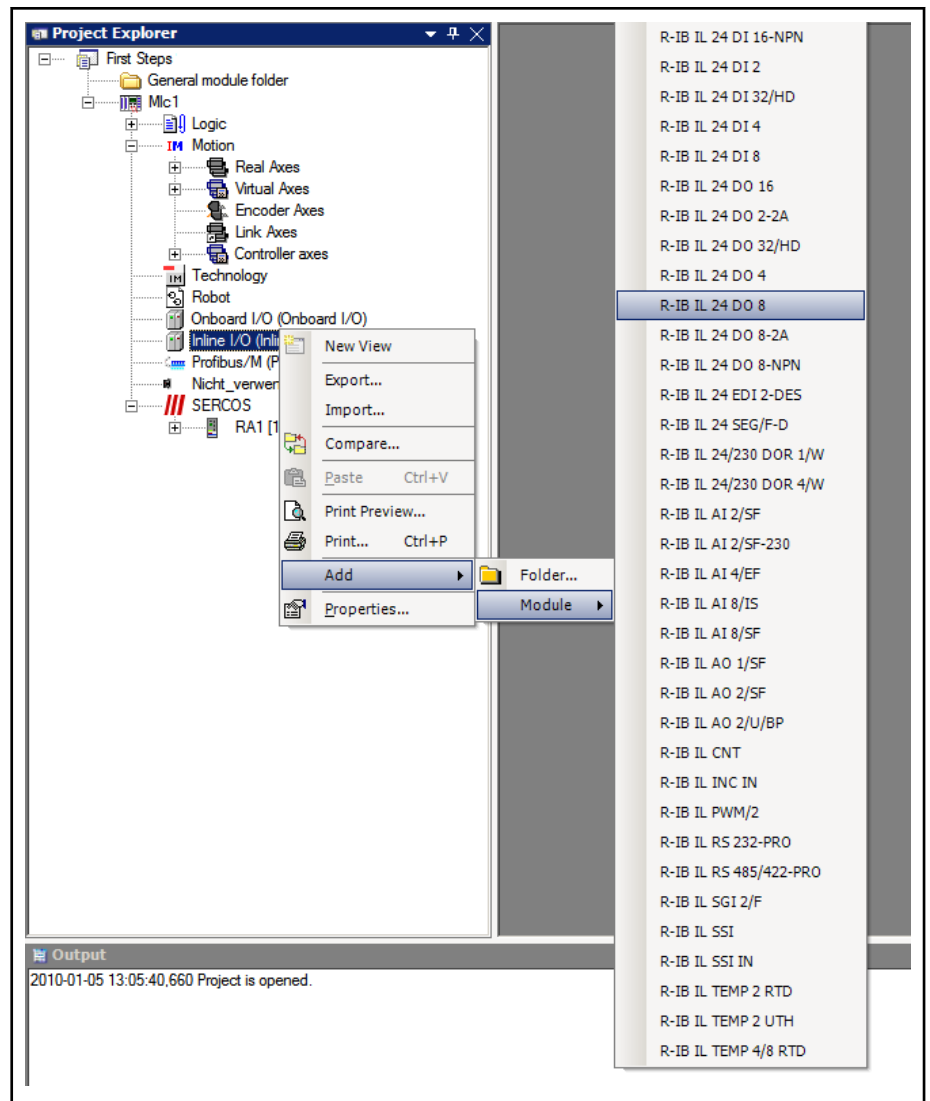


Fig.5-32: Creating Inline I/Os – Selection via context menu

Activating/deactivating modules

"Enable" or "disable" the Inline modules configured on a control in the Project Explorer if required. To do this, select/deselect the button at the icon of the Inline module.

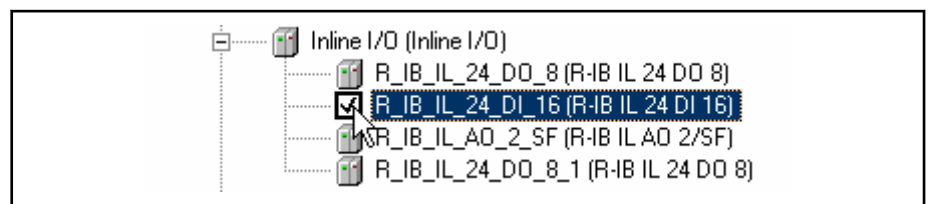


Fig.5-33: Selecting/deselecting an Inline module

If a new Inline module is created, it is enabled by default. When loading the configuration, all enabled Inline modules are taken into account. Inline modules switched to passive mode are ignored.

Commissioning and Operation

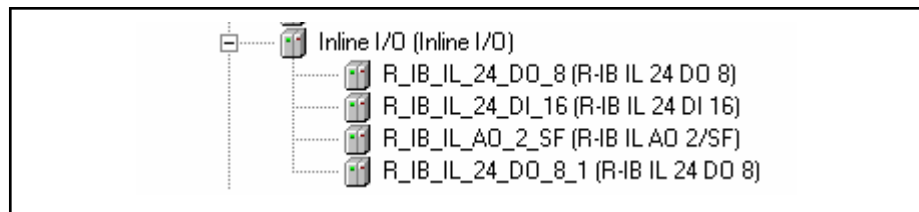


Fig.5-34: Enabled state of an Inline module

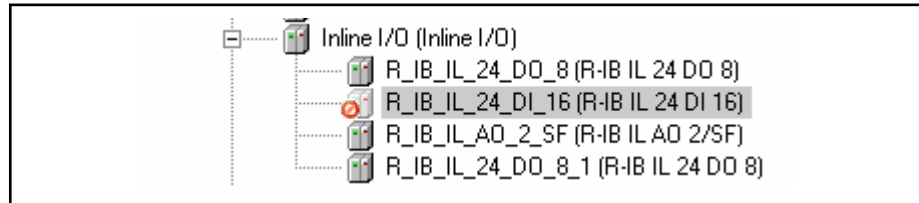


Fig.5-35: Disabled state of an Inline module

Settings Make the required module settings for the respective module.

In the Project Explorer, double-click on the desired Inline module.

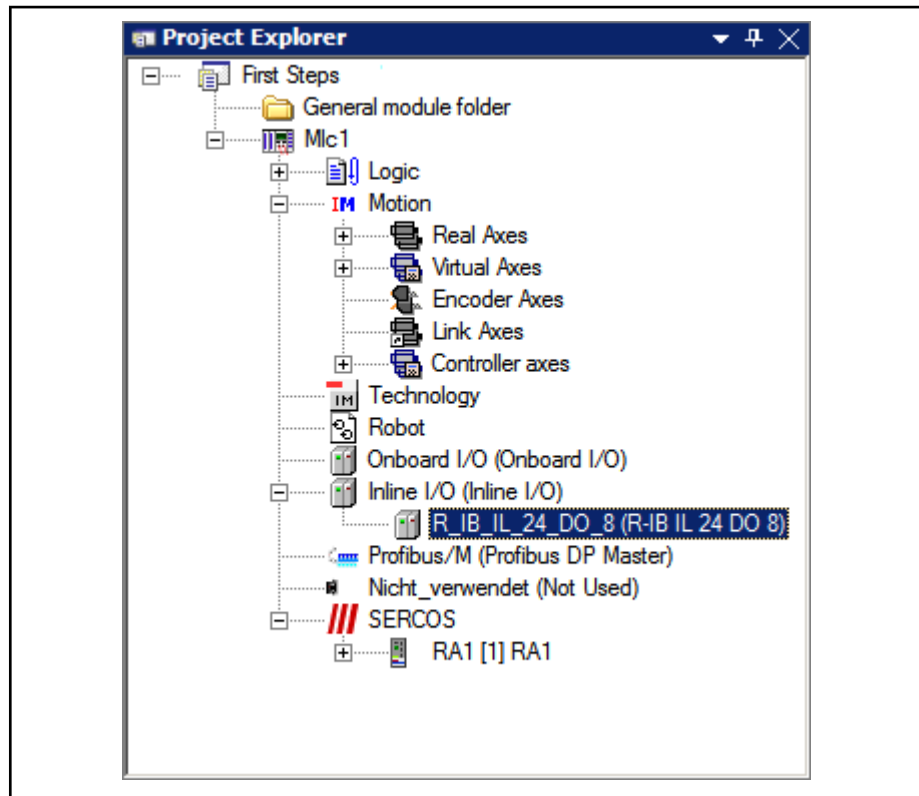


Fig.5-36: Inline I/O module in the Project Explorer

This opens the "Inline Module I/O Mapping" window in the workspace:

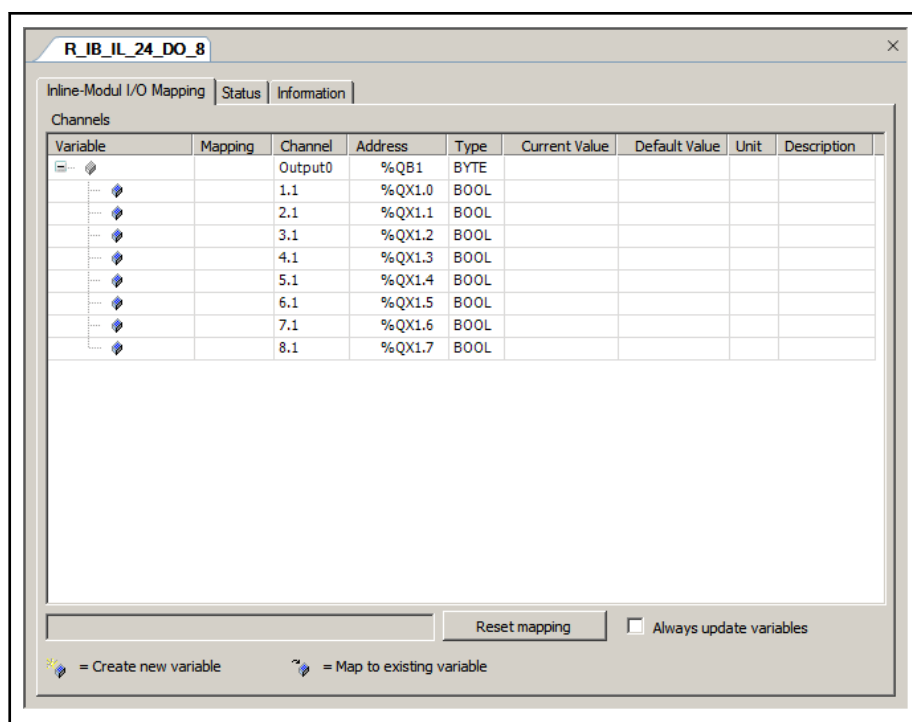


Fig. 5-37: "Inline Module I/O Mapping" workspace

"Inline Module I/O Mapping" tab

Assign the I/O ranges of the Inline I/Os to the physical control addresses (I/O addresses for the PLC).

- Variable:** This column shows the outputs and inputs of the module (in this example). Click on the plus or minus icon to switch between byte view and bit view.
 Also assign a symbolic address to each absolute address (double-click on the respective field). After it has been entered, the symbolic address is automatically created as global variable in the PLC project.
- Mapping:** If a **new variable is defined**, only one variable name has to be entered in the "Variable" column, for example: "bVar1". In this case, the symbol is inserted in the Mapping column. Clicking on the icon with the mouse automatically **declares the variable internally as global variable** and the icon changes to . After this point in time, the variable is globally available in the application.
- Channel:** Symbolic name of the channel
- Address:** I/O Address. Enter the desired I/O address as byte address (e.g. %IB10). Entries in italics are for display purposes only and cannot be edited.
- Type:** Byte addresses are identified with "BYTE" and bit addresses with "BOOL".
- Current value:** Physical state of the input/output. The state is only displayed in diagnostic mode during communication between IndraWorks and the control.
- Default value:** Pre-assignment of the variable (TRUE or FALSE)
- Unit:** Unit for the parameter value, e.g. "ms" for milliseconds
- Description:** Enter any comment on an address.
- Reset mapping:** Delete all variable assignments.

Commissioning and Operation

- **Always update variable:**
 - Checkmark not set:
Only the variables used in the PLC program are updated.
 - Checkmark set:
All variables are updated.

"Status" tab Click on the "Status" tab to display the list.

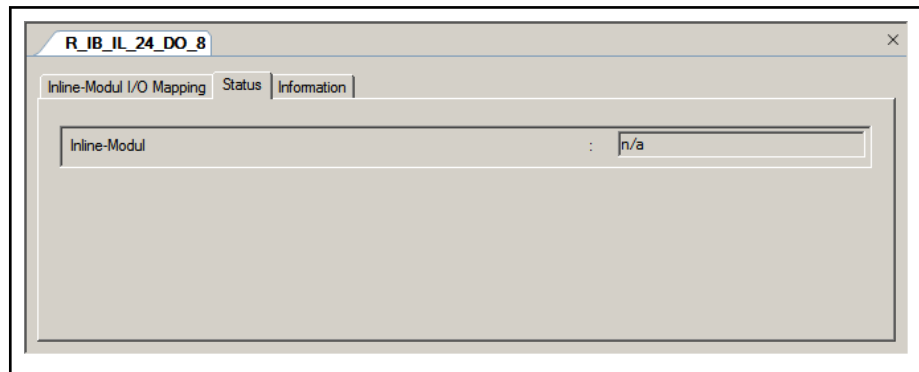


Fig.5-38: Inline status workspace

- **Inline module:** This dialog displays the status information (e.g. "Running", "Not running (n/a)" for the respective device.

"Information" tab Click on the "Information" tab to display the list.

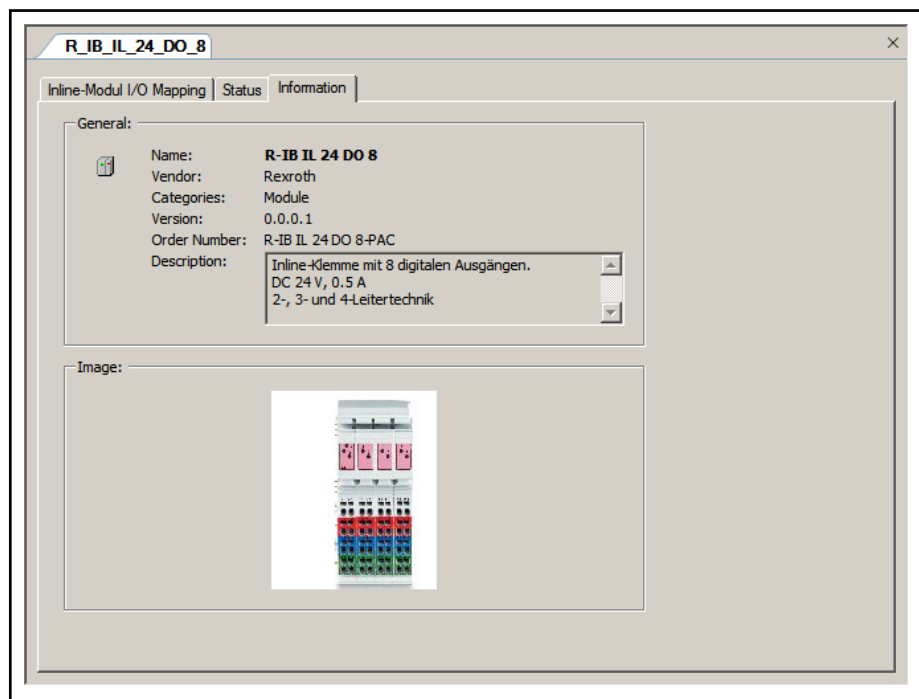


Fig.5-39: Inline information workspace

This subdialog of the device dialog (Device Editor) displays some general information taken from the device description file regarding the device currently selected from the device tree:

- Name
- Vendor
- Categories

- Version
- Ordering number
- Description
- Figure, if available

5.9.3 Creating Profibus I/Os

The Profibus node is located below the Profibus I/O node. By default, this node is configured as master. Via context menu or when creating a control (see Slot0 configuration in [Creating a Control, page 53](#)), this node can also be reconfigured as slave. Thus, the I/O created in the slave can be addressed by a master in another control. However, the rest of this description is limited to the configuration and creation of a master and its slaves.

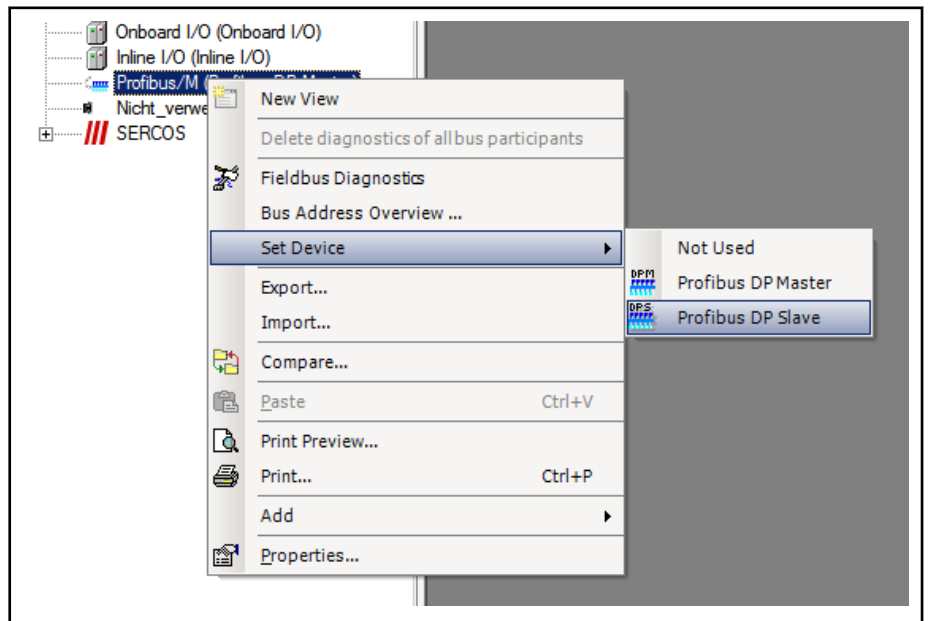


Fig. 5-40: Reconfiguring Profibus node as slave

Profibus DP master and slave configuration

The configuration of the Profibus DP master and slave is described in "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Field Buses" in the chapter "Profibus DP" (see /4/, [Required and Supplementing Documentations, page 14](#))

5.9.4 Creating a Profinet IO Controller

The configuration in the control is based on the device description files for the Profinet I/O Devices and modules used and can still be adapted by the user in the configuration dialogs.

Use the **Scan** functionality to easily determine the existing hardware structure and transfer it to the device tree.

General information on the Profinet I/O:

Profinet IO is based on the Profibus DP function model, but uses **Ethernet TCP/IP** for cyclic communication between the central control and the decentralized field devices. It is well-suited to fast I/O data transfer, but it can also transmit service data, IT functions and parameters at the same time.

The field devices ("Profinet IO Devices") are described by the manufacturer in so-called "device description files", which can be read in using the device database after they have been installed on the local system. The field device signals are cyclically read into the PLC, processed there and output to the field devices again.

Commissioning and Operation

In general, for the Profinet IO, the master-slave procedure known from the Profibus is replaced by a "provider-consumer" model. However, for cyclic user data transfer, the "Profinet IO RT communication" requires a defined master/slave relationship. Therefore, there is also a master here ("Profinet IO Controller"), slaves ("Profinet IO Devices") and under one slave, the I/O modules.

Configuration des Profinet IO-Controller and Device

The configuration and topology for the Profinet IO is described in "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Field Buses" in the chapter "Profinet IO" (see /4/, [Required and Supplementing Documentations, page 14](#))

5.9.5 Creating sercos III I/Os

The sercos III master has the following functional characteristics:

- Support of 32 sercos III I/O devices
- Support of 61 modules per device
- Support of one cyclic producing connection (in AT) and one cyclic consuming connection (in MDT) per device
- Support of devices with the SCP_FixCFG and SCP_VarCFG communication classes
- The "Configuration with connection length" configuration type is supported with the SCP_VarCFG communication class.



sercos III I/O data is only exchanged in the operating mode. If the control is switched to parameterization mode, e.g. for reconfiguration purposes, the I/Os are no longer changed. It is no longer possible to control or display via these I/Os.

Configuration of the sercos III master und slave

The configuration of the sercos II master and slave is described in "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Field Buses" in the chapter "sercos III IO" (see /4/, [Required and Supplementing Documentations, page 14](#))

Hardware exchange of sercos III I/O devices

When changing a sercos III I/O device, the sercos address cannot be set directly on the device, as it is possible e.g. in case of an IndraDrive drive via the display of the drive. For sercos III I/O devices, the address has to be set via the sercos III mechanism "Remote address specification". There are different options:

Option 1: With IndraWorks and the matching IndraWorks project to this machine.

IndraMotion MLC reports the F0140001 error "Drive command configuration incorrect" after the device has been changed. To set the sercos address, perform the following steps:

1. Switch the PLC to STOP.
2. Switch IndraWorks online.
3. The "Adjustment sercos configuration" dialog opens automatically.
4. In this dialog, compare the configuration found with the configured devices and perform the assignment.
5. Set the sercos address of the devices with the **Remote Address Assignment** button.
6. Clear error and restart the PLC.

Option 2: With the IndraMotion Service Tool.

Commissioning and Operation

IndraMotion MLC reports the F0140001 error "Drive command configuration incorrect" after the device has been changed. To set the sercos address, perform the following steps:

1. Log into the IndraMotion Service Tool (rights to change parameters are required).
2. Select Diagnostics -> "Control name" in the navigation tree and check control mode 2. The following steps have to be executed in the parameterization mode P2.
3. Select Data -> Parameters -> C-parameters in the navigation tree.
4. Enter the parameter C-0-0510 (address configuration) in the IDN field and press <RETURN>.
5. A list of found sercos devices is displayed according to the topology address. The topology addresses are output according to their wiring order, starting from X7E1 of the IndraMotion MLC (also for ring wiring). For devices connected to X7E2 of the IndraMotion MLC, enter the last device of the line first.
6. In the parameter list, correct the address of the new device to the address defined in the project. The list is written with <RETURN> and the "remote address specification" is executed.
7. Clear error and switch to operating mode.

Option 3: Via machine HMI with support in the PLC program.

Using the PLC program and a respective extension of the machine HMI, also exchange the devices via the machine HMI. In the following examples, the IMC Interface (IndraMotion Control Interface with the ImcStatus, ImcCtrl structures) from the ML_TechInterface library is used for a simplified access to control data.

Variant 3 a): Permanently defined configuration in the PLC program.

By way of example, writing a configuration permanently defined as constant in the PLC program to the parameter C-0-0510 "Address configuration" is shown: Declaration part:

Program:

```
// This example writes the Parameter C-0-0510, Address configuration
PROGRAM Test_Set_SERCOS_Adr_W

VAR_INPUT
    bConfirmWriteSercosConfig: BOOL;
    // Start writing the sercos configuration
END_VAR

VAR_OUTPUT
    bDoneWriteSercosConfig: BOOL;
    // Successful writing of the sercos configuration
END_VAR

VAR
    fbWriteC510: MB_WriteListParameter;
    Dummy: AXIS_REF;
END_VAR

VAR CONSTANT
    NOF_SERCOS_DEVICES : UINT := 3;
    // define the sercos configuration constant
    arSERCOS_Adr: ARRAY [1..NOF_SERCOS_DEVICES] OF UINT:= [1, 65, 66];
    uiNoOfBytes:  UINT:=2*NOF_SERCOS_DEVICES;
    // no. of SIII devices * 2 (2 bytes per device)
```

Commissioning and Operation

END_VAR

Implementation part:

```

// Error in drive configuration -> write back in Phase 2
// in case of an exchange of an I/O or drive...
// the user has to acknowledge this by
// bConfirmWriteSercosConfig

IF bConfirmWriteSercosConfig = TRUE AND
// conformation by user from HMI
// only allowed in CP2. Alternative to IMC Interface:
// read C-0-0452

ImcStatus.Admin.ModeStatus_P2 = TRUE AND
// allow only when "Error in drive configuration"
// is pending (C-0-0627)

ImcStatus.Diag.Number = 16#F0140001 THEN
// write SERCOS configuration into C-0-510
// "Address configuration"

fbWriteC510(
  Execute:= TRUE,
  ParameterNumber:= GVL_C_Param_ID.FP_C_0_0510,
  NoOfBytes:= uiNoOfBytes,
  ValueAdr:= ADR(arSERCOS_Adr),
  Axis:= Dummy,
  Done=> bDoneWriteSercosConfig);

// error handling not implemented in this example

ELSE

// Nothing to do...

fbWriteC510(
  Execute:= FALSE,
  Axis:= Dummy);

bDoneWriteSercosConfig:=FALSE;
END_IF

```

The sercos configuration is predefined in "arSERCOS_Adr". In this example, the devices with the addresses 1, 65 and 66 are configured and connected in this order to X7E1 of the IndraMotion MLC (or at ring wiring starting from X7E1). If the devices are connected to X7E2, the order has to be reversed [66, 65, 1] if address 1 is directly connected to X7E2.

It is advisable to read out the C-0-0510 on an executable and correctly wired system and then to define "arSERCOS_Adr". Also read out via HMI and store the configuration "arSERCOS_Adr" as RETAIN data (see below).

In the example, writing "arSERCOS_Adr" is triggered via the MB_WriteList-Parameter function block via the "bConfirmWriteSercosConfig" input. The "bConfirmWriteSercosConfig" input has to be set by the HMI.

Check the conditions, the parameterization mode P2 and the pending error F0140001 "Drive command configuration incorrect" as prerequisites. Depending on the application, it may be necessary to check further conditions or specify defined states. After the configuration has been written successfully, the error can be deleted and switch to operating mode. Both can be executed via the ImcCtrl structure of the IMC interface.

Variant 3 b): Reading out configuration via HMI.

By way of example, reading the configuration from the parameter C-0-0510 "Address configuration" and writing back the configuration to this parameter is shown:

Program:

Global variables

```

VAR_GLOBAL RETAIN
  // configuration data of the sercos Addresses
  arSERCOS_Adr: ARRAY [1..99] OF UINT;
  // sercos configuration
  uiNoOfBytes: UINT;
  // no. of sercos III devices * 2 (2 bytes per device)
END_VAR

```

Declaration part:

```

  // This example reads and writes the Parameter C-0-0510,
  // Address configuration

```

PROGRAM Test_Set_SERCOS_Adr_RW

VAR_INPUT

```

  bConfirmReadSercosConfig: BOOL;
  // Start reading the sercos configuration
  bConfirmWriteSercosConfig: BOOL;
  // Start writing the sercos configuration

```

END_VAR

VAR_OUTPUT

```

  bDoneWriteSercosConfig: BOOL;
  // Successful reading of the sercos configuration
  bDoneWriteSercosConfig: BOOL;
  // Successful writing of the sercos configuration

```

END_VAR

VAR

```

  fbReadC510: MB_ReadListParameter;
  fbWriteC510: MB_WriteListParameter;
  Dummy: AXIS_REF;

```

END_VAR

Implementation part:

```

  // read the sercos configuration -> read in case of an exchange of an I/O or drive...
  // the user has to acknowledge this by bConfirmReadSercosConfig

```

IF bConfirmReadSercosConfig = TRUE THEN

```

  // conformation by user from HMI
  // read sercos configuration into C-0-510 "Address configuration"

```

```

  fbReadC510(
    Execute:= TRUE,
    ParameterNumber:= GVL_C_Param_ID.FP_C_0_0510,
    NoOfBytes:= 198,
    ValueAdr:= ADR(arSERCOS_AdrConfig),
    Axis:= Dummy,
    NoOfRecBytes=> uiNoOfBytes_AdrConfig,
    Done=> bDoneReadSercosConfig);

```

```

  IF fbReadC510.Done THEN
    bValidSERCOS_AdrConfig := TRUE;

```

END_IF

```

  // error handling not implemented in this example

```

ELSE

```

  // Nothing to do...

```

```

  fbReadC510(
    Execute:= FALSE,
    Axis:= Dummy);

```

```

  bDoneReadSercosConfig:=FALSE;

```

END_IF

```

  // Error in drive configuration -> write back in Phase 2 in case of an exchange of an
  // I/O or drive... // the user has to acknowledge this by bConfirmWriteSercosConfig

```

IF bConfirmWriteSercosConfig = TRUE AND

```

  // conformation by user from HMI
  ImcStatus.Admin.ModeStatus_P2 = TRUE AND
  // only allowed in CP2. Alternative to IMC Interface: read C-0-0452
  ImcStatus.Diag.Number = 16#F0140001 AND
  // allow only when "Error in drive configuration" is pending (C-0-0627)
  bValidSERCOS_AdrConfig THEN

```

Commissioning and Operation

```
// check that the configuration was read before
// write sercos configuration into C-0-0510 "Address configuration"

fbWriteC510(
    Execute:= TRUE,
    ParameterNumber:= GVL_C_Param_ID.FP_C_0_0510,
    NoOfBytes:= uiNoOfBytes_AdrConfig,
    ValueAdr:= ADR(arSERCOS_AdrConfig),
    Axis:= Dummy,
    Done=> bDoneWriteSercosConfig);

// error handling not implemented in this example

ELSE

    // Nothing to do...

    fbWriteC510(
        Execute:= FALSE,
        Axis:= Dummy);

    bDoneWriteSercosConfig:=FALSE;

END_IF
```

The sercos configuration is read from C-0-0510 into "arSERCOS_Adr" and stored as RETAIN. Reading out has to be triggered on an executable system via HMI. The configuration is stored as RETAIN data in "arSERCOS_Adr". The parameter C-0-0510 is read via MB_ReadListParameter.

As before, writing "arSERCOS_Adr" is triggered via the MB_WriteListParameter function block via the "bConfirmWriteSercosConfig" input. The "bConfirmWriteSercosConfig" input has to be set by the HMI.

After the configuration has been written successfully, the error can be deleted and switch to operating mode. Both can be executed via the ImcCtrl structure of the IMC interface.

5.10 IP Communication

5.10.1 Overview

The control system provides several Ethernet-based interfaces for different tasks and system topologies. The continuous networking based on the Ethernet provides a direct passage between the process and field levels, as well as between the different Ethernet interfaces. Thus, functions, such as a firmware update and fast backup/restoration of the drives of the control via Ethernet, or direct communication of the control with a superordinate HMI or Enterprise Resource Planning System (ERP), are possible.

The Ethernet-based interfaces provide the option of a non-real-time-capable channel based on the TCP/IP protocol family. This chapter describes the configuration and capabilities of the control system to integrate it into an existing network.

To sum up, the following Ethernet-based interfaces are available:

1. The onboard **Engineering interface** is a standard TCP/UDP/IP interface to exchange data with the process control level/HMI and the Engineering PC.
2. The onboard **sercos III** interface is an open, real-time capable data bus to communicate with drives and the I/O of the automation system
3. In preparation: The onboard **Realtime Ethernet** interface is a configurable multi-protocol interface to exchange data with the periphery or a superordinate third-party head control. This interface is configurable and

Commissioning and Operation

supports several Realtime Ethernet protocols such as Ethernet-IP, Profinet, Standard TCP/IP...

4. Up to 64 controls can communicate in a cross link (C2C) in real time and thus exchange master axis positions via the optional **CrossComm sercos III (CFL01.1-R3)** function module for example.

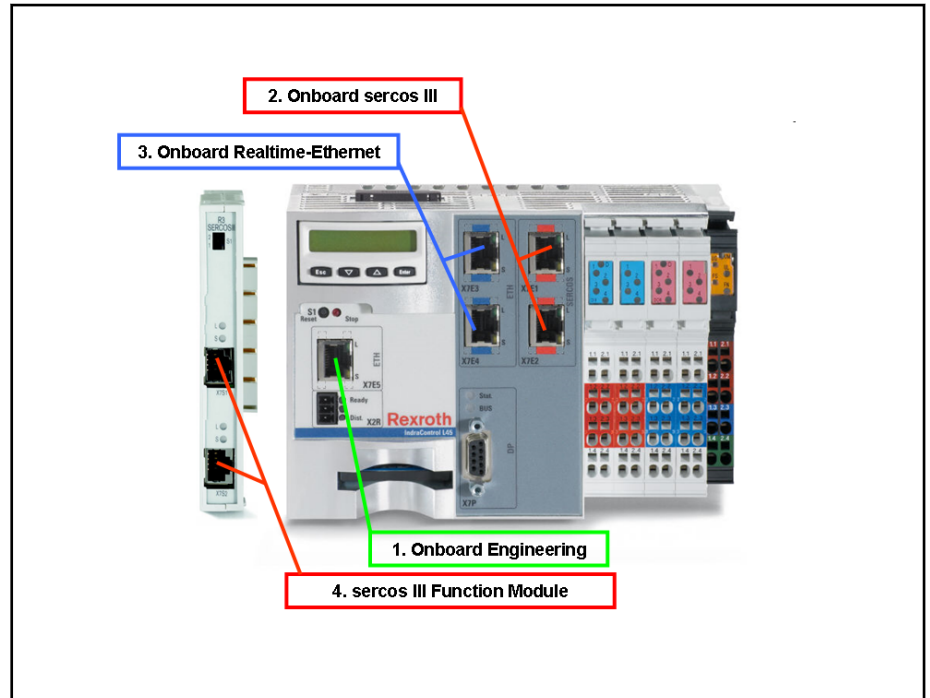


Fig.5-41: Ethernet interfaces

To minimize configuration effort, the control provides presettings/default values for most of the typical automation applications for the above interfaces. The objectives of the default setting are:

- **Connectivity:** Each free Ethernet interface provides the option of participating and the IP communication. Furthermore, communication with each IP device in the automation system can be established.
- **Zero config:** The communication options are provided without additional configuration settings of the IP devices.
- **Flexible:** Implement special requirements and topologies by means of manual IP settings.

5.10.2 Basics

Bridge/bridging

Bridging refers to an IP forwarding procedure. A bridge connects several physical networks to one logical network. A bridge consists of several network interfaces, but is only provided with one IP address. Thus, a bridge with several physical network interfaces can be reached with the same IP address.

A bridge is mainly used to divide a network into different collision domains. Thus, the load in large networks can be reduced, since each network thread can only receive packages, whose recipient is also located in this network. The bridge knows all connected communication devices. It forwards the received data telegram using the MAC address. Bridges work on layer 2 (back-up/data link layer) of the OSI reference model.

- Advantage: Simple design or implementation, zero-maintenance
- Disadvantage: Possible loops can interfere with the network.

Commissioning and Operation

Router/routing

Routing refers to an IP forwarding procedure. A router consists of several network interfaces, which also logically possess their own IP address. In contrast to the bridge, the router analyses the incoming data packages according to their IP target address and forwards them in a corresponding fashion (the packages are routed). Forwarded packages either arrive in a directly connected target network (also target subnetworks) known to the router itself or they are forwarded to another router, also located in a directly connected network. Routers work on layer 3 (placement/network layer) of the OSI reference model.

- Advantage: Network size is not limited, looping is not important
- Disadvantage: More technical effort than for bridging, greater configuration effort

5.10.3 Default Setting

The IP default values contain the following settings:

- All sercos III interfaces of the control are part of the **bridge** and thus have only one IP address. **Routing** is used as the forwarding procedure between all other Ethernet interfaces. The routing table is generated internally and automatically.
- The **Automatic IP address generation** for all sercos III interfaces (on-board, sercos III drives and I/Os and function module) and for the bridge is activated. The IP addresses of the Engineering and Realtime Ethernet (in preparation) interface are excluded and set separately.
- With the default settings, the IP addresses are automatically generated (except for Engineering and onboard Realtime Ethernet). The user only has to make the following settings:
 - Addresses of the sercos drives and sercos I/O
 - When using the sercos III function module: control number (C-0-0031)

Automatically generated IP address and subnet mask

	IP address	Subnet mask
Configuration without sercos III function module (C-0-0031 = 0)		
All sercos III interfaces of the control/bridge	172.31.254.254	255.255.0.0
All sercos III drives and I/Os	172.31.254.<sercos address>	255.255.0.0
Configuration with sercos III function module (C-0-0031 > 0)		
All sercos III interfaces of the control/bridge	172.31.<C-0-0031>.254	255.255.0.0
All sercos III drives and I/Os	172.31.<C-0-0031>.<sercos address>	255.255.0.0

Fig. 5-42: Automatically generated IP address and subnet mask

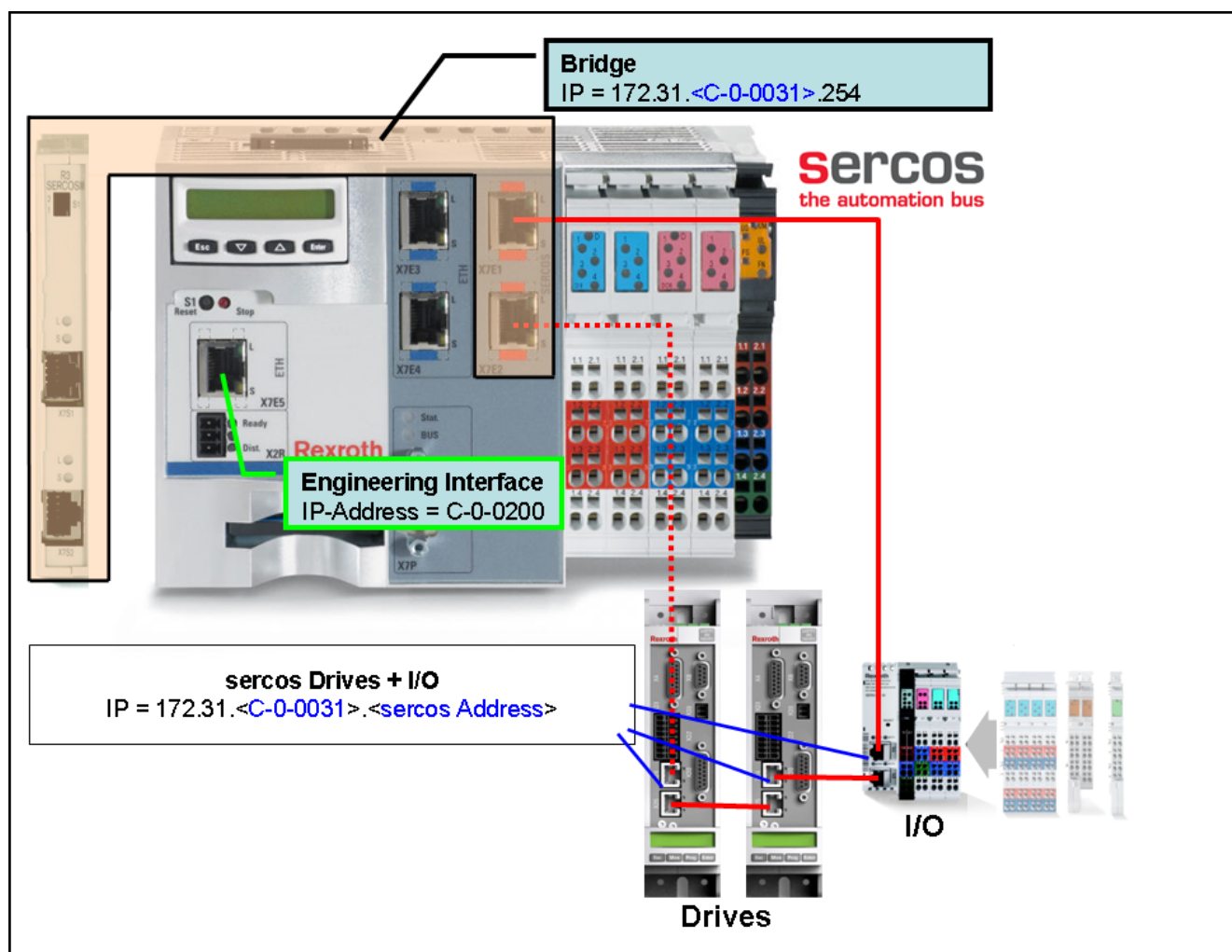


Fig.5-43: IP settings with the default configuration without sercos III function module (C-0-0031=0)

Commissioning and Operation

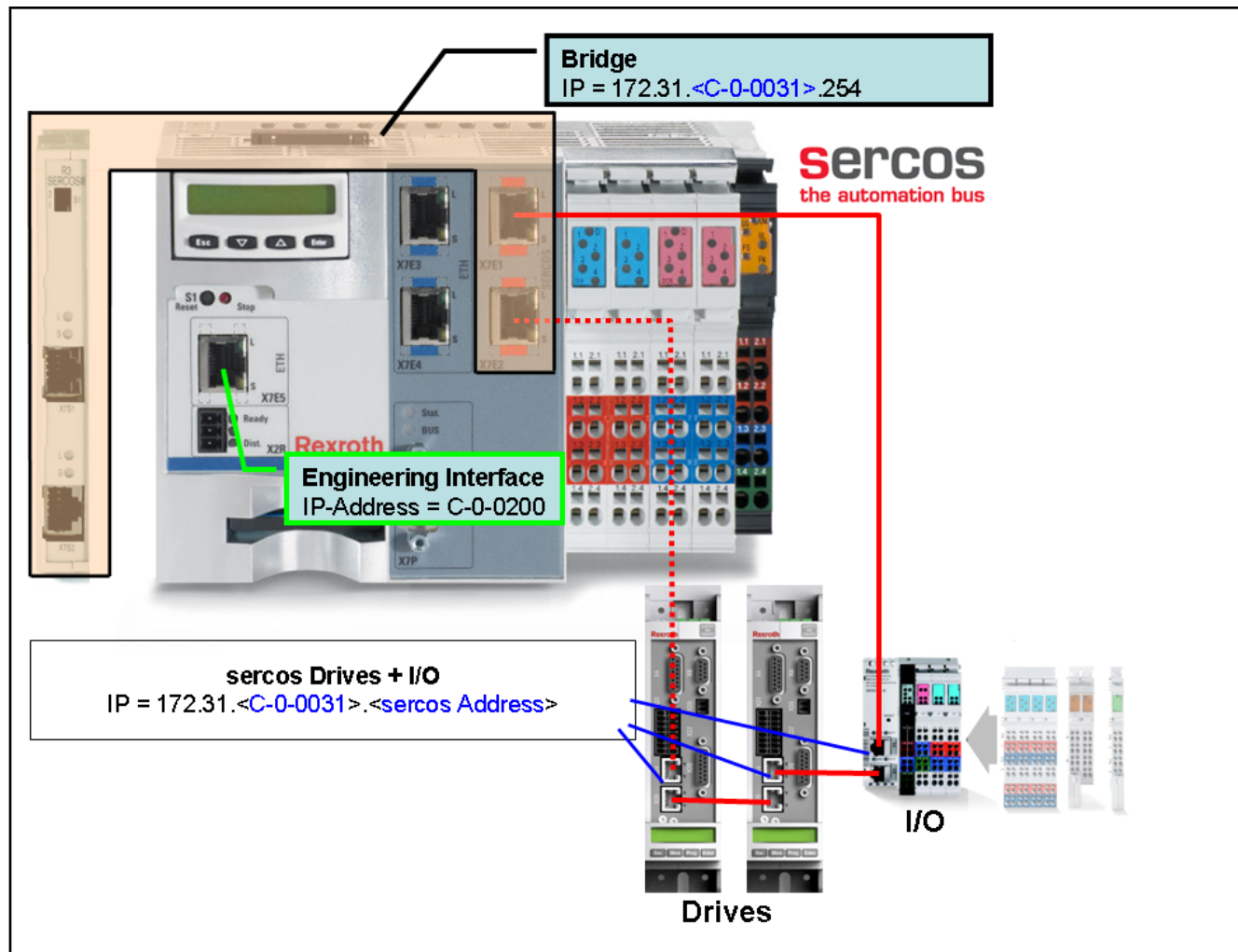


Fig.5-44: IP settings with the default configuration with sercos III function module (C-0-0031 > 0)

Commissioning and Operation

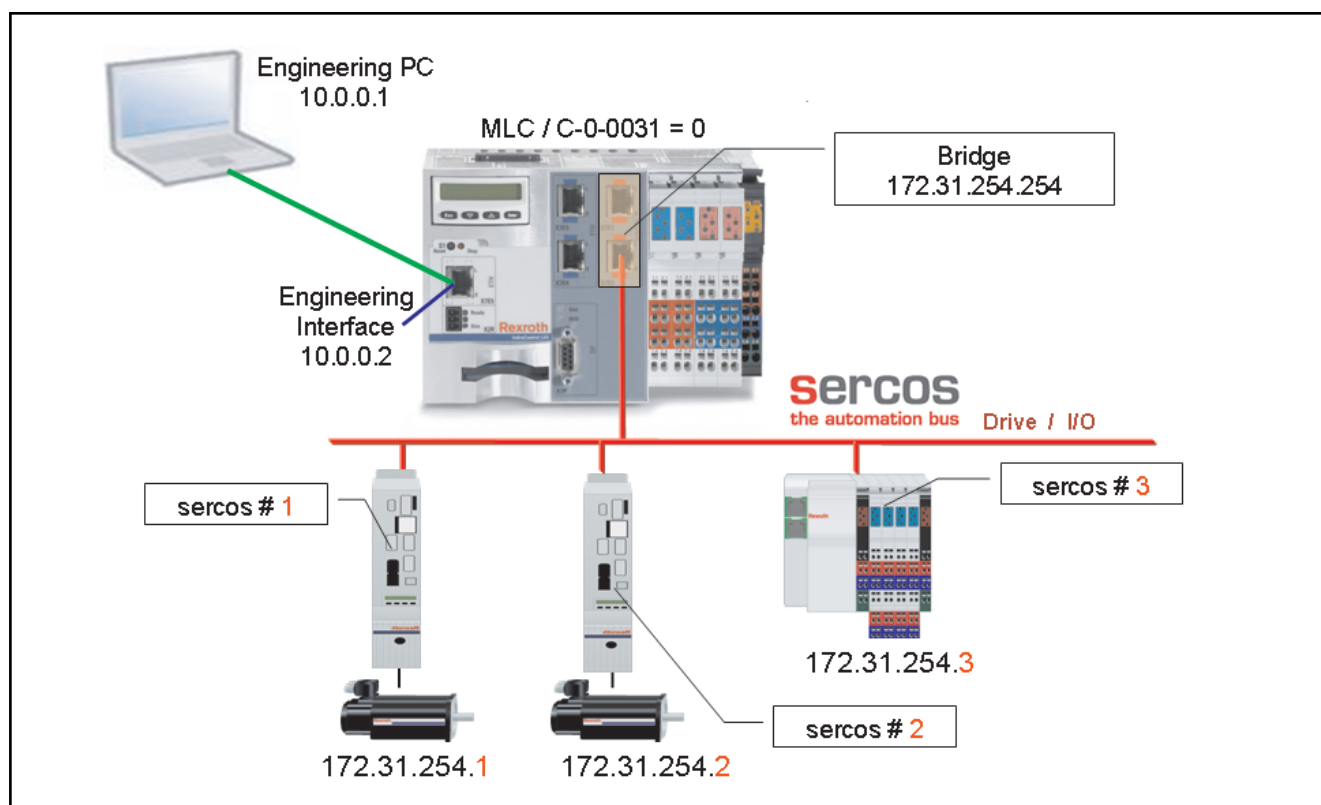


Fig. 5-45: Example: Minimum topology (default configuration)

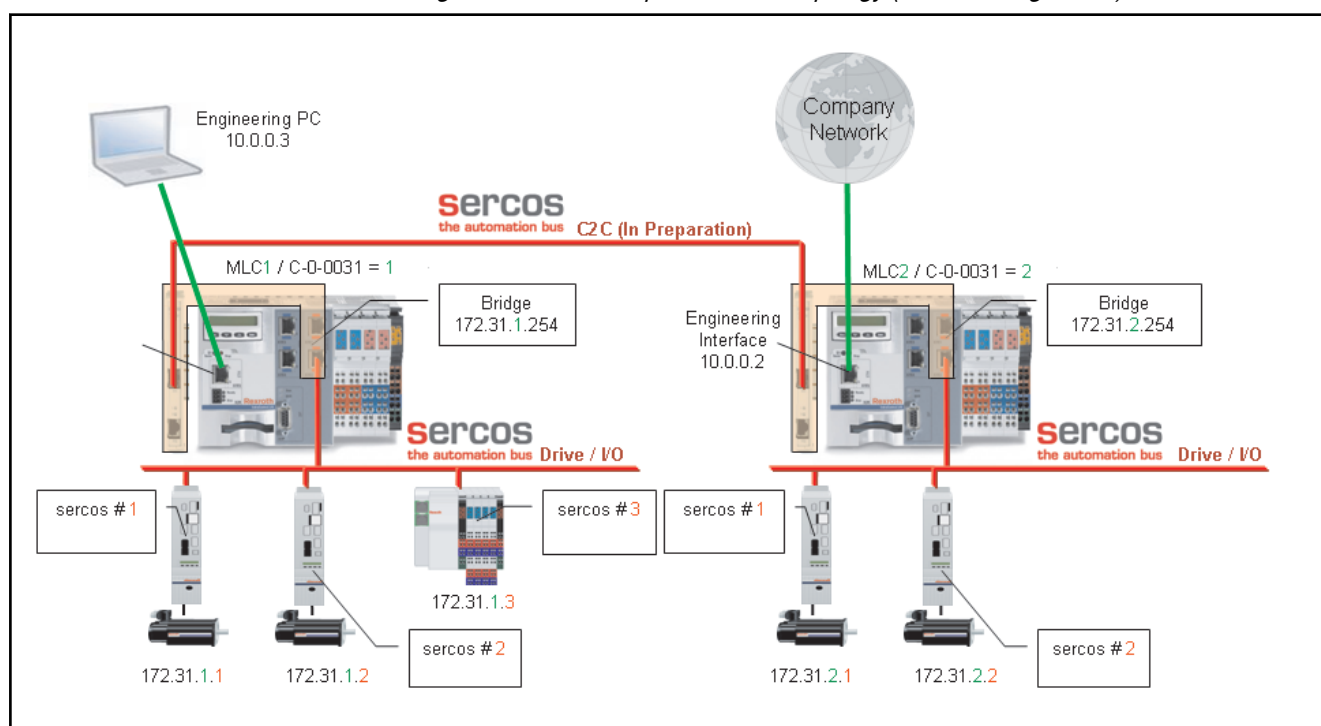


Fig. 5-46: Example: Bridging with sercos III function module (default configuration)

Commissioning and Operation



Via the Engineering PC, even the MLC2 control can be reached via the IP channel of the sercos III C2C connection. In this case, make the following settings:

- Parameterize controls via parameter C-0-0701 as C2C master or slave
- Control address of the respective control is specified via parameter C-0-0031 in the range 1...64
- Set the IP address of the C2C master as default gateway of the C2C slave controls

5.10.4 IndraWorks Commissioning Dialogs

To configure the Ethernet-based interfaces, commissioning dialogs are provided in IndraWorks. For these dialogs, go to **Communication ► Interfaces** in the context menu of the control.

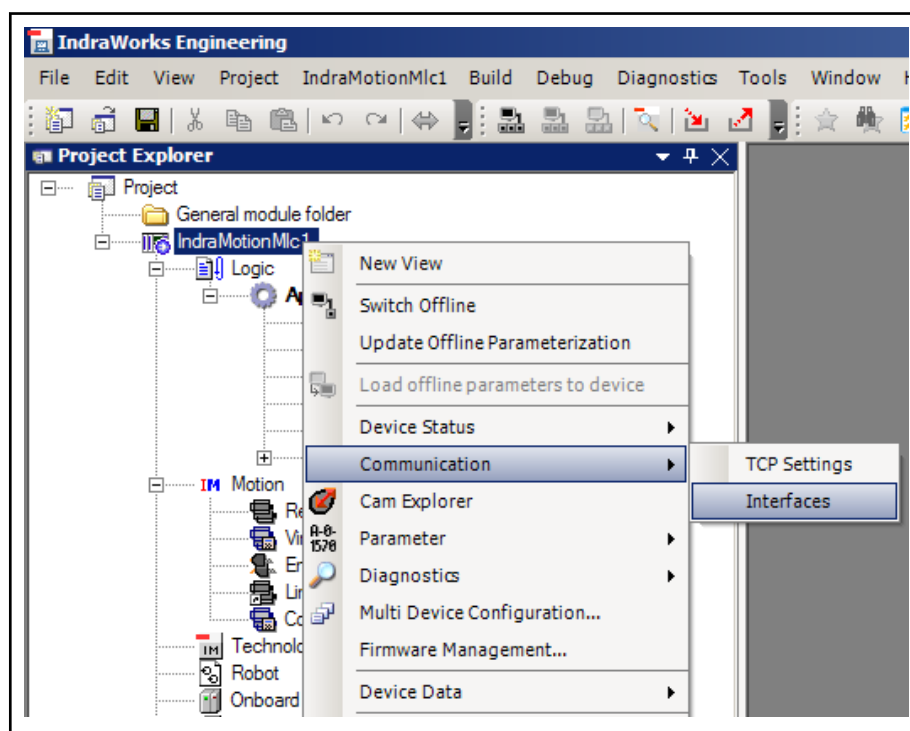


Fig.5-47: Context menu to commission the Ethernet-based interfaces

Enable or disable the automatic address generation and the assignment to the bridge for respective interfaces. The changed settings are only applied after pressing the **Apply command configuration** button. Furthermore, the IP default settings [chapter 5.10.3 "Default Setting" on page 78](#) can be loaded and enabled via the **Load Basic Network Parameters** button.

Commissioning and Operation

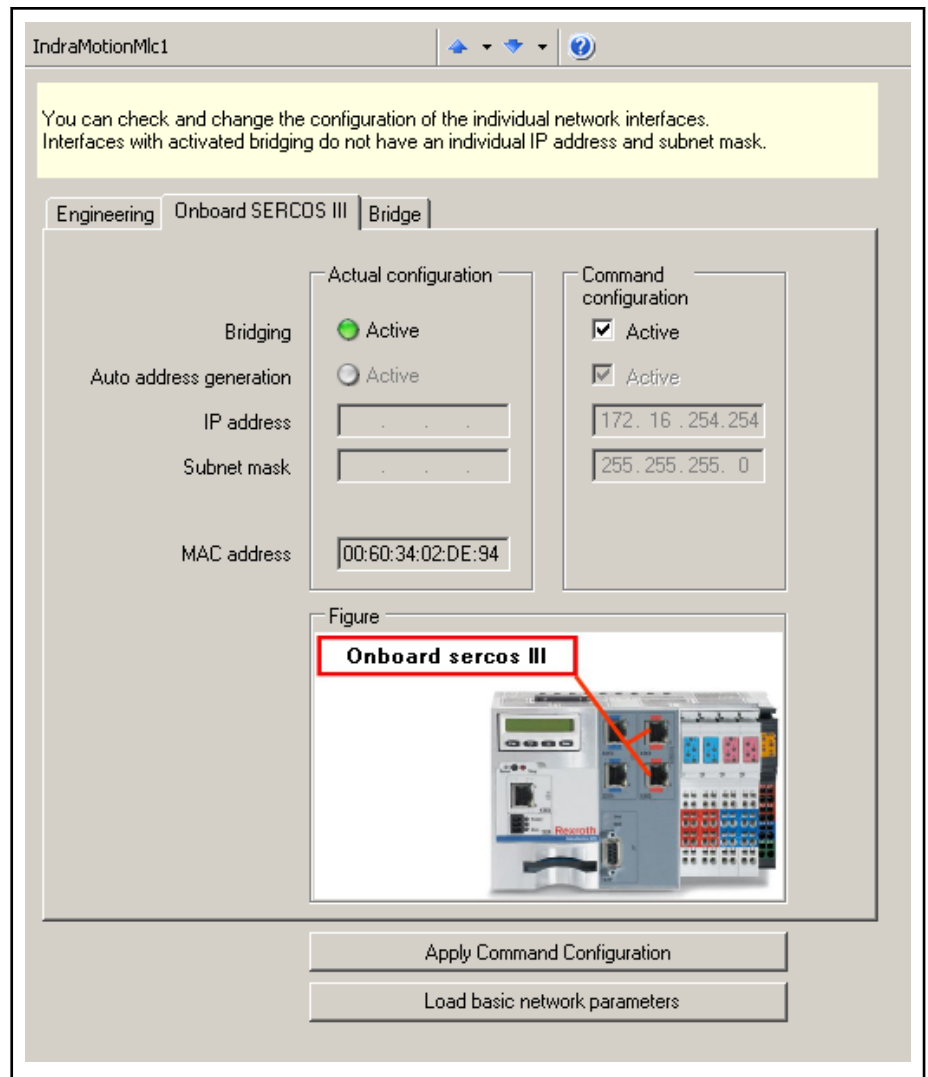


Fig. 5-48: Commissioning dialog to set the IP communication (in this case, bridging and automatic address generation enabled)

In preparation: Special features of the onboard Realtime Ethernet

The automatic address generation and the bridging for the onboard Realtime Ethernet is switched off by default. In this case, set the command configuration of the onboard Realtime Ethernet (e.g. IP address) via additional IndraWorks dialogs. Depending on the control configuration, these dialogs are provided via the PROFINET_IO_Controller, EtherNet_IP_Scanner, Ethernet_Interface... nodes below the control.

Commissioning and Operation

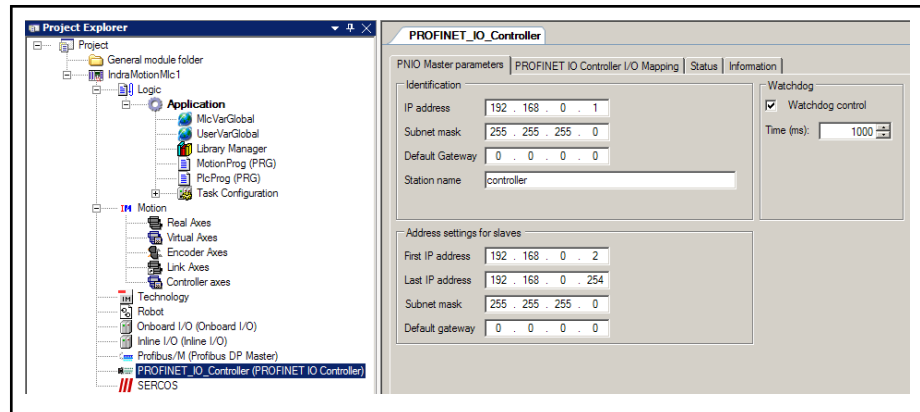


Fig.5-49: Onboard real-time Ethernet PROFINET_IO_Controller dialog



If the **automatic address generation** or the **bridging** is active for the onboard Realtime Ethernet interface, the IP address is automatically assigned to the interface. The address settings in the onboard Realtime Ethernet dialogs are therefore **not applied**.

5.10.5 Automatic IP Address Generation

Depending on the setting in C-0-0234, the IP addresses are assigned automatically (exception: Engineering interface), to assign a unique IP address to each IP device of the automation system. The IP addresses are generated according to the following rules:

Interface		Generated IP address	Generated subnet mask
Onboard sercos III	C-0-0031 = 0	172.16.254.254	255.255.255.0
	C-0-0031 > 0	172.16. <C-0-0031>.254	255.255.255.0
In preparation: Onboard Real-time Ethernet	C-0-0031 = 0	172.17.254.0	255.255.255.0
	C-0-0031 > 0	172.17. <C-0-0031>.254	255.255.255.0
sercos III function module	C2C master (C-0-0701 Bit1 = TRUE)	172.16. 0.254	255.255.255.0
	C2C slave (C-0-0701 Bit1 = FALSE)	172.16.0.<C-0-0031>	255.255.255.0
Bridge	C-0-0031 = 0	172.31.254.254	255.255.0.0
	C-0-0031 > 0	172.31.<C-0-0031>.254	255.255.0.0

Fig.5-50: Automatic IP address generation of the control interfaces



In preparation: The IP address in the IndraWorks dialogs Ethernet-IP (scanner/ adapter), Profinet (controller/device) is not applied if the automatic IP address assignment of the onboard Realtime Ethernet is switched on.

		Onboard sercos III in bridge	Onboard sercos III not in bridge
sercos III drive or I/O address	C-0-0031 = 0	172.31.254.<sercos address>	172.16.254.< sercos address>
	C-0-0031 > 0	172.31.<C-0-0031>.<sercos address>	172.16.<C-0-00031>.<sercos address>

Fig.5-51: Automatic IP address generation of sercos III drives and sercos III I/Os

5.10.6 Configuring the Bridge

With the default settings, all sercos III interfaces (onboard and function module) are part of the **bridge**. Broadcast telegrams thus create data traffic (sercos III broadcast telegrams are excluded from this) in all sercos III networks. This can cause reduced performance in very large sercos III networks (in total more than 500 IP devices). In such cases, individual sercos III interfaces can be removed from the bridge via the parameter **C-0-0232 Bridge, command configuration of assigned interfaces**. If the interface is not part of the bridge, routing is used as an alternative to other interfaces.

Furthermore, the onboard Engineering interface and the Realtime Ethernet (in preparation) interface can be included in the bridge via the parameter **C-0-0232 Bridge, command configuration of assigned interfaces**. Thus, the broadcast telegrams are available across networks in the networks assigned to the bridge.

5.10.7 Involved IP Communication Parameters

IndraWorks commissioning dialogs [chapter 5.10.4 "IndraWorks Commissioning Dialogs" on page 82](#) are available for the typical use cases which are described in the following. It is only necessary to directly describe the respective parameters in special use cases.

The following table shows a brief overview on the relevant IP communication parameters. The following rules apply:

- The IP settings of the Engineering interface, the sercos III interfaces and the bridge are applied with the command **C-0-1023 Command: Apply network settings**. The command specification is defined for the onboard sercos III interface via the parameter C-0-0220 for example. It is applied to the parameter C-0-0222 when the command "Apply IP settings" is started.
- In preparation: The IP settings of the onboard real-time Ethernet are specified in the IndraWorks dialogs (e.g. PROFINET_IO_Controller node). They are applied after a control restart. The command specifications are not explicitly available via parameters
- Via the command **C-0-1024 "Command: Load basic parameter of IP configuration"**, the default values for the sercos III interfaces (onboard and function module) and the bridge are copied to the command configuration parameters and then automatically applied. Furthermore, the default value of the IP address generation is written to the command value parameter (C-0-0235) and then automatically applied.
- The parameter **C-0-0219, Status of network configuration** indicates for the individual interfaces bit by bit whether changes are entered but not yet applied in the IP settings.
- The parameter **C-0-0234, Automatic IP address generation, actual configuration** indicates the current setting of the automatic IP address generation (actual configuration) bit by bit.
- The parameter **C-0-0235 Automatic IP address generation, command configuration** controls the automatic IP address generation bit by bit for all optional network interfaces on the control (command configuration).
- The parameter **C-0-0218 List of existing network interfaces** indicates the network interfaces on the hardware.

Commissioning and Operation

Interface		IP address	Subnet mask	MAC address	Corresponding bit in C-0-0219 C-0-0235 C-0-0234
Engineering	Command configuration	C-0-0200	C-0-0201	C-0-0205	
Onboard sercos III	Actual configuration	C-0-0222	C-0-0223	C-0-0224	Bit 0:
	Command configuration	C-0-0220	C-0-0221		
sercos III function module	Actual configuration	C-0-0227	C-0-0228	C-0-0229	Bit 1
	Command configuration	C-0-0225	C-0-0226		
In preparation: Onboard Realtime Ethernet	Actual config.	C-0-0242	C-0-0243	C-0-0244	Bit 2
	Command configuration	IW dialogs	IW dialogs		
Bridge	Actual configuration	C-0-0236	C-0-0237	C-0-0238	Bit 3
	Command config.	C-0-0230	C-0-0231		

Fig. 5-52: Overview on the most important IP communication settings

5.10.8 Ethernet Ports and Firewall

The control and PC communicate via Ethernet. Several protocols are used for this purpose, depending on the application area. Firewalls, which block the communication on some Ethernet ports, are installed by default on most of the PCs. This is not critical in most use cases. If communication problems between the control and PC occur, the firmware settings have to be checked in any case.

The control communicates with the other communication devices via the following Ethernet ports:

Port	Direction	Purpose	Communication partner
TCP 5000..5003	Incoming	Parameter communication (SIS)	IndraWorks
TCP 5300	Incoming	MLPI	
TCP 20, 21	Outgoing and incoming	Firmware download (FTP)	IndraWorks
TCP 80	Incoming	IMST/web server (HTTP)	Web browser
TCP 23	Incoming	Telnet	
ICMP	Incoming	Ping	IndraWorks
TCP 6091	Outgoing	Robot Control	IndraWorks

Commissioning and Operation

Port	Direction	Purpose	Communication partner
UDP 69	Outgoing	Firmware download, drives (TFTP)	IndraWorks
TCP 35021	Incoming	Parameter communication (SIP)	IndraWorks
TCP 1317	Incoming	PLC	IndraWorks
UDP 1740..1743	Incoming	PLC	IndraWorks

Fig. 5-53: Ethernet ports of the control

The "Direction" column refers to the control.

Incoming: Connection from the PC in the direction of the control. These connections are generally not critical for firewalls.

Outgoing: Connection from the control in the direction of the PC. These connections are generally blocked by a firewall. A firewall present on the PC has to be configured for these connections.

5.10.9 More Notes on the Use of Ethernet-Based Interfaces

To use Ethernet-based interfaces, multiple functionalities are available. The following table provides an overview and refers to further documentation.

Functionality	Description	Related documentation
OPC server	Data exchange of the control with master level or HMI via OPC server	MLD/MPx 06VRS OPC Communication DOK-IM*ML*-OPC*COM*V12-AWxx-EN-P
ModbusTCP	Data exchange of the control with third-party HMIs. The control is the ModbusTCP server.	Rexroth IndraLogic 11VRS Modbus TCP DOK-IL*2G*-MOD*TCP*V12-APxx-EN-P
Network variables	Control-comprehensive data exchange of PLC variables via broadcast and UDP	Rexroth IndraWorks 12VRS IndraLogic 2G PLC Programming System DOK-CONTRL-IL2GPRO*V12-APxx-EN-P
sercos III CrossComm	A control-comprehensive data exchange in the link of up to 64 IndraMotion MLC controls with real-time properties to e.g. exchange master axis positions	Rexroth IndraControl Lxx 12VRS Function Modules DOK-CONTRL-FM*LXX**V12-APxx-EN-P

xx Edition

Fig. 5-54: Further notes on the use of Ethernet-based interfaces

5.11 Axes

The creation of the different axis types is described in [chapter 8 "Axes" on page 227](#).

5.12 Creating a Kinematics



The use of the RoCo functionality (RobotControl) is based on the use of a **S-RAM modules (CFL1.1-Y1)** function module for a permanent data backup (recovery of the RoCo program).

Create this function module in the "Configuration Dialog" of the IndraMotion MLC L25/L45/L65 (alternatively, do this by "dragging" from the **Device Library** ► **FM** and "dropping" to the control in the Project Explorer).

See also /24/, [Required and Supplementing Documentations](#), page 14.

Procedure:

Right-click to open the context menu of the "Robot" node.

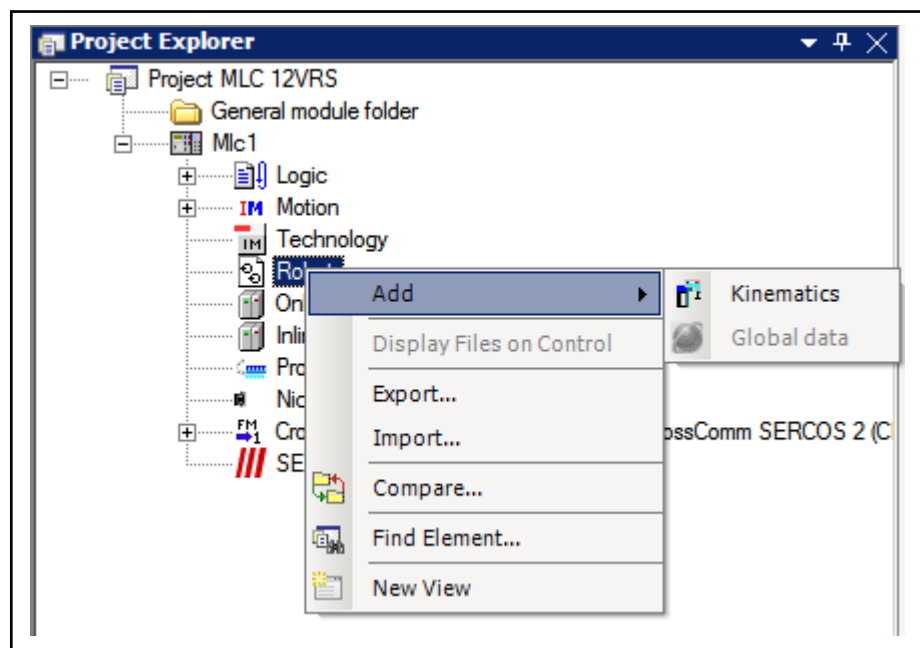


Fig. 5-55: Creating a kinematics

Create the kinematics via the **Add...** menu item.

A wizard, in which all the required specifications have to be made step-by-step, is displayed.

Enter the name of the kinematics in the first step.

Commissioning and Operation

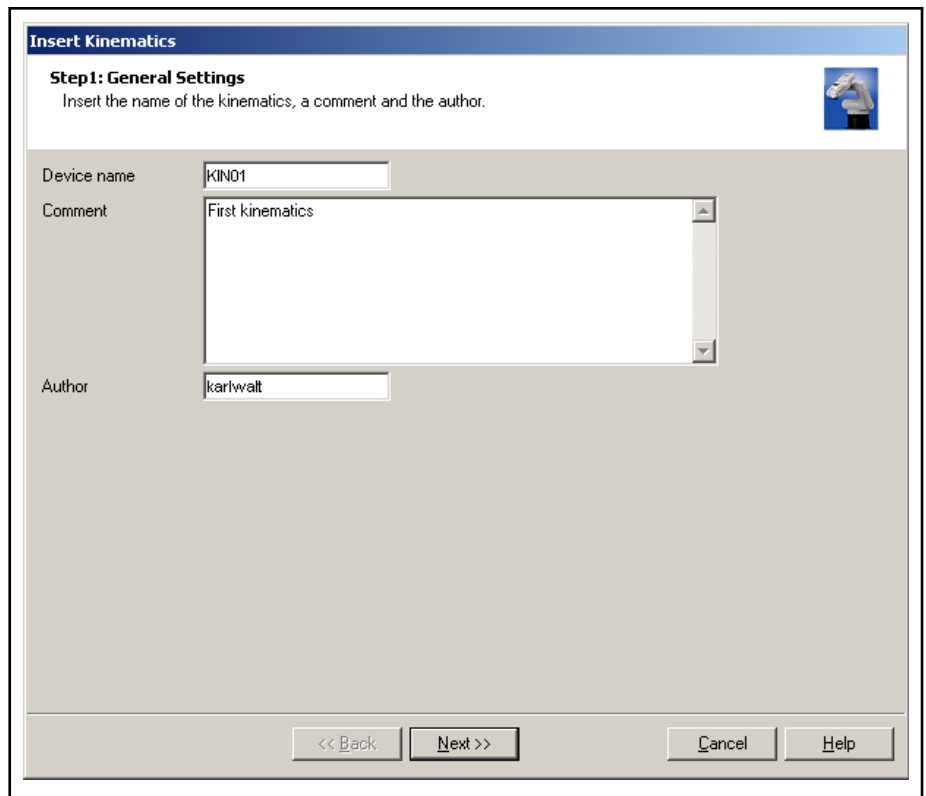


Fig. 5-56: Assigning kinematic name

Select the kinematic type in the second step.

In this example, the **Cartesian** kinematic type is selected.

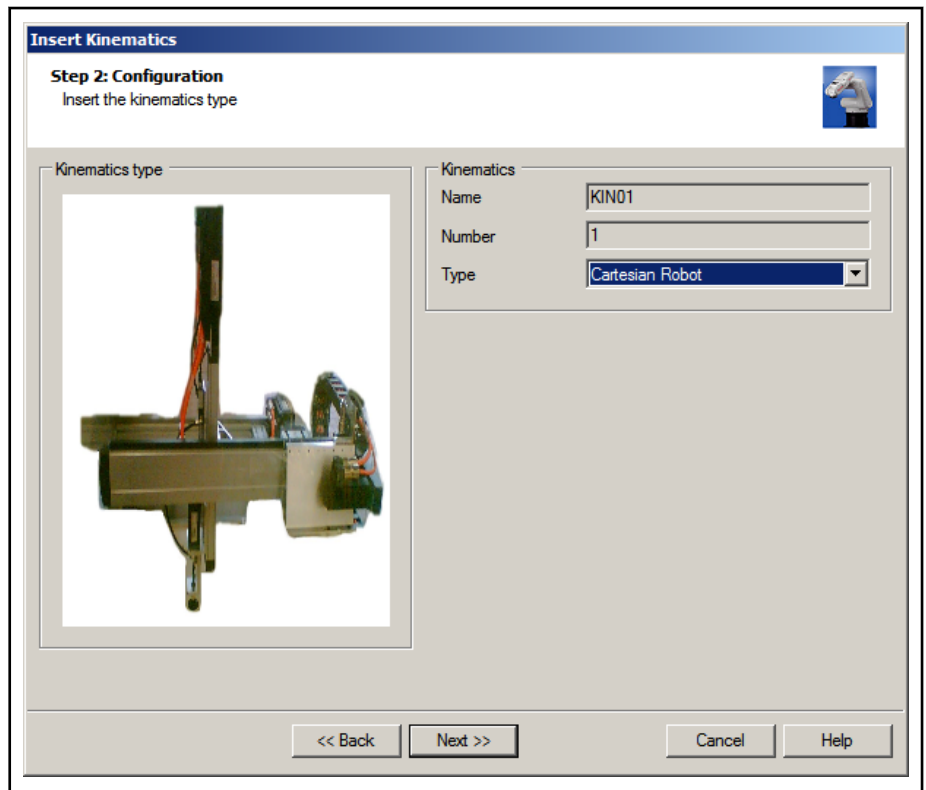


Fig. 5-57: Selecting kinematic type

Commissioning and Operation

Assign axes to the kinematics in the third step.

All (real or virtual) axes physically belonging to the kinematics have to be assigned here.

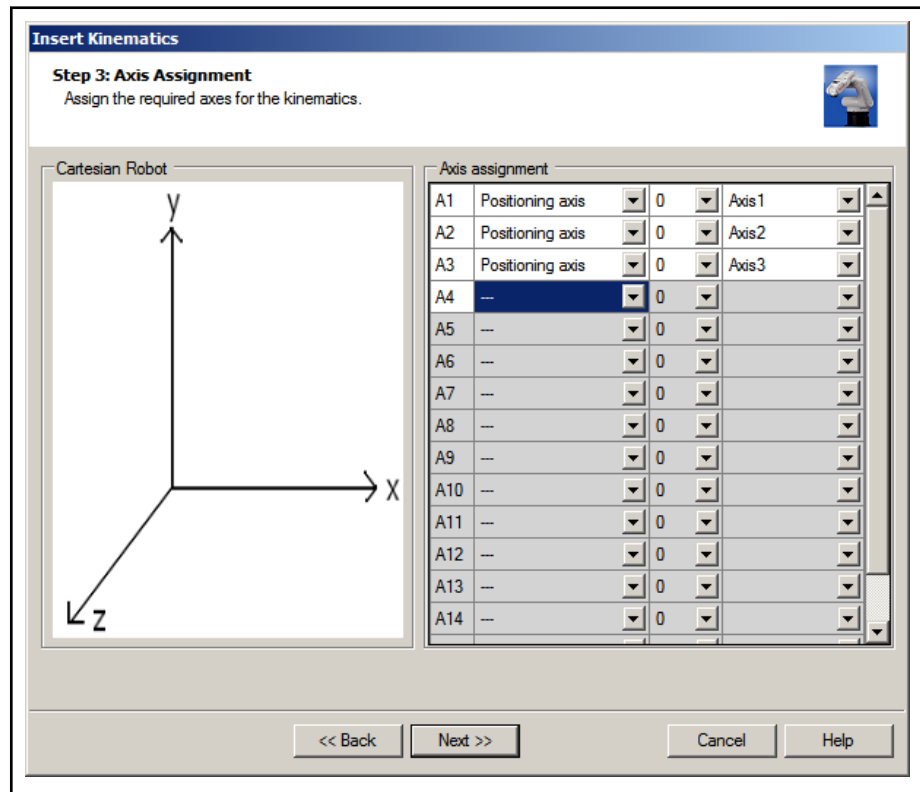


Fig.5-58: Assigning axes belonging to the kinematics

Define the axis coordinate names in the fourth step.

Use the kinematic names defined in the subsequent RoCo programming.

Commissioning and Operation

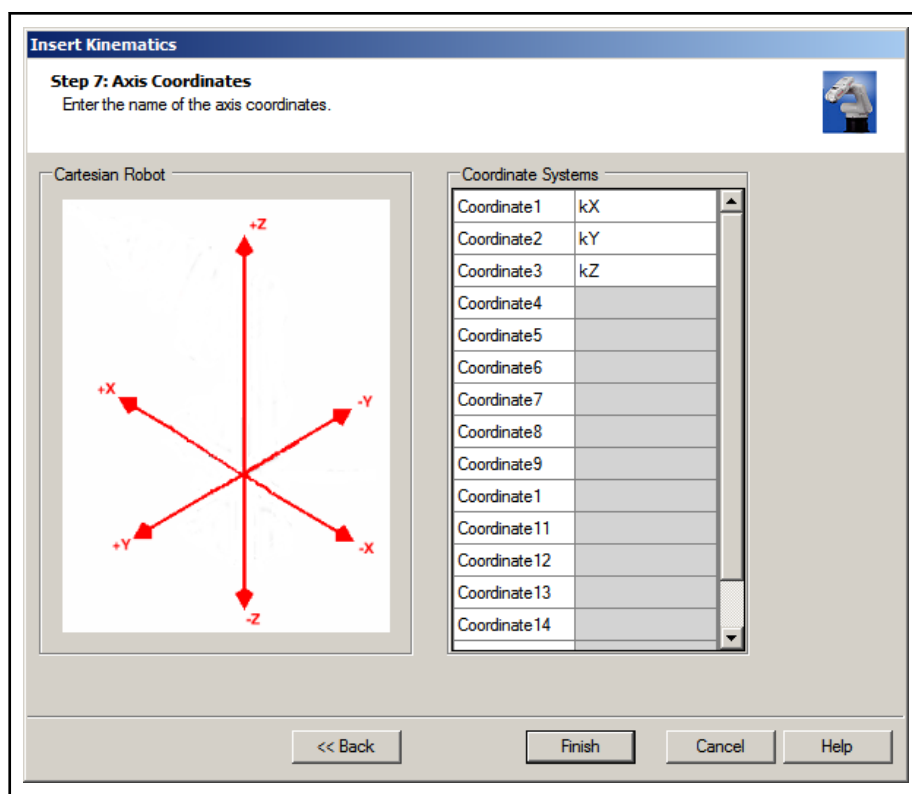


Fig.5-59: Defining axis coordinate names

Click on **Finish** to complete the creation of the kinematics.

The creation of a RoCo program and its programming is described in "Rexroth IndraMotion MLC 12VRS RCL Programming Instruction" (see /31/, [Required and Supplementing Documentations, page 14](#)).

6 Control

6.1 Project Structure

An IndraMotion MLC project is characterized by a control instance (e.g. Mlc1).

The control instance consists of its context menu and several nodes: a Logic, a Motion and a sercos node; Robot, Onboard I/O**, Inline I/O, Profibus/M**, Profinet_IO_Controller folders are also available. Function module folders can be added to the control (e.g. CrossComm sercos II (CFL01.1-Q2) for sercos II cross communication).

** - only IndraMotion MLC L45/L65

The "Logic" node is provided with the "Application" subfolder. This contains:

- MlcVarGlobal
- UserVarGlobal
- Library Manager
- MotionProg (PRG)
- PlcProg (PRG)
- Task Configuration (includes MotionTask and PlcTask)

The "Motion" folder is provided with the subfolders:

- Real Axes
- Virtual Axes
- Encoder axes
- Link Axes
- Controller Axes

in which the individual axis instances can be created.

Context menu and folder differ depending on their operating state (offline or online/offline parameterization). The offline state is described in the section [Commissioning and Operation, page 39](#),.

Control

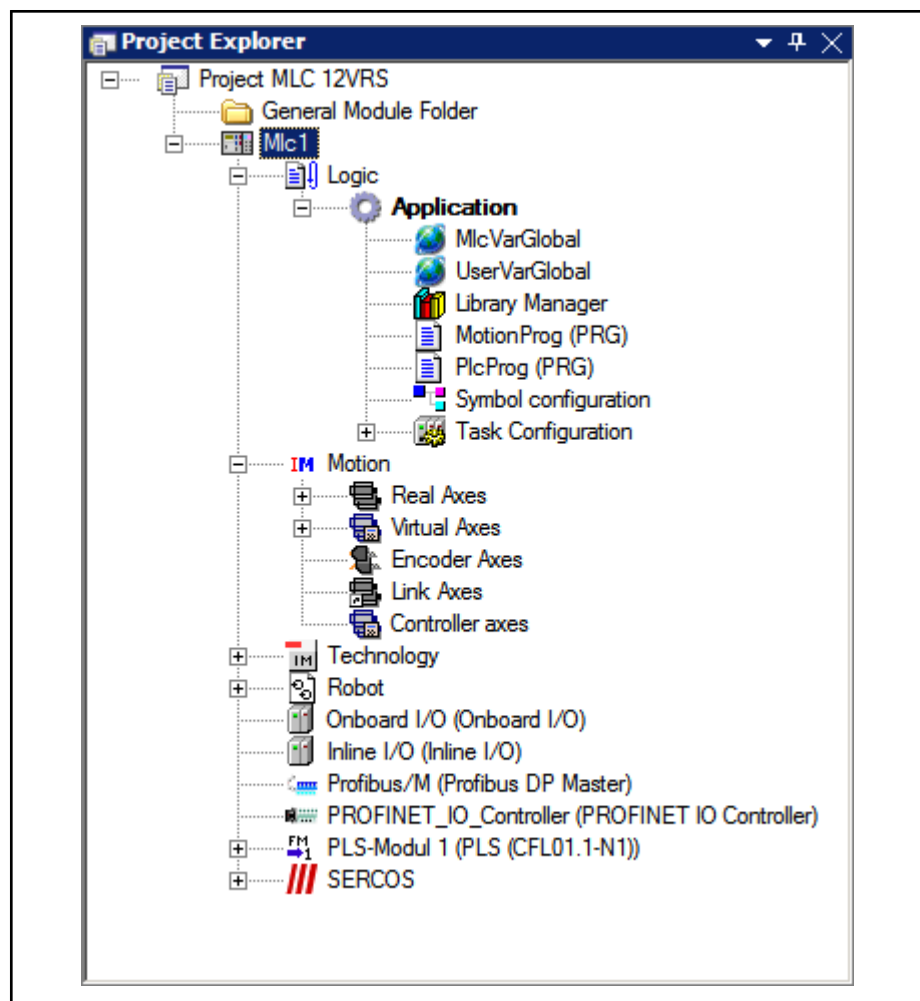


Fig.6-1: Example: Tree structure of an IndraMotion MLC project

6.2 Toolbar of the Control

This toolbar shows the current status and provides direct access to frequently required functions. It is structured as follows:



Fig.6-2: Toolbar of the control

The fields have the following functions (from left to right):

- IP address of the control
- Axis status: This icon is enabled if the axes are rotating. Clicking on the icon opens the "Device Status" dialog (see [Device Status](#))
- Warning: This icon is enabled if a warning is pending. Click on the icon to open the "Error memory" dialog (see [Error/Diagnostic Memory](#))
- Error: This icon is enabled if an error is pending. Click on the icon to open the "Error memory" dialog (see [Error/Diagnostic Memory](#))
- Clear error: Click on this field to execute the "Clear error" command in the control and all connected sercos drives

6.3 Control - Nodes

6.3.1 Node Overview

The nodes below the control result from their configuration:

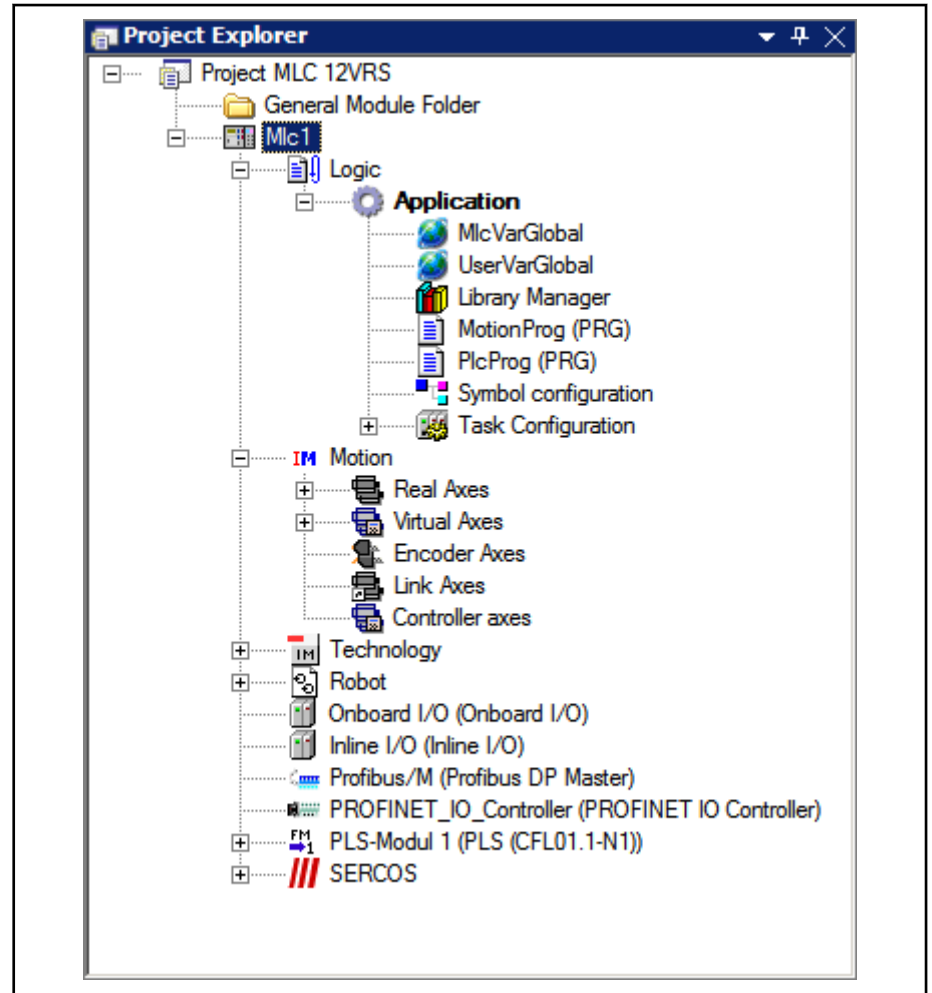


Fig. 6-3: Example: Nodes of the IndraMotion MLC L65 control

- [Logic, page 96](#), transition to PLC programming
- [Motion, page 100](#), basic principle of motion programming
- [Creating a Kinematics, page 88](#), creation of a kinematics
- [Technology, page 104](#), dialogs for PLC technology function blocks
- Options to connect and configure the input/output periphery
 - [Onboard I/Os, page 116](#), 1 byte fast inputs and outputs each*
 - [Inline I/Os, page 116](#), configurable Inline modules
- [Profibus/M, page 116](#), field bus to connect Inline modules, HMI devices, etc.
- [Profinet IO-Controller, page 116](#), field bus to connect Profinet modules
- [Function Modules, page 117](#), here PLS (CFL01.1-N1)
- [sercos, page 117](#), field bus to operate standard drives

* - only IndraMotion MLC L45/L65

Control

6.3.2 Logic

"Logic" Subfolder

The "Logic" node maps the "IndraLogic" PLC programming system.

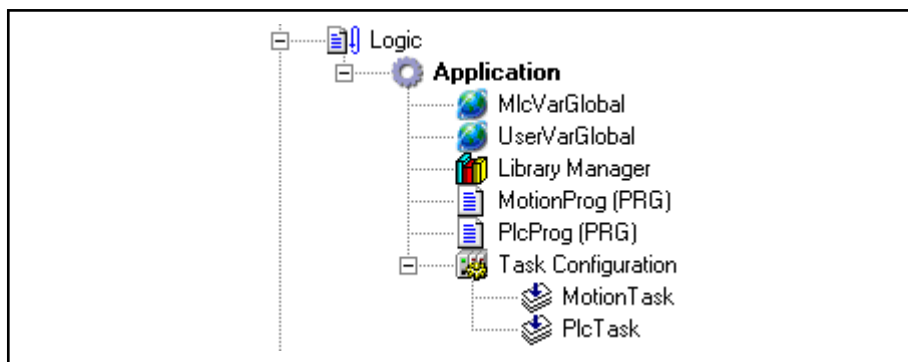


Fig. 6-4: "Logic" node and its subfolders

After the control was transferred to the Project Explorer, two program files are automatically created below the "Application" folder and assigned to the two configured tasks:

- "MotionProg" assigned to the "MotionTask" task (sercos-controlled high-priority task)
- "PlcProg" assigned to the "PlcTask" task (freely running low priority or time-controlled low priority task)

Both programs contain only one dummy network each to ensure that the PLC project can be compiled. The networks can be deleted later. The differences in program use result due to their task assignment.

The "MlcVarGlobal" file contains the AXIS_REF and, for the IndraMotion MLC, the touch probe declarations.

The "UserVarGlobal" file is empty. Users can declare global variables.

The "Library Manager" can be used to add and install libraries (see [Library Management, page 195](#)).

The two preconfigured tasks are integrated in the overall task system of the control (see [Task System, page 201](#)).

"Logic" Context Menu

The context menu of the "Logic" node contains the following menu items:

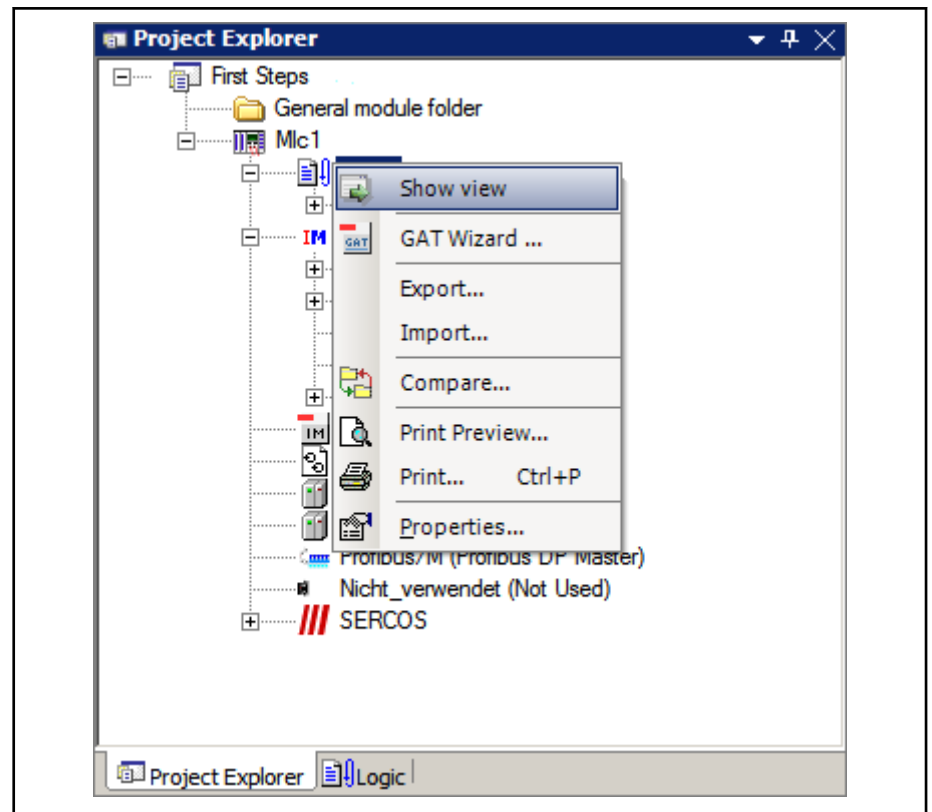


Fig. 6-5: Example: Context menu of the IndraMotion MLC "Logic" node

- **GAT Wizard...** - Wizard for **Generic Application Template** compact functions.
A detailed description on how to use the template can be found in /27/, [Required and Supplementing Documentations, page 14](#).
- **Export...** - Allows the "Logic" elements to be saved in a file
- **Import...** - Allows the "Logic" elements to be imported from an existing export file on the PC. The "Logic" elements to be imported can be selected before the import
- **Compare...** - Compares and merges objects, see [Comparing/Merging, page 176](#)
- **Find element...** - Allows searching for elements below the node
- **Print preview...** - Preview of the "Logic" components selected for printing
- **Print...** - Prints the selected "Logic" components



As the source code of a PLC project is generally very extensive, large volumes of paper can be generated! Use the print option of the individual PLC editors.

- **New View** - Displays node in individual view
- **Properties...** - Displays and modifies properties. The following properties can be modified:
 - Exclude from build
 - External implementation

Control

- Enable system call
- Specify compiler definitions

Application – Toolbar



Fig.6-6: IndraLogic toolbar

The toolbar provides the following functions (from left to right):

- **Build Application** - Compiles the currently active PLC application
- **Login** - Enables the PLC monitoring mode. A connection is established to the PLC of the control. If the PLC program is not current, it is compiled in advance and loaded to the control
- **Log Out** - Ends the monitoring mode. The PLC connection to the control is disconnected
- **Start** - Starts PLC
This command is located on the IndraLogic menu bar and in the "Debug" menu. This command starts the active application in the IndraWorks project. The active application can be set at the "Application" node via the "Set Active Application" context menu
- **Stop** - Stops PLC
This command is located in the IndraLogic toolbar and in the "Debug" menu. This command stops the active application in the IndraWorks project. The active application can be set at the "Application" node via the "Set Active Application" context menu

"Application" Context Menu

With the "Application" subfolder highlighted, right click to open the context menu.

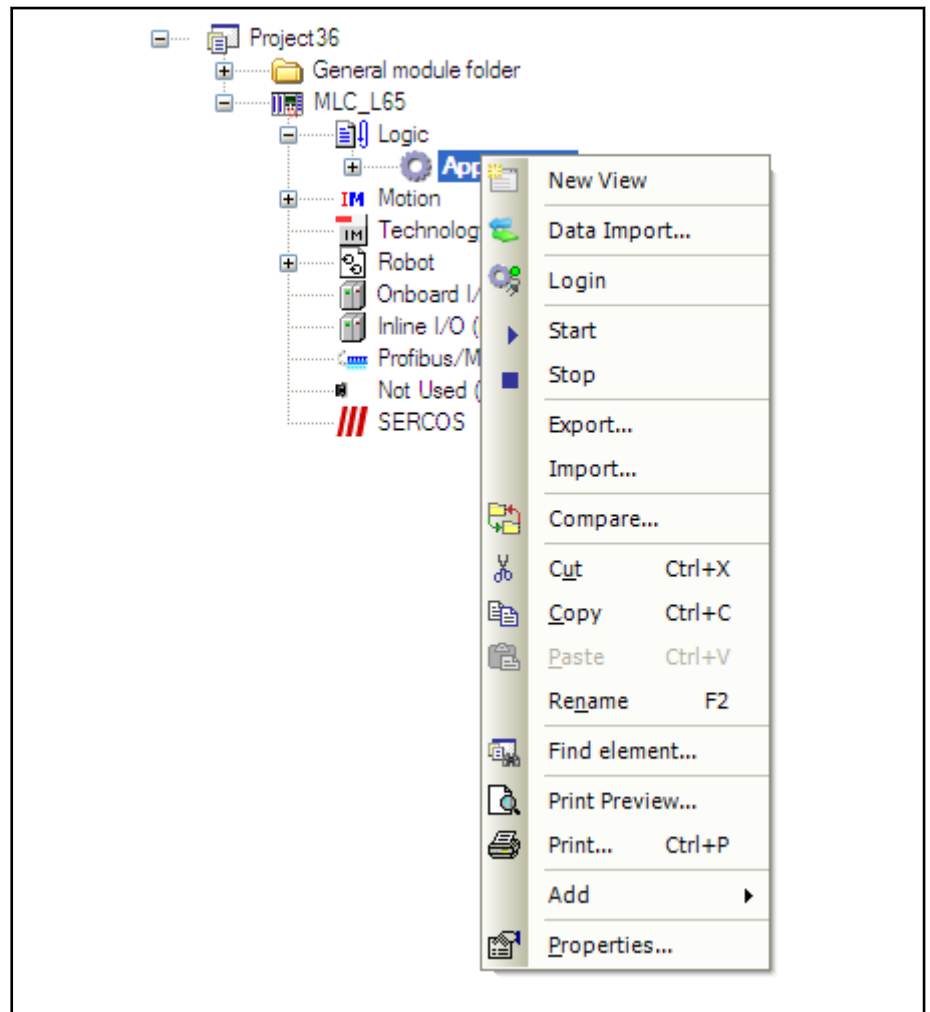


Fig. 6-7: Context menu of the "Application" subfolder

The context menu of the "Application" folder contains the following menu items:

- **Data Import...** - Applies PLC data from other projects
This menu item can also be used to apply the IndraLogic 1-based PLC part and field bus configuration of IndraWorks or IndraLogic projects.
- **Login** - Log in to PLC page



If there are several controls in the project, ensure that the desired application was set to active before using the "Debug" menu when logging in.

- **Start** - Starts PLC
This command is located in the IndraLogic menu bar and in the "Debug" menu. This command starts the active application in the IndraWorks project. The active application can be set at the "Application" node via the "Set Active Application" context menu
- **Stop** - Stops PLC

Control

This command is located in the IndraLogic menu bar and in the "Debug" menu. This command stops the active application in the IndraWorks project. The active application can be set at the "Application" node via the "Set Active Application" context menu

- **Export...** - Allows the "Application" elements to be saved in a file
- **Import...** - Allows "Application" elements to be imported from an existing export file on the PC. The "Application" elements to be imported can be selected before the import
- **Compare...** - Compares and merges objects, see [Comparing/Merging, page 176](#)
- **Rename** - Renames the node
- **Find element...** - Allows searching for elements below the node
- **Print preview...** - Preview of the "Application" components selected for printing
- **Print...** - Prints the "Application" components selected
- **Add** - Adds PLC-specific objects
- **New View** - Displays node in individual view
- **Properties...** - Displays and modifies properties. The following properties can be modified:
 - Node name
 - Exclusion of build application
 - External implementation
 - Enable system call
 - Specify compiler definitions

6.3.3 Motion

General Information

In **offline mode**, different types of axes can be created in the "Motion" node:

- Real axes based on IndraDrive, SercosDrive, EcoDriveCS or Hydraulic-Drive devices
- Virtual axes (can be used as virtual master axes)
- Encoder axes (real master axes). The encoder uses the plug-in card in the drive electronics of a real axis drive
- Link axes transferring axis information for the cross communication via the CrossComm sercos II (CFL01.1-Q2) function module.
- Controller axes are used to implement the controller of an axis into the PLC and to command this axis like all other axes

For each axis type, a folder is provided in which the respective axis instances can be configured.

At first, the folders are empty.

A real axis is created via **drag&drop** or **copy&paste** of the respective library element (see also [Creating a Real Axis, page 234](#)).

A virtual axis is created by using the "Add" context menu item (see also [Creating a Virtual Axis \(Virtual Master Axis\), page 243](#)).

An encoder axis is created using the "Add" context menu item (see also [Creating an Encoder Axis \(Real Master Axis\), page 262](#)).

A link axis is also created using the "Add" context menu item (see also [Creating a Link Axis, page 257](#)).

A controller axis is also created using the "Add" context menu item (see also [Creating a Controller Axis, page 248](#)).

Each axis of an axis type can be configured using the corresponding dialogs:

- Real axis - [Real Axis – Dialogs, page 238](#)
- Virtual axis - [Virtual Axis - Dialogs, page 246](#)
- Encoder axis - [Encoder Axis - Dialogs, page 266](#)
- Link axis - [Link Axis - Dialogs, page 260](#)
- Controller axis - [Controller Axis - Dialogs, page 253](#)

In **online mode**, the operating state of the "Motion" calculation is displayed at the "Motion" node. Currently, it is directly coupled to the respective sercos phase. The following states are possible:

- **"Initialized"**: Download mode: Motion calculation not active, parameterization only limited (respective sercos phase "P0")
- **"Stop"**: Parameterization mode: Motion calculation not active, parameterization possible (respective sercos phase "P2")
- **"Checks"**: Parameterization verification active. At missing parameterization, the Motion calculation can possibly not be started (respective sercos phase "P3")
- **"Running"**: Operating mode: Motion calculation active (respective sercos phase "P4")

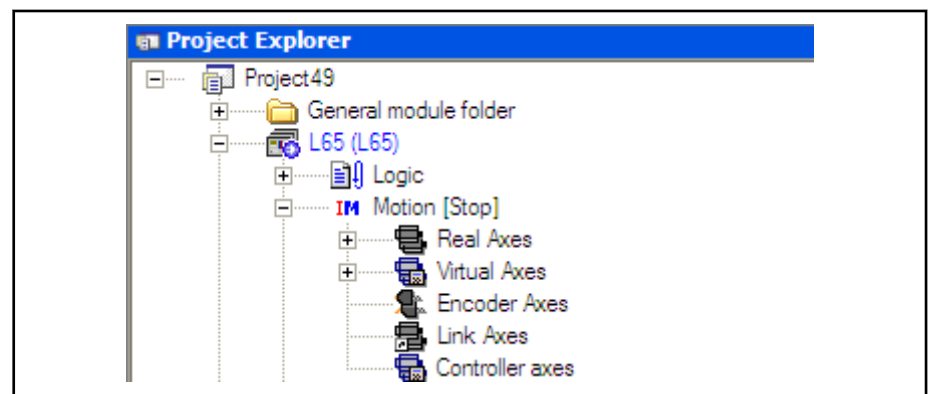


Fig. 6-8: "Motion" node in online mode:

Each axis creates its own environment:

- For the contents of the context menus of the different axis types, refer to
 - [Real Axis – Context Menu, page 241](#)
 - [Virtual axis - Context menu, page 246](#)
 - [Encoder axis - Context menu, page 267](#)
 - [Like Axis – Context Menu, page 260](#)
 - [Controller Axis - Context Menu, page 254](#)

Access the individual software tools via the context menus and dialogs.

"Motion" Context Menu

Overview on Menu Items

The context menu includes the following menu items:

Control

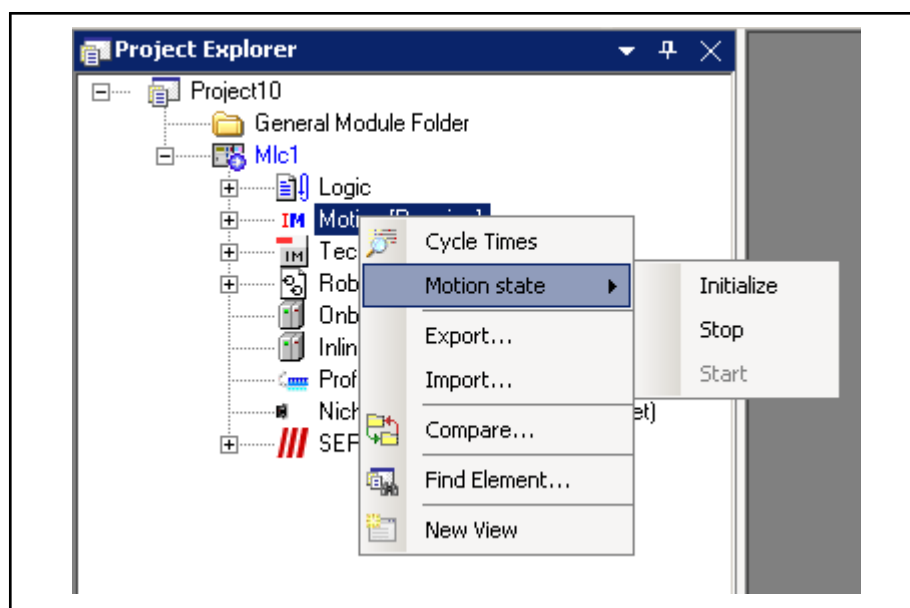


Fig.6-9: Context menu of the "Motion" node

Cycle Times

To support the task management (see also [Task System](#) , page 201), this dialog displays the settings of the Motion cycle time and the sercos cycle time for the Motion calculation sensitivity and for the current load values of the Motion.

This dialog is available as soon as the control is switched to online mode. The settings can be changed in the parameterization mode (P2).

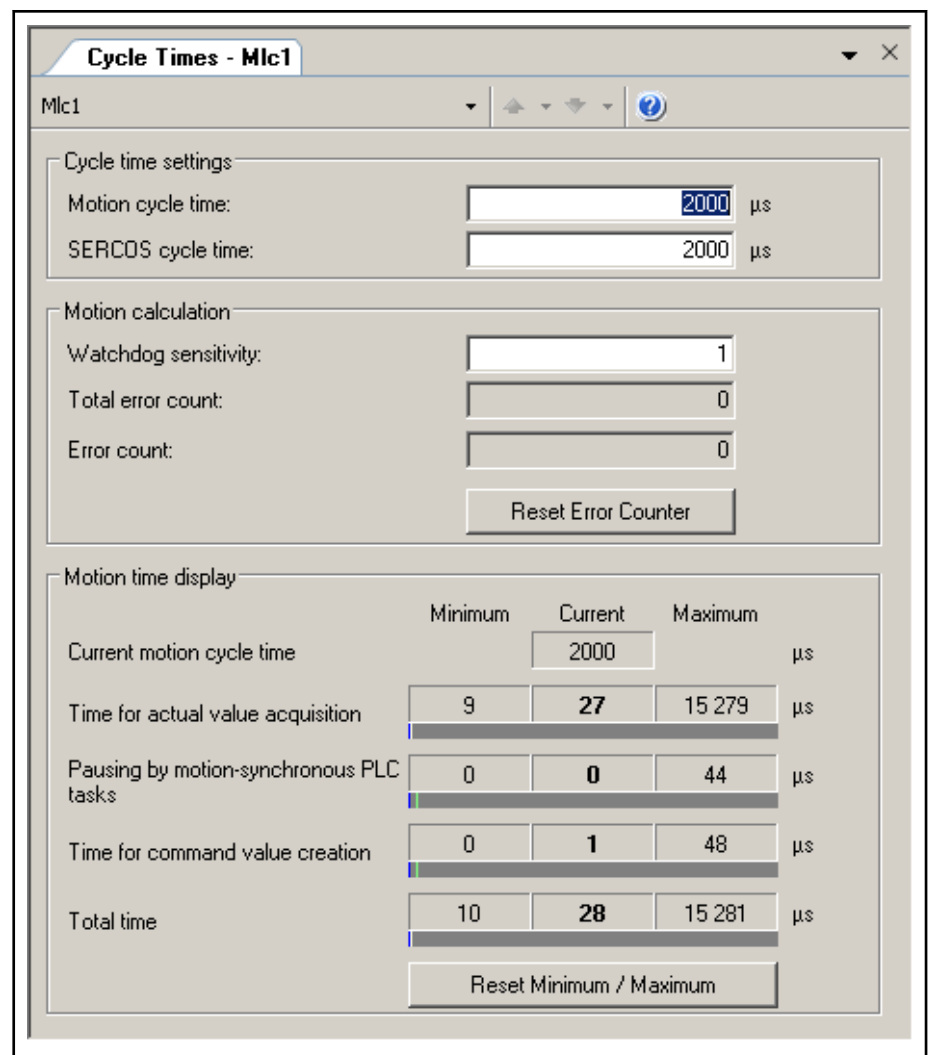


Fig. 6-10: Graphical evaluation of the time history of the task system

The Motion cycle time and the sercos cycle time can be set in "Cycle time settings". The Motion cycle time indicates the intervals the Motion control provides new command values for the connected periphery. The sercos cycle time indicates the intervals new data is available in the sercos ring. The sercos cycle time is only active in the operating mode (BB). The Motion cycle time has to be equal to or a multiple of the sercos cycle time. The Motion cycle time as well as the sercos cycle time are initially set to 2,000 µs.

The sensitivity of the Motion watchdog is set in "Sensitivity of the Motion calculation". This describes the number of successive cycles a timeout is accepted. Additionally, the absolute number of Motion errors and the number of successive Motion errors are displayed. Both the counters can be reset to "0" via the "Reset Error Counter" button.

The actual value of the Motion cycle time and the current values as well as the minima and maxima of the Motion times are displayed in the "Motion time display". The actual value of the Motion cycle time can deviate from the set command value if it is smaller than one millisecond. In this case, the set value becomes only active in the operating mode (BB). The Motion time is divided into the actual value acquisition, the interruption by Motion-synchronous PLC tasks, the command value generation and the total time. The total time consists of the other three Motion times. The minimum and maximum Motion times can be reset to "0" via the "Reset Minimum/Maximum" button.

Control



If many axes are used, corresponding error messages (cycle timeout) can occur when switching from the "parameterization mode" (P2) to the "operating mode" (BB). This happens since the Motion calculation is not able to operate so many axes in such a short time via the sercos ring. In this case, the cycle time has to be increased gradually (2 ms, 4 ms, 8 ms.... 40 ms).

Another reason for the increase of this value can be watchdog errors of the PLC, i.e. the PLC cannot process the time-controlled tasks in the available period of time. These watchdog errors can be activated separately for each task.

Parameters involved

- C-0-0400, Motion, cycle time, command value
- C-0-0503, Master communication, cycle time
- C-0-0410, Motion, cycle time (Tcyc) actual value
- C-0-0412, List of current time slot measurements
- C-0-0413, List of maximum time slot measurements
- C-0-0414, List of minimum time slot measurements
- C-0-0420, Motion, watchdog sensitivity
- C-0-0421, Motion, absolute error counter
- C-0-0422, Motion, number of successive errors
- C-0-1050, Command: Resetting lists C-0-0413 and C-0-0414

Motion State

To control the operating state of the motion calculation, the following commands are provided via the context menu of the "Motion" node:

- **"Initialize"**: Switching to "Initialized". The Motion calculation is stopped. At the same time, the sercos phase is switched to "P0"
- **"Stop"**: Switching to **STOP**. The Motion calculation is stopped. At the same time, the sercos phase is switched to "P2"
- **"Start"**: Switching to "Running". A parameter check is performed first. The Motion calculation is started. At the same time, the sercos phase is switched to "P4"

6.3.4 Technology

Introduction

Supported technology functions of the PLC program of the control are automatically visualized below the "Technology" node. Supported technology functions are organized in groups. Which groups and supported functions exist, can be determined via the "Properties" dialog. For more information, refer to [chapter "Context Menu" on page 105](#).

If an instance of a supported technology function is available in the PLC program, it is displayed below the "Technology" node after a successful program compilation. Instances are displayed in their respective group folder. The dialogs (visualizations) belonging to an instance are located below an instance.

Dialogs of an instance are only available if the PLC application is logged in. Otherwise, the dialogs are grayed out and not available.

Example:

Instance of the technology function "MB_TensionControlLoadCellType01"

This technology function belongs to the "Web tension" group. The instance is called "fbTensionControl" and is provided with the dialogs "Control cycle" and "Limit".

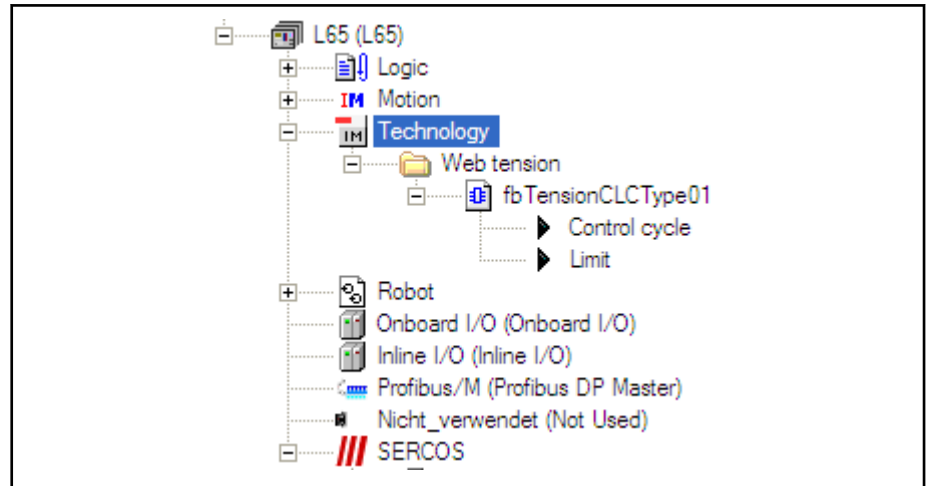


Fig. 6-11: "Technology" nodes with technology instances and dialogs

Tooltip

An instance in the project tree provides a tooltip and additional information on the technology function and the exact storage location in the PLC program. Therefore, the mouse pointer has to be moved via the instance.

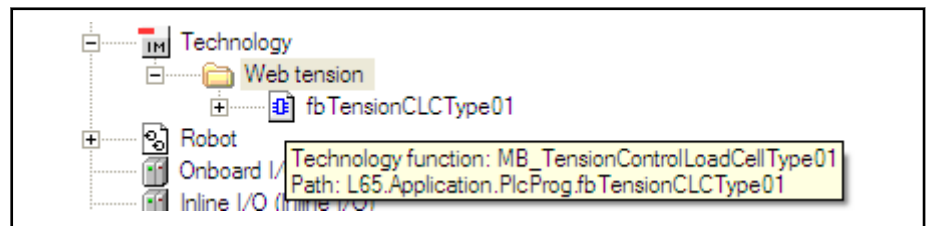


Fig. 6-12: More information on a technology instance

For more information on the technology details, see [chapter "Dialogs" on page 107](#).

Help connection

The "Technology" node is connected to a help. It can be opened with **F1** if the "Technology" node or a subordinate element was selected. Which help is displayed depends on the node currently selected:

- "Technology" node
The general help on the "Technology" node is opened
- Group, instance or dialog nodes
The special help on the respective technology group is opened

Context Menu

The context menu of the "Technology" node is provided with the following functions:

Control

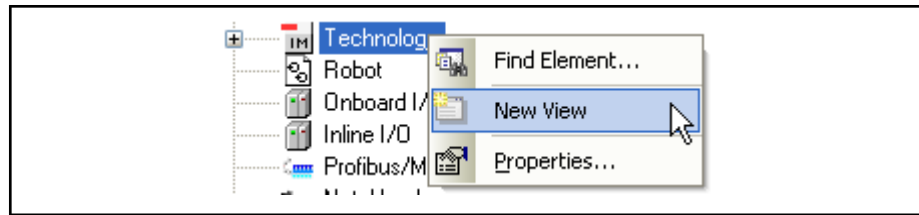


Fig.6-13: Context menu of the "Technology" node

- **New View**
This function displays the "Technology" node in a new Project Explorer window
- **Properties...**
The "Properties" dialog displays the groups supported by the "Technology" node and their respective technology functions. Additionally, the available technology dialogs are listed for each technology function (see [fig. 6-14 "Properties of the Technology node" on page 106](#))
- **Find element...**
This function can be used to search for an instance or dialog for example. The function is only available if the "Technology" node is not empty.

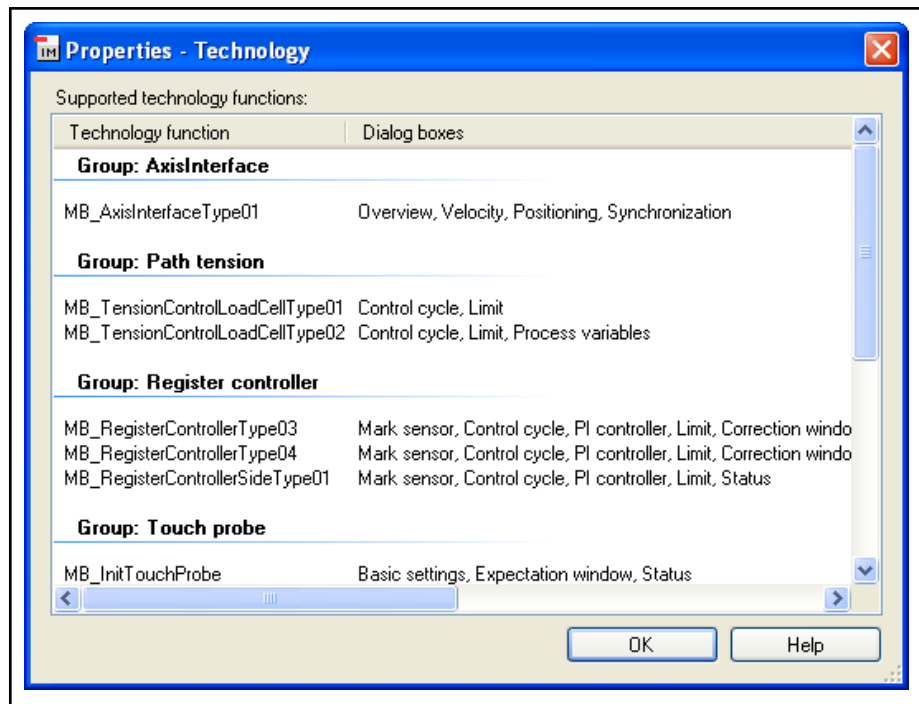


Fig.6-14: Properties of the "Technology" node

"Options" Menu

An "Options" menu is available for the "Technology" node. This can be accessed via **Tools ► Options ► Technology**.

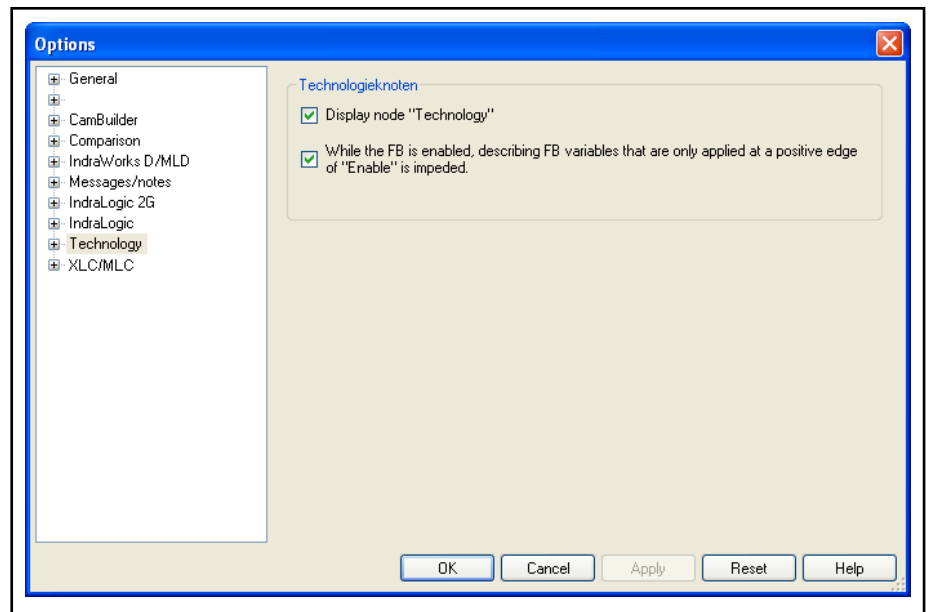


Fig. 6-15: "Technology" options menu

The settings of the "Options" menu affect all "Technology" nodes of the project:

- Displaying the "Technology" node
This option can enable and disable the "Technology" node in the device project.
- While the FB is enabled...
Some input variables of technology functions are read in when a technology function is enabled (e.g. at a positive edge at "Enable"). The technology dialogs impede the entry of new values for these variables while the technology function is active. A note dialog is displayed instead after the entry.

This behavior can be deactivated and activated again via the option described in the menu dialog. If the function is disabled, writing on such input variables is always allowed. If the function is enabled, a note dialog is displayed when entering a new value for a respective variable. For more information, also refer to [chapter "Dialogs" on page 107](#)

Dialogs

General Information

Availability

If an instance of a supported technology function is available after the compilation of the PLC program, it is displayed in the respective folder. One or several dialogs visualizing the technology function belong to the instance. To open a dialog, the respective node in the Project Explorer has to be executed (click on Enter or double-click).



Technology dialogs are only available if the PLC application is logged in. If the application is disconnected while a dialog is open, the dialog is deactivated.

Control

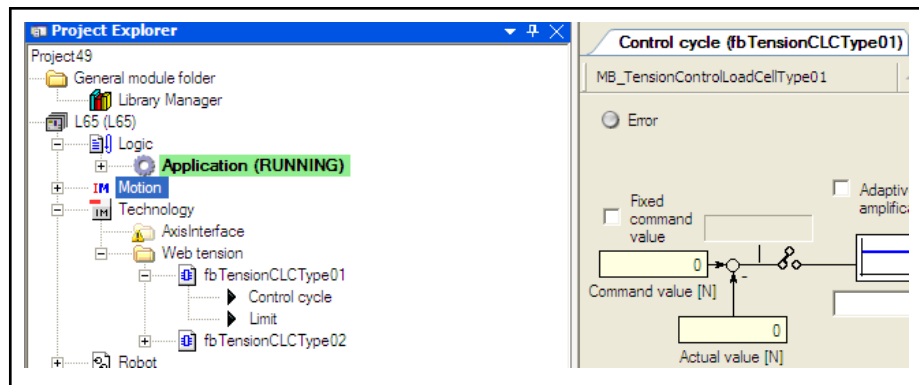


Fig. 6-16: Dialog "Control loop" of the technology instance "fbTensionCLC-Type01"

Navigation

To switch between the dialogs, different functions can be used.

1. Open the dialog via the Project Explorer.
2. Change between the dialogs via navigation arrows or the navigation menu in an open dialog. These buttons allow direct jumping to another dialog of the same technology instance.



Fig. 6-17: Navigation arrows and navigation menu of the "Control loop" dialog.

3. Switch the dialog via hyperlinks in the open dialog. These hyperlinks allow direct jumping to another dialog of the same technology instance.

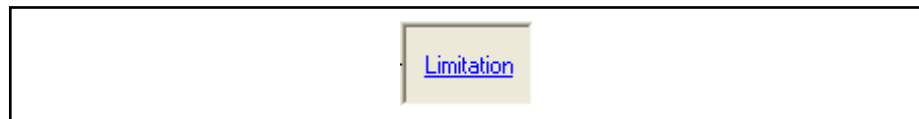


Fig. 6-18: Hyperlink to the "Limitation" dialog

To find out which dialog in the Project Explorer belongs to the currently open dialog, a function in the "Standard" toolbar can be used:

Dialog contents

In the technology dialogs, process-relevant information is displayed. This can include:

- PLC variables
- sercos parameter

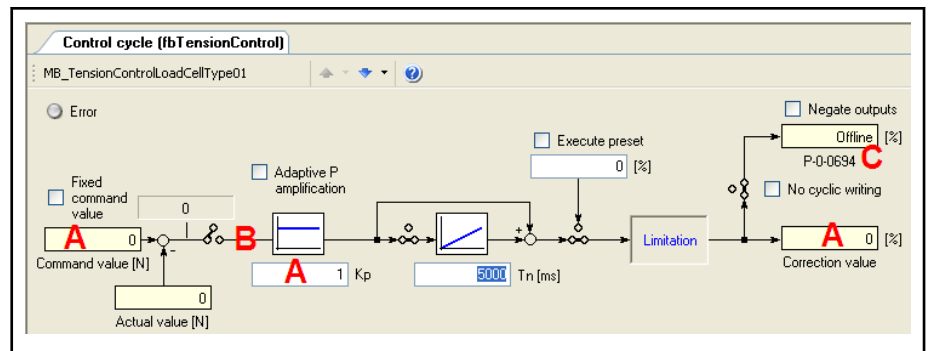


Fig. 6-19: Exemplary contents in the "Control loop" dialog of the technology function "MB_TensionControlLoadCellType01"

PLC variables

PLC variables can be write-protected (see 'A' in fig. 6-19 "Exemplary contents in the Control loop dialog of the technology function MB_TensionControlLoadCellType01" on page 109). A "Properties" dialog with detailed information on this PLC variable can be opened via the context menu of the respective dialog elements (e.g. text box). Therefore, the context menu (right-click) of the dialog element has to be opened.

If a dialog element of a PLC variable provides an input option (e.g. text box), this dialog element might be blocked. Some input variables are only applied by the technology function if it is enabled (e.g. at a positive edge at "Enable"). If the technology function is active, writing on the input is blocked with the following note.

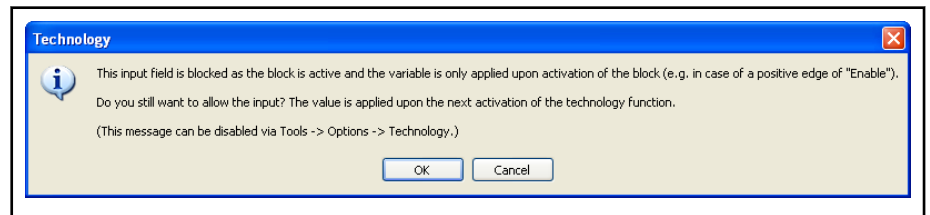


Fig. 6-20: Note at blocked variable of a technology function

With OK, the value is still written but does not influence the technology function. To activate and deactivate this note, see chapter "Options" Menu" on page 106.

PLC variables can also be visualized by signal flow paths (see 'B' in fig. 6-19 "Exemplary contents in the Control loop dialog of the technology function MB_TensionControlLoadCellType01" on page 109). Transmission elements in these paths (e.g. interconnect points, branches, lines or arrows) are shown, opened or deviated according to the state of one or several variables.

sercos parameter

In technology dialogs, sercos parameters are also shown (see 'C' in fig. 6-19 "Exemplary contents in the Control loop dialog of the technology function MB_TensionControlLoadCellType01" on page 109). Therefore, two conditions have to be fulfilled:

1. The device has to be "online". That means there has to be a connection to the Motion system. If this is not the case, the dialog element shows "offline".
2. A respective axis has to be assigned to the technology instance. The example above shows the parameter "P-0-0694" of the "slave" axis. If no "slave" axis was assigned at a cyclic call to the technology instance, the dialog element does not display any content.

The parameter editor can be directly opened via the context menu (right-click) of such a dialog element.

Control

"AxisInterface" Group

In this group, instances of the technology function MB_AxisInterfaceType01 are displayed. The special feature of this group is that instances are only displayed if an axis is also assigned to them. This takes only place via the variable AdminCtrl.Axis at PLC program runtime. If the "AxisInterface" is empty, either no axis is configured or the instance has not yet been assigned to an axis.

For a detailed description of the function block, refer to /27/, [Required and Supplementing Documentations, page 14](#), chapter "MB_AxisInterface-Type01".

MB_AxisInterfaceType01

Dialog	Description
Overview	Overview on the respective axis, status and operating state. Input option for position and velocity command values
Velocity	State monitoring in the operating mode MB_AX-IS_MODE.ModeVel. Command value specification possible for: • Velocity • Acceleration • Deceleration
Positioning	State monitoring in the operation modes for positioning. Command value specification possible for: • Position • Velocity • Acceleration • Deceleration
Synchronization	State monitoring in the operation modes for synchronization. Command value specification possible for: • Input and output compilation • Fine adjustment • Phase offset • Direction of synchronization • Starting mode • Special input fields for cams and FlexProfiles

Fig.6-21: Dialogs of the technology function MB_AxisInterfaceType01

"Web Tension" Group

The "Web Tension" group contains instances of the following technology functions:

- MB_TensionControlLoadCellType01

For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_Tension-ControlLoadCellType01"

- MB_TensionControlLoadCellType02

For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_TensionControlLoadCellType02"

MB_TensionControlLoadCellType01

Dialog	Description
Control loop	Schematic representation of the control loop
Limitation	Limitation variables

Fig. 6-22: Dialogs of the technology function MB_TensionControlLoadCell-Type01

MB_TensionControlLoadCellType02

Dialog	Description
Control loop	Schematic representation of the control loop
Limitation	Limitation variables
Process variables	Overview on process variables

Fig. 6-23: Dialogs of the technology function MB_TensionControlLoadCell-Type02

"Register Control" Group

The "Register Control" group contains instances of the following technology functions:

- MB_RegisterControllerType02
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_RegisterControllerType02"
- MB_RegisterControllerType04
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_RegisterControllerType04"
- MB_RegisterControllerSideType01
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_RegisterControllerSideType01"

MB_RegisterControllerType02

Dialog	Description
Mark reader	Settings of the mark reader
Control loop	Schematic representation of the control loop
PI-controller	Settings of the PI-controller
Limitation	Limitation variables
Correction window	Settings of the correction window
Status	General status

Fig. 6-24: Dialogs of the technology function MB_RegisterControllerType02

Control

MB_RegisterControllerType04

Dialog	Description
Mark reader	Settings of the mark reader
Control loop	Schematic representation of the control loop
PI-controller	Settings of the PI-controller
Limitation	Limitation variables
Correction window	Settings of the correction window

*Fig. 6-25: Dialogs of the technology function MB_RegisterControllerType04***MB_RegisterControllerSideType01**

Dialog	Description
Mark reader	Settings of the mark reader
Control loop	Schematic representation of the control loop
PI-controller	Settings of the PI-controller
Limitation	Limitation variables
Status	Settings of the correction window

*Fig. 6-26: Dialogs of the technology function MB_RegisterControllerSideType01***"Touch Probe" Group**

The "Touch Probe" group contains instances of the following technology functions:

- **MB_InitTouchProbe**
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_InitTouchProbe"
- **MB_TouchProbe**
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_TouchProbe"
- **MB_TouchProbeContinuous**
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_TouchProbeContinuous"
- **MB_WriteExpectWindow**
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_WriteExpectWindow"

MB_InitTouchProbe

Dialog	Description
Basic settings	Basic settings of the technology function

Expectation window	Settings of the expectation window
Status	General status

Fig. 6-27: Dialogs of the technology function MB_InitTouchProbe

MB_TouchProbe

Dialog	Description
Status	General status

Fig. 6-28: Dialogs of the technology function MB_TouchProbe

MB_TouchProbeContinuous

Dialog	Description
Status	General status

Fig. 6-29: Dialogs of the technology function MB_TouchProbeContinuous

MB_WriteExpectWindow

Dialog	Description
Expectation window	Settings of the expectation window
Status	General status

Fig. 6-30: Dialogs of the technology function MB_WriteExpectWindow

"Winder" Group

The "Winder" group contains instances of the following technology functions:

- MB_WinderDancerCtrlType01
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_WinderDancerCtrlType01"
- MB_WinderTensionCtrlType02
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_WinderTensionCtrlType02"

MB_WinderDancerCtrlType01

Dialog	Description
Dancer control	Control loop of the dancer
Dancer limitation	Limitation of the dancer controller
Diameter calculation	Diameter calculation of the winder
Diameter limitation	Limitation and average calculation of the diameter, gear fine adjustment
Status	General status

Fig. 6-31: Dialogs of the technology function MB_WinderDancerCtrlType01

Control

MB_WinderTensionCtrlType02

Dialog	Description
Open-loop tension control	Sensorless open-loop tension control and calculation of the torque limit. Display only in "OPEN_LOOP" mode.
Closed-loop tension control	Closed-loop tension control with load cell and calculation of the torque limit. Display only in "CLOSED_LOOP" mode.
Friction torque compensation	Characteristics for the compensation of the friction torque
Acceleration compensation	Characteristics for the compensation of the acceleration torque
Web tension	Parameterization of the web tension behavior
Diameter calculation	Diameter calculation of the winder
Diameter limitation	Limitation and average calculation of the diameter, gear fine adjustment
Status	General status

Fig. 6-32: Dialogs of the technology function MB_WinderTensionCtrlType02

"SmartBelt" Group

This group displays the instances of the technology function MB_SmartBeltType01.

For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "MB_SmartBeltType01".

MB_SmartBeltType01

Dialog	Description
Belt parameters	Measurements of the correction belt and basic settings
Correction parameters	Parameterization of the correction behavior
Status	General status, correction status and statistics

Fig. 6-33: Dialogs of the technology function MB_SmartBeltType01

"Sag Control" Group

The "Sag Control" group contains instances of the following technology functions:

- ML_SagControlA
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "ML_SagControlA"
- ML_SagControlC
For a detailed description of the function block, refer to /30/, [Required and Supplementing Documentations, page 14](#), chapter "ML_SagControlC"

ML_SagControlA

Dialog	Description
Sag control	Parameterization of the sag control
Status	General status, sag controller status

Fig. 6-34: Dialogs of the technology function ML_SagControlA

ML_SagControlC

Dialog	Description
Sag control	Parameterization of the sag control
Status	General status, sag controller status

Fig. 6-35: Dialogs of the technology function ML_SagControlC

6.3.5 Robot General Information

Predefined kinematics can be created in the "Robot" folder. How to create a kinematics is described in section [Creating a Kinematics, page 88](#).

Robot - Context Menu

The context menu of the "Robot" folder contains the menu items:

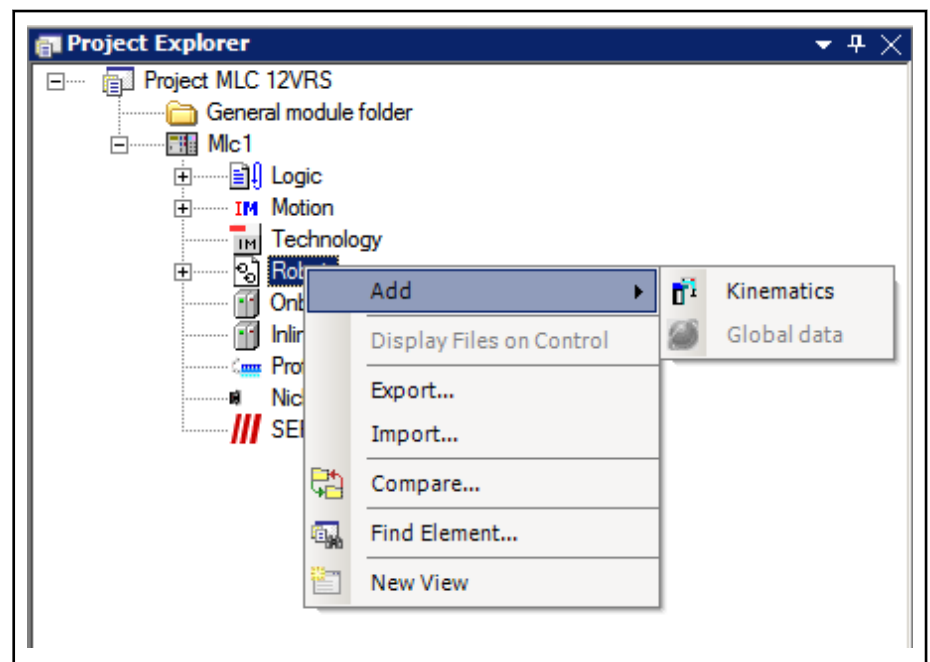


Fig. 6-36: Context menu of the "Robot" folder

- **Add...** - Adds a kinematics to the project, see [Creating a Kinematics, page 88](#)
- **Display Files on Control** - Selection dialog to display files (*.IRD, *.PKT, *.QLL, *.SYM) on the IndraMotion MLC with name, size and date (only in online mode).
- **Export...** - Saves robot elements in a file, see [Exporting, page 172](#)
- **Import...** - Imports robot elements from a file, see [Importing, page 174](#)

Control

- **Compare...** - Compares and merges objects, see [Comparing/Merging, page 176](#)
- **Find element...** - Allows searching for elements below the node
- **New View** - Displays node in individual view

6.3.6 Onboard I/Os



This node is only available for an IndraMotion MLC L45/L65.

The input/output environment of the control can be configured as follows:

- Onboard inputs/outputs (see also [Creating Onboard I/Os, page 62](#))

The individual options are described in detail in /4/, [Required and Supplementing Documentations, page 14](#), chapter "Configuring Onboard I/Os".

6.3.7 Inline I/Os

The input/output environment of the control can be configured as follows:

- Inline modules (see also [Creating Inline I/Os, page 64](#))

The individual options are described in detail in /4/, [Required and Supplementing Documentations, page 14](#), chapter "Configuring Inline I/Os".

6.3.8 Profibus/M

The individual options are described in detail in [Creating Profibus I/Os, page 71](#).

6.3.9 Profinet IO-Controller

If a control is created but no controller has been selected for the Slot1 configuration, the folder name "Not_Used" is displayed in the Explorer. Via the context menu item **Set Device** of the node, the Profinet IO controller can be created.

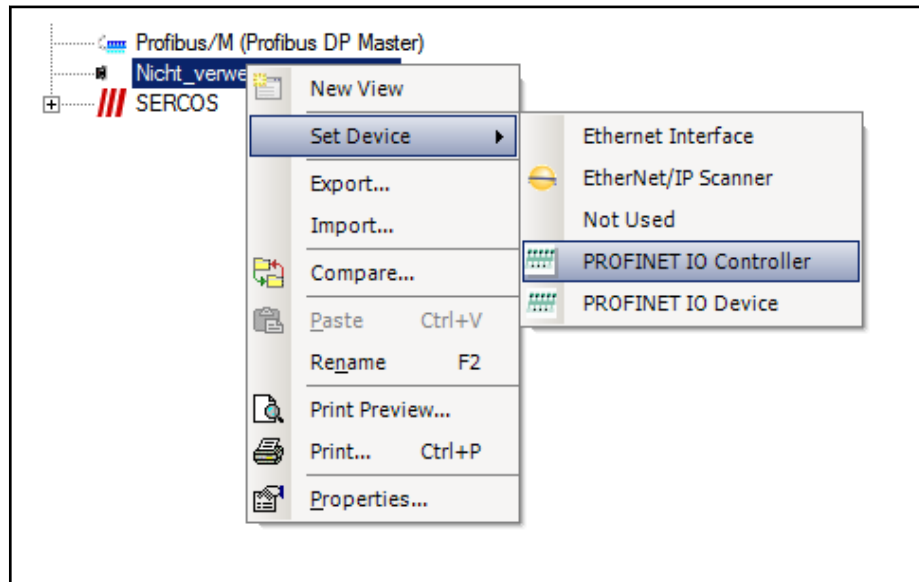


Fig.6-37: Creating a Profinet IO-Controller

The individual options are described in detail in [Creating a Profinet IO Controller, page 71](#).

6.3.10 Function Modules

The function module folders follow the "Profinet IO-Controller" folders.

For a detailed description of the individual function modules and their configurations, refer to /24/, [Required and Supplementing Documentations](#), page 14.

6.3.11 sercos

General Information

This node represents the sercos II and sercos III master communication. The sercos II master communication is only available when using the sercos II (CFL01.1-Q2) function module.

The sercos node of the control accepts all devices that run physically on the sercos bus (e.g. IndraDrive, SercosDrive, EcoDriveCS).

The current state of the sercos bus is displayed in square brackets in the node name if the control is in online mode.

The devices are transferred from the library to the node via drag&drop or copy&paste (for example, see [Creating a Real Axis](#), page 234). The addressing of the devices is specified by their set sercos address. Technically speaking, the user has thus fulfilled all prerequisites for the configuration (sequence in the Project Explorer).

If a device (real axis) was dragged to the sercos node, it is displayed twice in the Project Explorer:

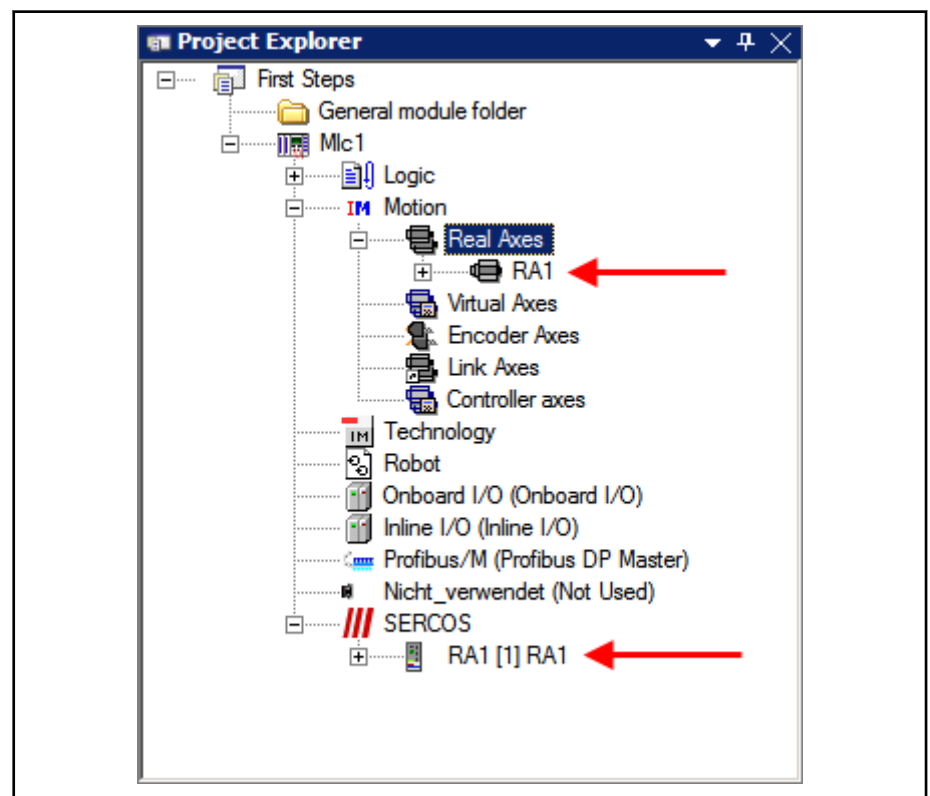


Fig.6-38: Creating a real axis RA1 - Overview in the Project Explorer

Control



Use the upper dialogs under **Motion -> Real Axes-> <AxisName>** to configure the respective device.

The dialogs under **sercos -> <AxisName>** are special drive dialogs only accessing S- and P-parameters.

Caution: The settings made in the drive dialogs by the user are partially overwritten by the A-parameter setting of the upper dialogs without warning (see chapter "Parameterization and Programming, A- and C-parameters Overlap S- and P-parameters).

sercos - Context Menu

Overview on Menu Items

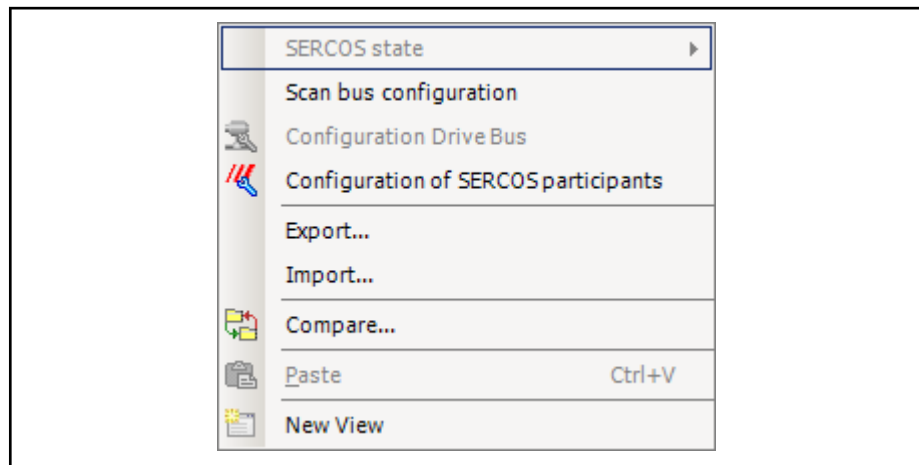


Fig. 6-39: Context menu of the sercos node

The context menu of the sercos node contains further functions apart from the default entries:

- **sercos state** - see [sercos State](#)
- **Scan bus configuration** - see [Scanning Bus Configuration](#)
- **Drive bus configuration** - see [Configuring the Drive Bus \(sercos II\)](#)
- **Configuring sercos devices** - see [Configuring sercos Devices](#)

sercos State

The "sercos state" menu item allows switching to the target phase of the control (see also [chapter 9.2 "Motion Mode" on page 345](#)).

Scanning Bus Configuration

The "Scan bus configuration" menu item determines the devices connected to the sercos bus (see also [chapter "Scanning the Ring" on page 352](#)).

Configuring sercos Devices

The "Configuration of sercos Devices" menu item opens a display of all sercos devices (see also [chapter "Adjustment of the Command/Actual Configuration" on page 348](#)).

Configuring the Drive Bus (sercos II)

This dialog allows setting the baud rate and the synchronization of control parameters to the drive parameters during phase switching from P2 to BB for the sercos II master communication.

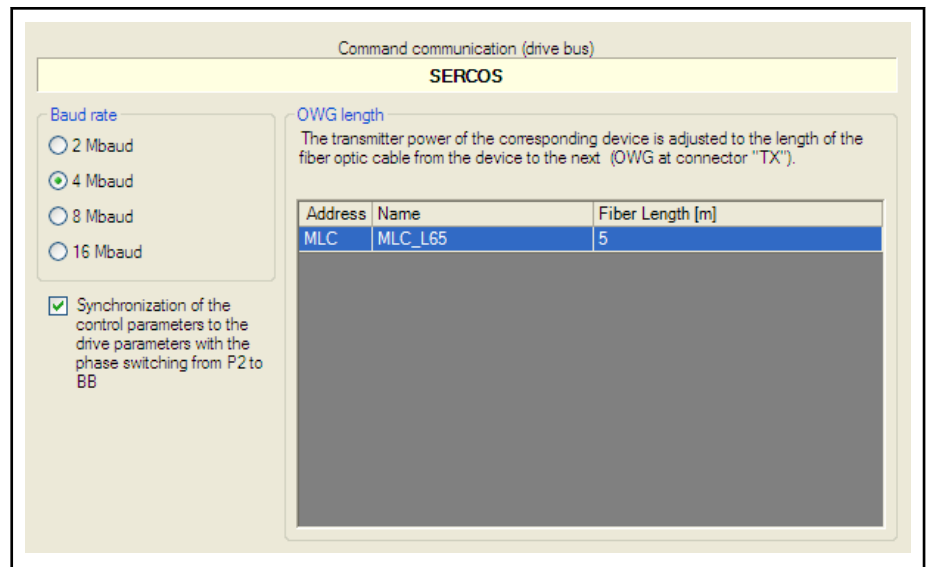


Fig.6-40: Settings for the sercos II bus

sercos - Toolbar

The sercos toolbar shows the current phase and provides direct access to frequently required functions. It is structured as follows:

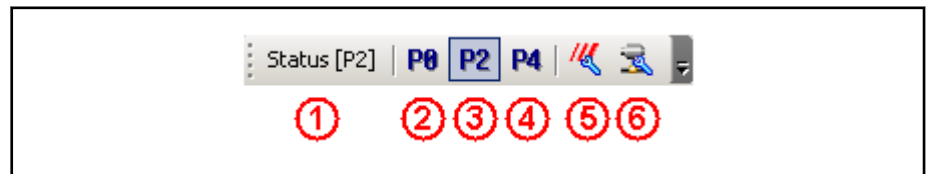


Fig.6-41: sercos toolbar

- Display of the current sercos phase ①
- Phase switching to P0 (only online) ②
- Phase switching to P2 (only online) ③
- Phase switching to P4 (only online) ④
- Start of the "Configuration of sercos Devices" (always available) ⑤
- Start of the "Configuration Drive Bus" dialog (only online if a sercos II master communication is configured) ⑥

6.4 Control – Context Menu

6.4.1 Overview on Menu Items

The context menu of the control contains the following menu items:

Control

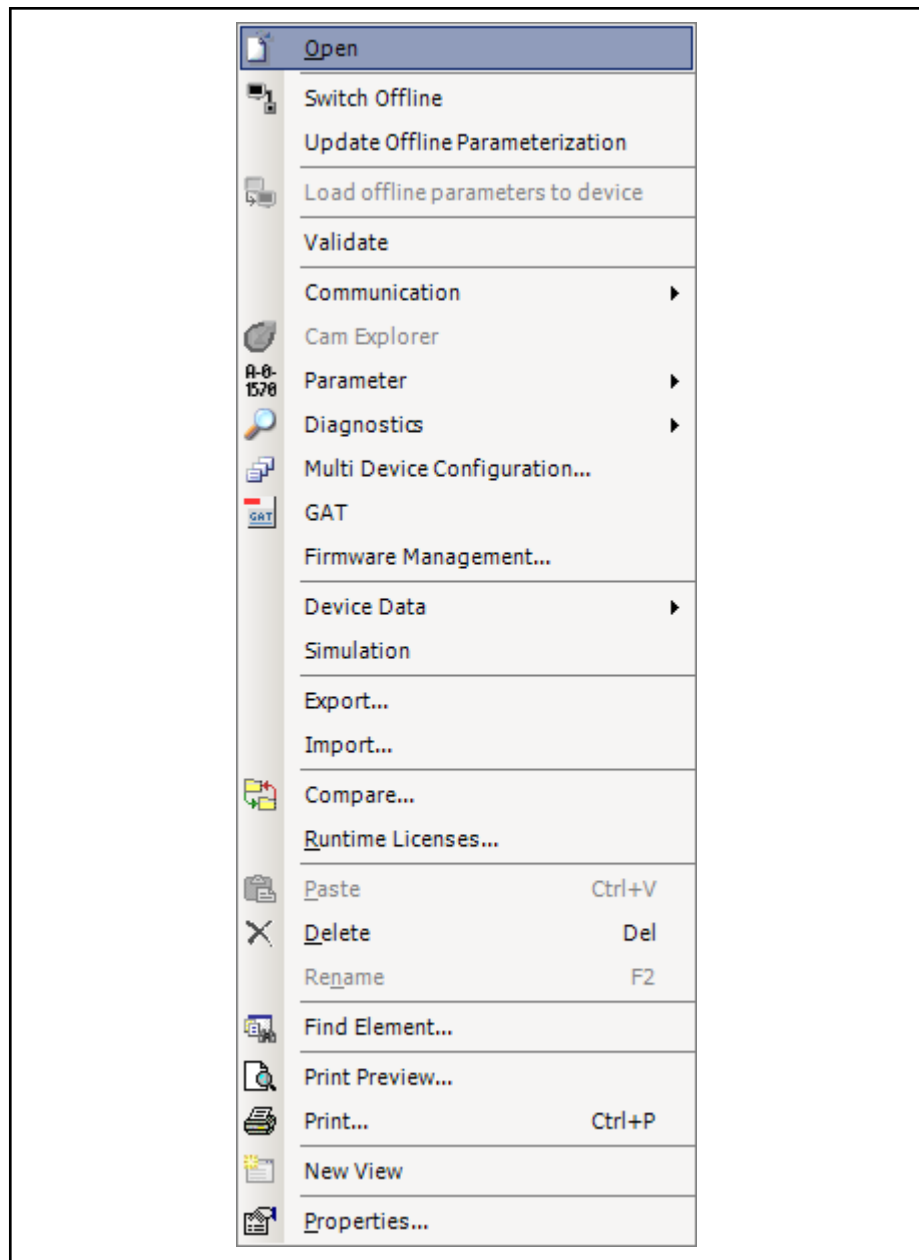


Fig. 6-42: Context menu of an IndraMotion MLC in online mode

- **Open**, see [Device Editor](#), page 122
- **Switch Offline** - see [Modes for Project Editing](#), page 346
- **Update Offline Parameterization** - see [Offline Parameterization](#), page 347
- **Load Offline Parameters to Device** - see [Loading Offline Parameters to the Device](#), page 356
- **Validate**, see [Validating Motion Configuration](#), page 379
- **Communication** - Setting the communication parameters, see [Communication](#), page 125 or setting the control IP communication, or [IP Communication](#), page 76
- **Language Settings** - Language setting of the control, see [Language Settings](#), page 126

- **Cam Explorer**, see [Cam Explorer](#), page 127
- **Parameter**, see [Parameters](#), page 133
- **Diagnostics** - Contains the subitems
 - [Device Status](#), page 141
 - [Advanced Properties](#), page 143
 - [Clearing Errors](#), page 144 - Allows the deletion of all control errors and drive errors
 - [Error/Diagnostic Memory](#), page 144 - Provides an overview on the current diagnostic status of the control and its drives
 - [Invalid Parameters](#), page 147 - Displays the parameter "Diagnostics (C-0-0626)" and lists the invalid parameters
 - [Cycle Times](#), page 102 - Displays the time history of the Motion kernel, PLC and communication tasks
 - [Task List/Configuration](#), page 210 - allows the configuration of the existing tasks and priorities
 - [Task Execution Viewer](#), page 212 - Displays the existing tasks and priorities
- **Multi Device Configuration...** - Allows the use of an application and its parameters for several controls, see [Multi-Device](#), page 150
- **GAT - Generic Application Template**, structured template to add and edit PLC program code (also refer to /27/ in [Required and Supplementing Documentations](#), page 14)
- **Firmware Management...** - Loads and updates the control firmware, see [Firmware Management](#), page 161
- **Device Data*** - Saves and restores device data or files and folders from PC to control and vice versa, see [Device Data](#), page 163
- **Simulation** - Simulates the control without real hardware, see [Simulation](#), page 172
- **Export...** - Saves elements from the current project in a file, see [Exporting](#), page 172
- **Import...** - Imports elements into the current project from a file, see [Importing](#), page 174
- **Compare...** - Compares and merges objects, see [Comparing/Merging](#), page 176
- **Runtime licenses...** - This dialog outputs all licenses used by the PLC program at runtime, see [Runtime Licenses](#), page 178
- **Delete** - Deletes the control and its subfolders. **Only available offline**
- **Rename** - Renames the control. **Only available offline**
- **Find element...** - Allows searching for elements in the project
- **Print preview...** - Provides a preview of project and device data, see [Print Preview](#), page 178
- **Print...** - Prints project and device data, see [Printing](#), page 180
- **New View** - Displays node in individual view
- **Properties...** - Displays and modifies the control properties, see [Creating a Control](#), page 53

Control

6.4.2 Device Editor

Device Editor - General Information

The device editor provides dialogs to configure a control.

By default, double-click on the control to open the **Device dialog** when it is selected in the Project Explorer. Another option is the "Open" menu item in the context menu of the control.

The title of the main dialog is the device name ("Mlc1" in the following figures) and, depending on the device type, it contains a combination of the following subdialogs:

- [Applications, page 122](#): List of the applications on the control
- [Log, page 123](#): Displays the PLC log file
- [PLC settings, page 124](#)
- [Information, page 125](#): General information on the device (name, vendor, version, etc.)

Applications

This tab is used to display and, if required, to delete applications currently located on the control.

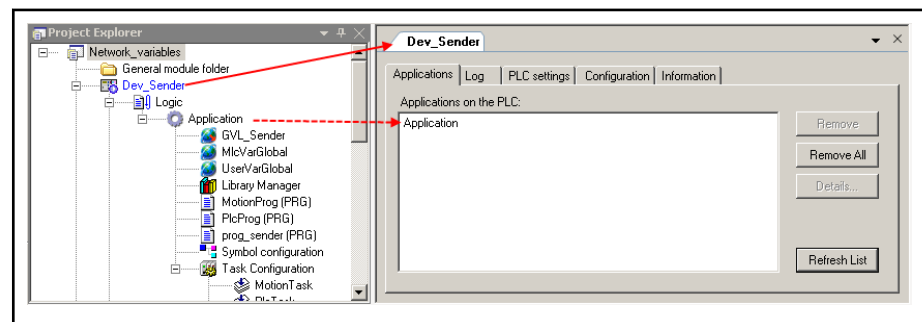


Fig. 6-43: Device dialog, Application

Applications on the PLC: List of the applications found on the control during the latest "scan" (with "Refresh").

Refresh List: The control is searched for applications and the list is refreshed accordingly.

Remove or **Remove All:** The currently selected applications or all applications are deleted from the control.

Details: Provides more information on the application.

If an application is loaded to the control, the following is checked first:

- The list of applications on the control is compared with those available in the project. If the lists do not match, the corresponding dialogs are displayed either to load the applications not already present on the control or to delete other applications on the control.
- The "externally implemented" function blocks in the application to be loaded are checked to see if these are also available on the control. If this is not the case, a corresponding message ("Unresolved reference(s)") is output in a message box and in the message window
- The parameters (variables) of the function blocks in the application to be loaded are compared with those in the function blocks of the same name in the application already present on the control (signature check).

If they do not match, a corresponding message ("Invalid signature(s)") is output in a message box and in the message window

Log

This tab is used to display the control logbook, i.e. to display events recorded on the target system. This applies to the:

- Events at system start and shutdown (loaded components with version)
- Application download and loading of the boot project
- Customer-specific entries
- Log entries from I/O drivers
- Log entries from the data server

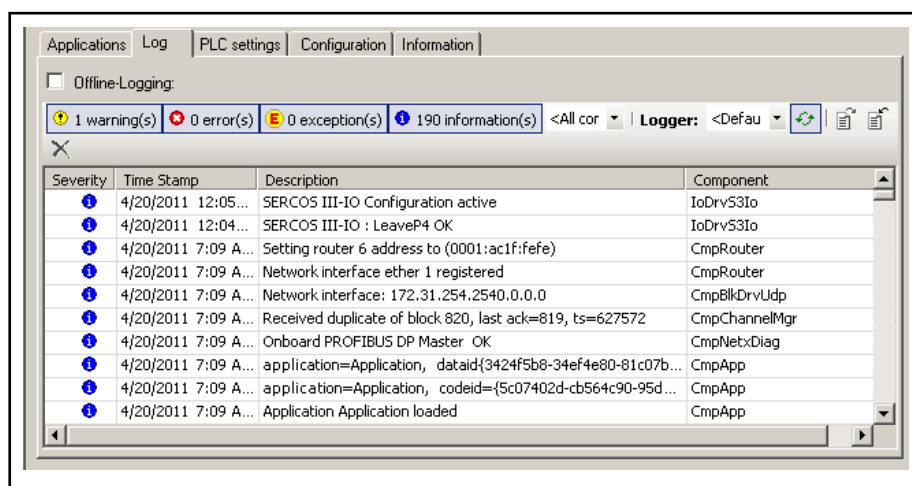


Fig.6-44: Device editor, Log

A logbook entry is displayed with the following information:

Severity (Scaling): There are four categories for the severity of the event: Warning(s), error(s), exception(s), information. The display of each category can be displayed or hidden by using the corresponding button in the bar above the list. The respective number of log entries are displayed for the respective category on the button.

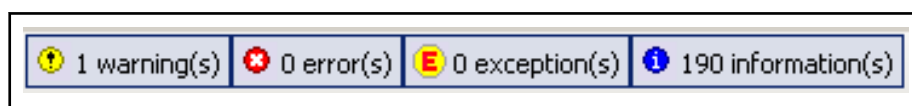


Fig.6-45: Severity of event

Time Stamp: Date and time, e.g. "116.11.2010 14:00".

Description: Event description, e.g. "sercos III I/O: SetParamP4 OK".

Component: Name of the respective component.

Com Name (component name): It can be searched for an individual component in a selection list to display only related logbook entries. The default setting is "<All components>".

Logger: The selection list contains the available recordings. The default setting is the "<Default>" specified by the target system, currently the same as "PlcLog" for the IndraLogic runtime system.

Automatic update is enabled using the button.

The content of the list can be exported into an XML file. Press the button to open the standard dialog for saving files. The file filter is set to "xml-files"

Control

("*.xml)". The log list is stored in the selected directory with the file name entered and the extension ".xls". If the Offline logging option is enabled, even actions that are not related to the connection with the control are recorded. However, at this time it can only be implemented in the safety version of the programming system.

An already existing log list in XML format can be imported via the symbol .

Select the  icon to delete all entries in the log list.



For further diagnostic options, refer to [chapter 6.4.7 "Diagnostics" on page 141](#).

PLC Settings

This tab is used to specify the control behavior in the "Stop" state.

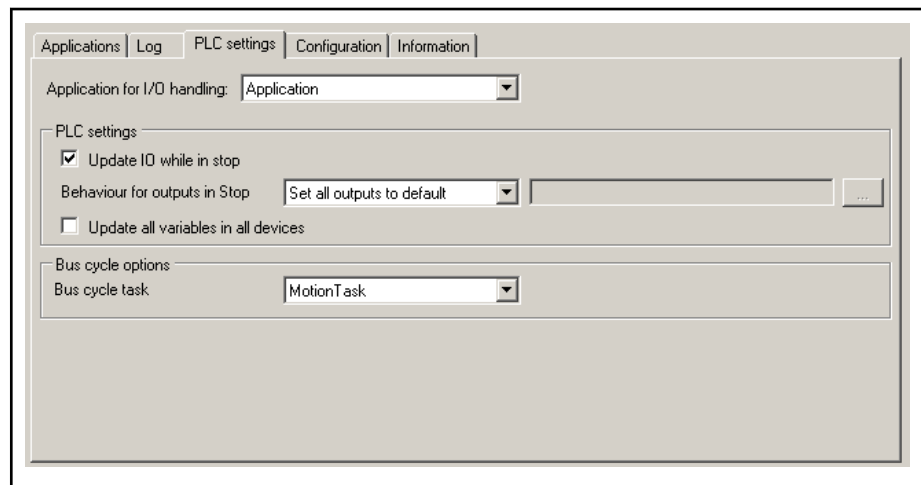


Fig. 6-46: Device editor, PLC settings

Application for I/O handling: If several applications are available for the device, they are listed here. The default application, which is automatically created with a default project, is always entered first.

PLC settings: **Update IO while in stop:** If this option is enabled (default), the values for input and output channels are updated if the PLC goes into "Stop" state.

Behavior of the outputs at stop: The selection list provides the following options to handle the values of the output channel when the control goes into "Stop" state:

- **Retain current values:** The current values are retained.
- **Set all outputs to default:** The default values from the I/O mapping are assigned.
- **Execute program:** Control the handling of the output values using a program in the project. Enter this program name here and it is executed when the control goes into "Stop" state. Use the ... button to open the input help to facilitate the program selection.

Update all variables in the devices: If the option is enabled, all variables are updated.

Bus cycle options: **Bus cycle task:** The selection list provides all tasks defined in the task configuration of the active application (e.g. "MotionTask", "PlcTask").



Check the task data of the application and enter the desired task!

Control

Select a specific task to control the bus cycle (I/O mapping) or select the "Unspecified" setting if the task with the shortest cycle time, i.e. the fastest task, is to be used here. This setting is global for all I/O devices. However, it can be overwritten locally by every I/O device (see [chapter "Profibus, Profinet and Inline I/Os" on page 214](#)).

Information

This tab in the **Device dialog** displays some general information taken from the device description file on the device currently selected in the Project Explorer:

- Name
- Vendor
- Categories
- Version
- Ordering number
- Description
- Possibly figure

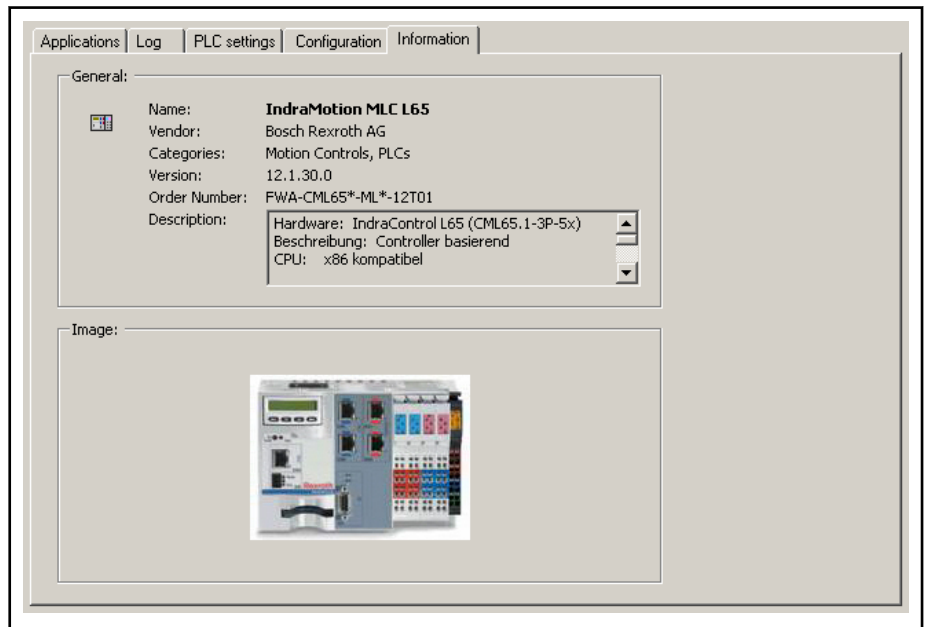


Fig.6-47: Device editor, Information

6.4.3 Communication

TCP Settings

The dialog box displays the settings for the Ethernet connection between PC and control. Some values can be adjusted.

Control

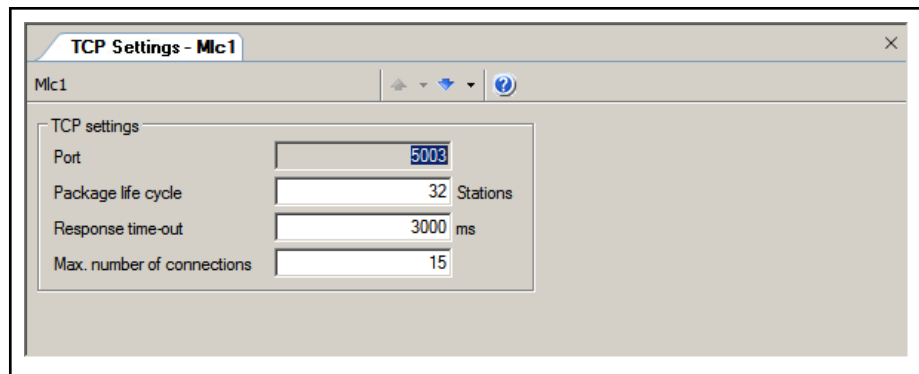


Fig.6-48: TCP settings

Port: Specifying the port used for the TCP communication.

Package life cycle: Specifies the amount of network nodes forwarding, rejecting or discarding an IP package.

Response time-out: "Response timeout" specifies the period that is waited for the absent response until an error message is output.

Max. number of connections: Specifies the number of connections that a control can operate simultaneously. If this number is exceeded, the connection cannot be established.

Parameters involved

- C-0-0208, TCP port for acyclic channels
- C-0-0206, TCP time to live
- C-0-0209, Sender Timeout
- C-0-0210, TCP maximum number of connections

Interfaces

The settings for the IP communication are made via this menu item. For a detailed description, refer to [IP Communication](#) , page 76.

6.4.4 Language Settings

Call	It is set via the context menu item "Language Settings" of the control.
Goal	Use this dialog to set the language of the control and of all connected devices (IndraDrive/HydraulicDrive).
Language of the control	The control supports the languages "German" and "English". Additionally, a user-defined language can be selected. The user-defined language can load language files for the parameter names, the diagnostic messages and the display texts to the control. If a language file is loaded to the control, the texts from this file are used subsequently. The default user-defined language is "English".
Language of the drives	Every language supported by the drive can be selected. This language is immediately transferred to all connected devices. A drive has to be online and active to transfer a language. If the number of connected active drives changes, the language can be transferred again via the Rewrite button.

Language settings of the control

Control language: German

User-defined language files

File with parameter names: <Please select language file.> ...

File with diagnostic messages: <Please select language file.> ...

File with control display names: <Please select language file.> ...

Download

Language settings of the drives (IndraDrive / HydraulicDrive)

Drive language: French

Last selected language: French

Rewrite

Fig. 6-49: Language settings

For more information on the language setting, refer to [chapter 13 "User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts"](#) on page 495.

Parameters involved

- C-0-0001, Language selection
- C-0-0002, Name of parameter file
- C-0-0003, Name of diagnostic file
- C-0-0004, Name of the menu file
- S-0-0265, Language selection

6.4.5 Cam Explorer

The "Cam Explorer" dialog provides access to three areas:

- CamPool
- Compact Flash card
- Control

The Cam Explorer can be used to copy cams into each of the areas (via drag&drop or using the arrow keys) or copy them back.

Control

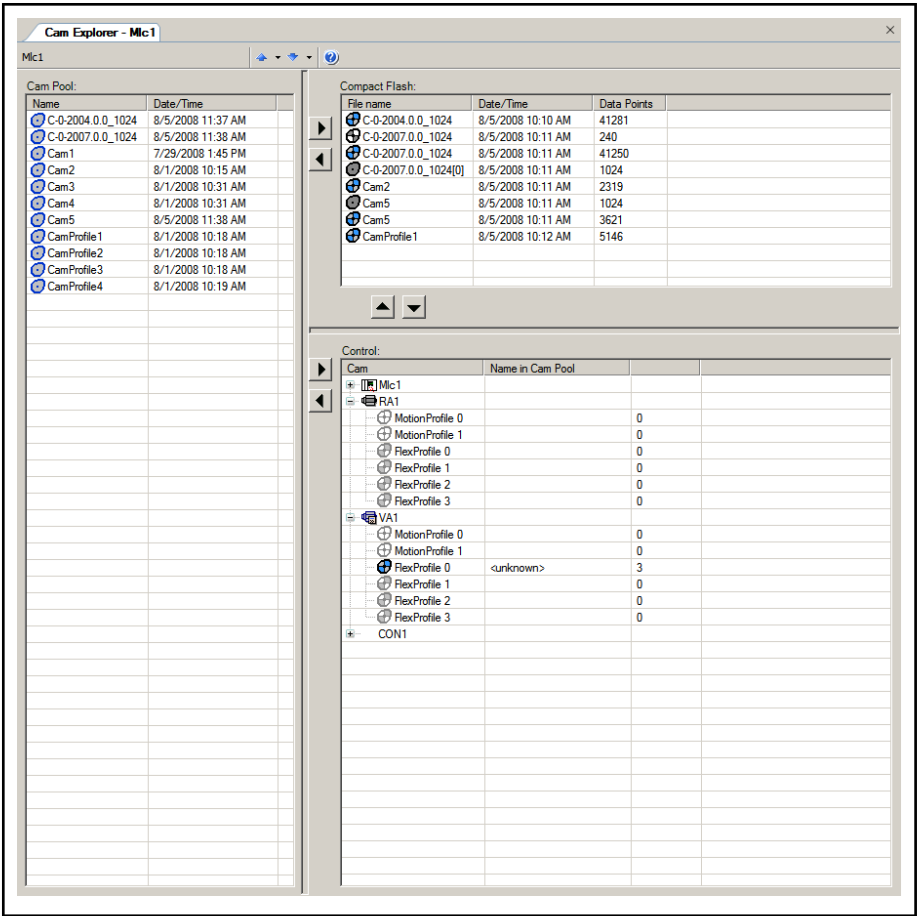


Fig. 6-50: "Cam Explorer" dialog:

The stock of cams is displayed in the left field of the dialog under "Cam Pool". In general, they can be generated with the CamBuilder. However, they can also be copied back from the Compact Flash card, the control or an axis.



To use the CamBuilder, a license is required. This has to be entered under **Tools ▶ Options ▶ General ▶ Software Licenses ▶ CamBuilder**.



Cam tables and MotionProfiles or FlexProfiles are generated almost identically using the CamBuilder. Principally, each cam table is also a MotionProfile or a FlexProfile, i.e. it consists of segments with certain mathematical properties (= profile).

While transferring a cam into the control or the Compact Flash card, a cam table with the desired number of points is always generated. The respective parameters (C-0-2001 ... C-0-2099) are list parameters that contain only the points. Reloading a cam table from one parameter always leads to a new cam that only contains data points and that no longer indicates any segments or their mathematical description.

MotionProfiles or FlexProfiles are stored in the control as a description of individual segments and can usually be reloaded in the same way.

In general, a MotionProfile or FlexProfile requires less storage area, but more computing power for the execution. Due to the kept segmentation, it is also possible to carry out modifications on the existing segments at runtime (e.g. program-controlled fine adjustment). When modifying cam tables, the complete cam table always has to be transferred.

In the dialog on the right top, all cam table files and MotionProfile or FlexProfile files - stored on the Compact Flash Card - are displayed. Cam tables, MotionProfiles and FlexProfiles are displayed with the corresponding icon in each line. For cam tables, the number of data points in the cam table is displayed in the third column. For MotionProfiles and FlexProfiles, the file size is displayed in bytes.

In the dialog on the right bottom, all cam tables, MotionProfiles and FlexProfiles - stored in the parameters of the control or the axes - are displayed. The following applies: Cam tables can only be stored in the parameters C-0-2001 ... C-0-2099 of the control. Motion and FlexProfiles can only be stored in the A-parameters of the respective axes. For cam table parameters, the number of data points in the cam table is displayed in the third column. For MotionProfiles and FlexProfiles, the number of segments used is displayed.



MotionProfiles and FlexProfiles exist in axes, whose interpolation takes place in the control. This always applies for virtual axes, real axis with interpolation on the control and sercos axes.

Cams (cam tables, MotionProfiles or FlexProfiles) can be copied from any of the three partial windows into the other. There are three possibilities:

- Arrow keys between the partial windows
- Via a command in the context menu
- Drag&drop



If the arrow keys are used, it is necessary to at least select the element to be copied (single click). If the elements are copied to parameters, the copy target has to be selected as well.

When copying a cam from the CamPool to the Compact Flash card, the target format has to be selected via the dialog first.

Control

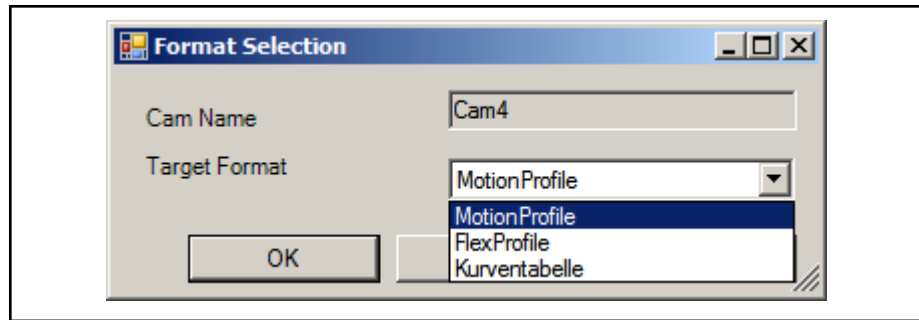


Fig.6-51: Selecting the target format on the Compact Flash card

If the target format is a cam table, a transformation into a cam table is always performed. Another dialog is displayed in which the number of data points and the file names on the Compact Flash card are displayed.

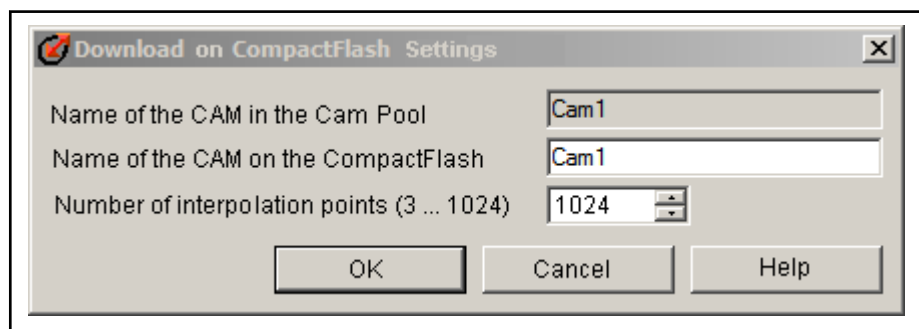


Fig.6-52: Specifying the name and number of data points

When copying from the Compact Flash card to the control, the number of data points can no longer be changed. The input field is disabled.

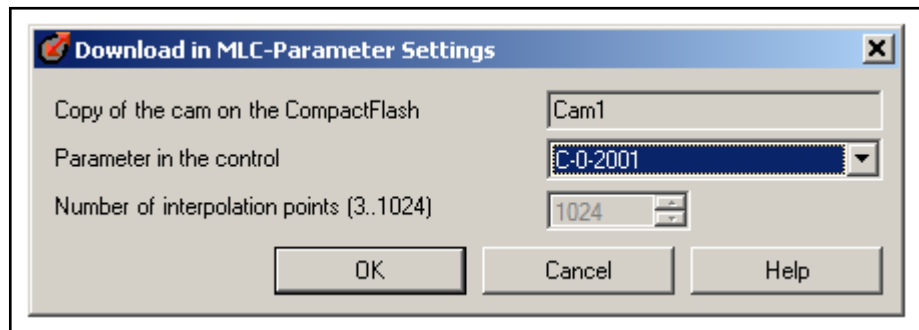


Fig.6-53: Settings to copy a cam table from the Compact Flash card to the control



Cam tables copied to the Compact Flash Card are stored so that they can be copied into cam table parameters without being converted. No mathematical segment description, but a cam table, is stored on the Compact Flash card.

In the PLC program, suitable function blocks can be used to access stored files.

MotionProfiles or FlexProfiles are also stored on the Compact Flash card to copy them with reasonable effort into the parameters involved.

Cam tables stored on the Compact Flash card have the file extension ".cam", MotionProfiles ".cpf" and FlexProfiles ".xml".

MotionProfiles or FlexProfiles can also contain cam tables in individual segments. While copying these profiles to an axis parameter, the storage location of the cam tables contained have to be queried. The following window is opened for selection:

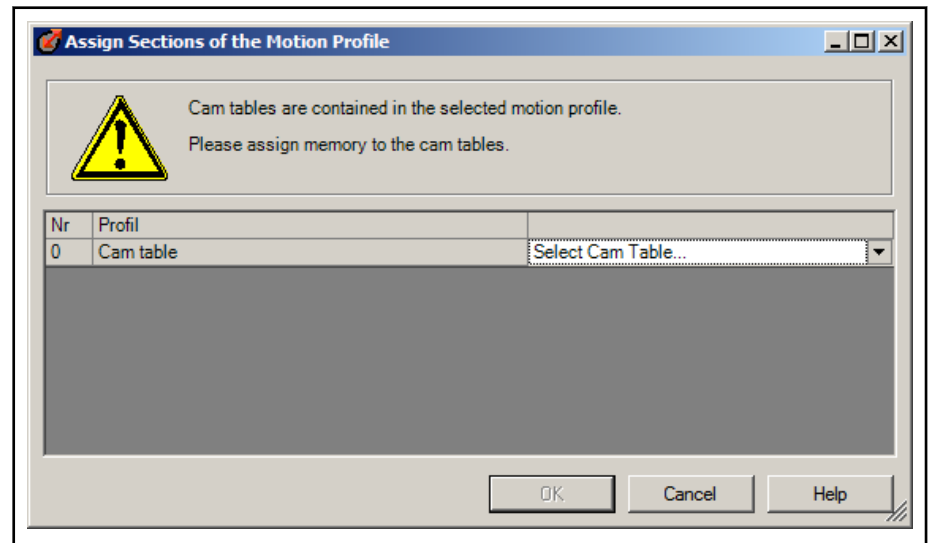


Fig.6-54: Selecting cam table parameter while downloading a MotionProfile or FlexProfile to an axis

Parameters involved:

Cam tables

- C-0-2001 ... 2099, Control cam tables

MotionProfile

- A-0-2910/A-0-2920, MotionProfile 0-1, number of motion steps
- A-0-2911/A-0-2921, MotionProfile 0-1, master axis velocity
- A-0-2912/A-0-2922, MotionProfile 0-1, list of initial master axis positions
- A-0-2913/A-0-2923, MotionProfile 0-1, list of motion laws
- A-0-2914/A-0-2924, MotionProfile 0-1, list of distances
- A-0-2915/A-0-2925, MotionProfile 0-1, list of slave axis velocities

FlexProfile

- A-0-3000, FlexProfile, current set
- A-0-3001, FlexProfile, current step
- A-0-3002, FlexProfile, master axis switching position
- A-0-3003, FlexProfile, slave axis, additive command position value
- A-0-3004, FlexProfile, master axis, additive position command value
- A-0-3009, FlexProfile, control word
- A-0-3010, FlexProfile, status word
- A-0-3011 FlexProfile, conditions: control variables
- A-0-3012 FlexProfile, conditions: status variables
- A-0-3013, FlexProfile, value, current time-master
- A-0-3020/A-0-3050/A-03080/A-0-3110, FlexProfile 0-3, master axis velocity
- A-0-3021/A-0-3051/A-03081/A-0-3111, FlexProfile 0-3, list of master axis sources

Control

- A-0-3022/A-0-3052/A-03082/A-0-3112, FlexProfile 0-3, list of slave axis distances
- A-0-3023/A-0-3053/A-03081/A-0-3113, FlexProfile 0-3, list of master axis ranges
- A-0-3024/A-0-3054/A-03081/A-0-3114, FlexProfile 0-3, list of master axis ranges, units
- A-0-3025/A-0-3055/A-03085/A-0-3115, FlexProfile 0-3, list of motion laws
- A-0-3026/A-0-3056/A-03086/A-0-3116, FlexProfile 0-3, list of motion step types
- A-0-3027/A-0-3057/A-03087/A-0-3117, FlexProfile 0-3, list of slave axis starting velocities
- A-0-3028/A-0-3058/A-03088/A-0-3118, FlexProfile 0-3, list of slave axis starting accelerations
- A-0-3029/A-0-3059/A-03089/A-0-3119, FlexProfile 0-3, list of slave axis starting jerks
- A-0-3030/A-0-3060/A-03090/A-0-3120, FlexProfile 0-3, list of slave axis end velocities
- A-0-3031/A-0-3061/A-03091/A-0-3121, FlexProfile 0-3, list of slave axis end accelerations
- A-0-3032/A-0-3062/A-03092/A-0-3122, FlexProfile 0-3, list of slave axis end jerks
- A-0-3033/A-0-3063/A-03093/A-0-3123, FlexProfile 0-3, list of slave axis travel velocities
- A-0-3034/A-0-3064/A-03094/A-0-3124, FlexProfile 0-3, list of slave axis travel accelerations
- A-0-3035/A-0-3065/A-03095/A-0-3125, FlexProfile 0-3, list of slave axis limit jerks
- A-0-3036/A-0-3066/A-03096/A-0-3126, FlexProfile 0-3, list of turning point displacements
- A-0-3040/A-0-3070/A-03100/A-0-3130, FlexProfile 0-3, events, list of motion step assignments
- A-0-3041/A-0-3071/A-03100/A-0-3131, FlexProfile 0-3, events, list of trigger modes
- A-0-3042/A-0-3072/A-03100/A-0-3132, FlexProfile 0-3, events, list of action modes
- A-0-3043/A-0-3073/A-03100/A-0-3133, FlexProfile 0-3, events, list of trigger values
- A-0-3044/A-0-3074/A-03100/A-0-3134, FlexProfile 0-3, events, list of trigger values, units
- A-0-3046/A-0-3076/A-03106/A-0-3136, FlexProfile 0-3, Command: Check and apply event
- A-0-3047/A-0-3077/A-03107/A-0-3137, FlexProfile 0-3, Diagnostics
- A-0-3048/A-0-3078/A-03108/A-0-3138, FlexProfile 0-3, Command: Check and apply profile
- A-0-3049/A-0-3079/A-03109/A-0-3139, FlexProfile 0-3, checksum

6.4.6 Parameters

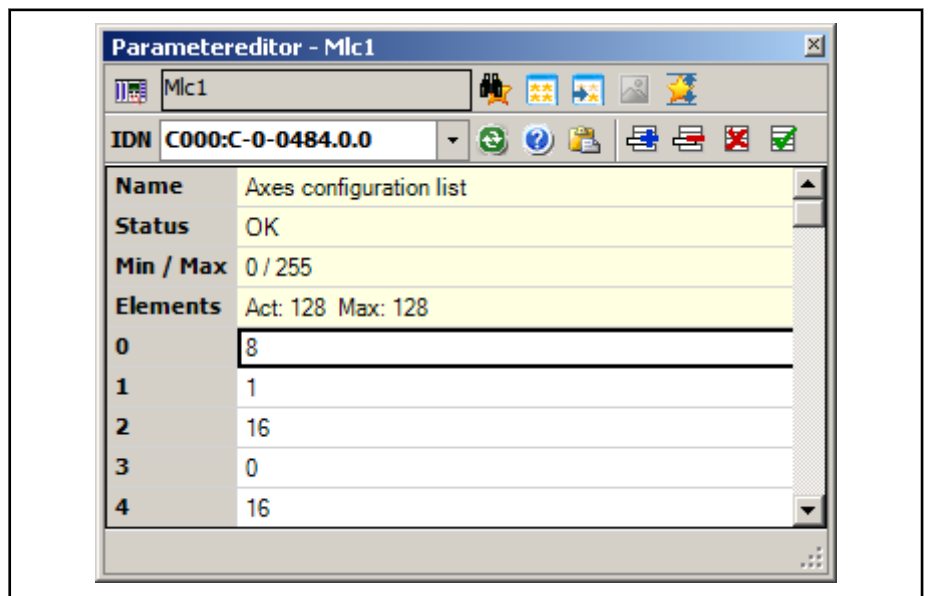
Parameter Editor

The parameter editor allows an individual parameter to be displayed and changed. Its call is context sensitive, that is it provides C-parameters in the control area and for each axis the A-, S- and P-parameters of the respective axis or the S- and P-parameters with drives.

Availability This function is enabled in

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

Structure The parameter editor can be called in the context menu of control, axes and drives and by right-clicking on a parameter in any dialog:



Name Parameter name
Status Query status (transfer)
Min Possible minimum value of the parameter
Max Possible maximum value of the parameter
Elements Value or list of values

Fig.6-55: Parameter editor call

Actual parameters are entered taking the source into account:

In the figure above, the number of the control "C000", local control, is placed in the front. For the parameter "C-0-0484, Axes configuration list", the complete designation "C000:C-0-0484" results.

The A-, S- and P-parameters can be called for an actual axis.

The parameter name is extended with the axis number (A001...A016):

A001: A-0-0002

for "A-0-0002, Axis name" of the axis "1".

If it is accessed from the **local axis**, the axis number can be omitted.



If the parameter can be written in the current mode, the values can be changed and transferred with <ENTER>.

Control

Toolbar of the parameter editor

The following functionalities can be accessed via the toolbar above the tabular parameter display:

Icon	Meaning
	Opens the Parameter selection list, page 134
	Opens the parameter group
	Transfers the parameter in the parameter group
	New window
	Adjusts window size to content
	New reading, updates the display
	The parameter help opens the .chm help file
	Copies the parameter values to the clipboard
	Inserts a list element at an actual position (list is not immediately updated) in case of list parameters
	Deletes a list element at an actual position (list is not immediately updated) in case of list parameters
	Deletes list elements
	Writes the list into the corresponding parameter

Fig. 6-56: *Toolbar of the parameter editor*

Parameter selection list

The parameter selection list provides a complete overview on the currently available parameters and allows the desired parameter to be selected for display and editing in the parameter editor and in the parameter group (depending on the location of the call).

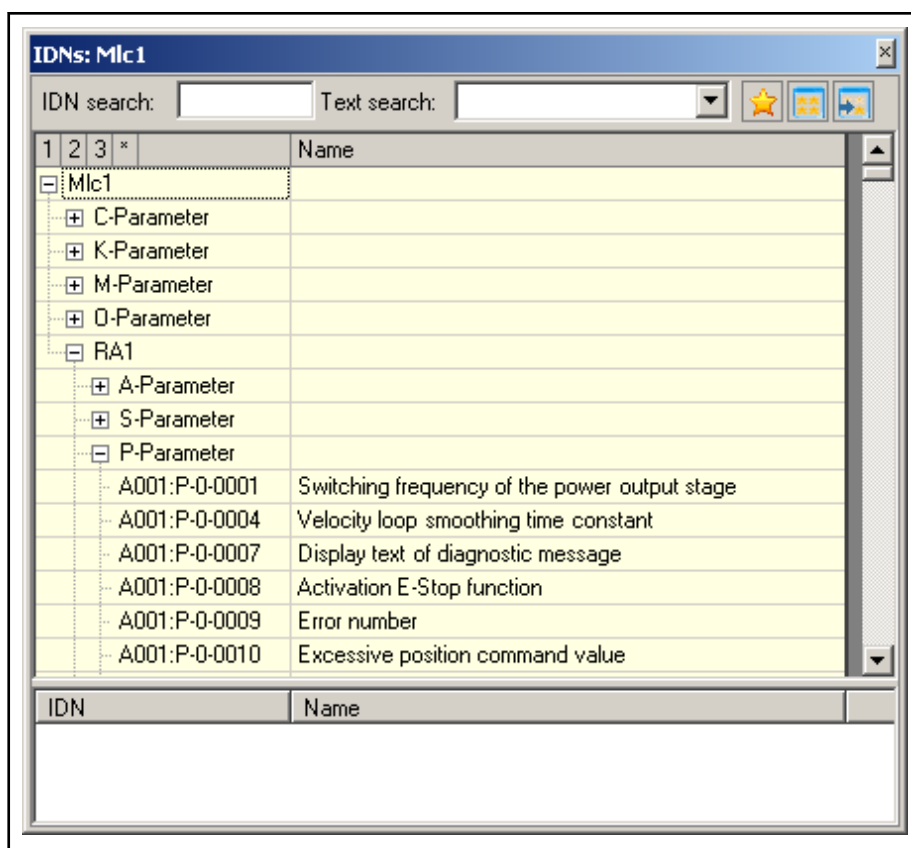


Fig.6-57: Calling the parameter list from the parameter editor

Parameter Group

Based on the individual parameter displayed in the [Parameter Editor](#), page 133, the user can compose a list of parameters whose values can be evaluated and changed.

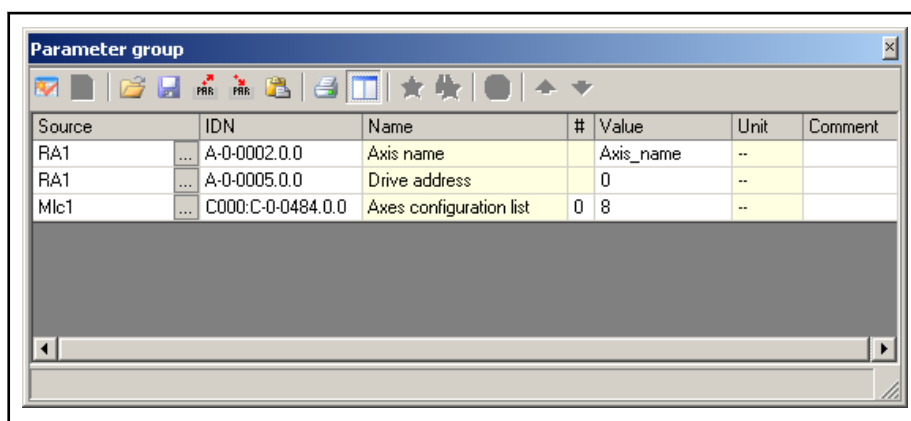


Fig.6-58: Parameter group

Availability

This function is enabled in

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

Toolbar of the parameter group

The following functionalities can be accessed via the toolbar above the tabular parameter display:

Control

Icon	Meaning
	Inserts a new line to enter parameters (selection dialog)
	Deletes the current line
	Loads the parameter list
	Saves the parameter list
	Imports parameter file (*.par)
	Exports parameter file (*.par)
	Copies the content to the clipboard as text
	Prints the parameter group
	Displays the parameter source
	Opens the Parameter Editor , page 133
	Selects a parameter (selection dialog), see also Parameter Editor , page 133
	Parameter description, opens the help file
	Moves the line up/down

Fig. 6-59: *Toolbar of the parameter group*

Saving Parameters in Drives

This function is used to explicitly save parameters in sercos drives in the Flash. This is necessary as the drives can only perform parameter changes in the working memory.

Availability

This function is enabled in

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

Parameters - Export

With the export function, the parameters can either be completely or individually saved from a control to a file.

The export function can be called via the context menu of the control.

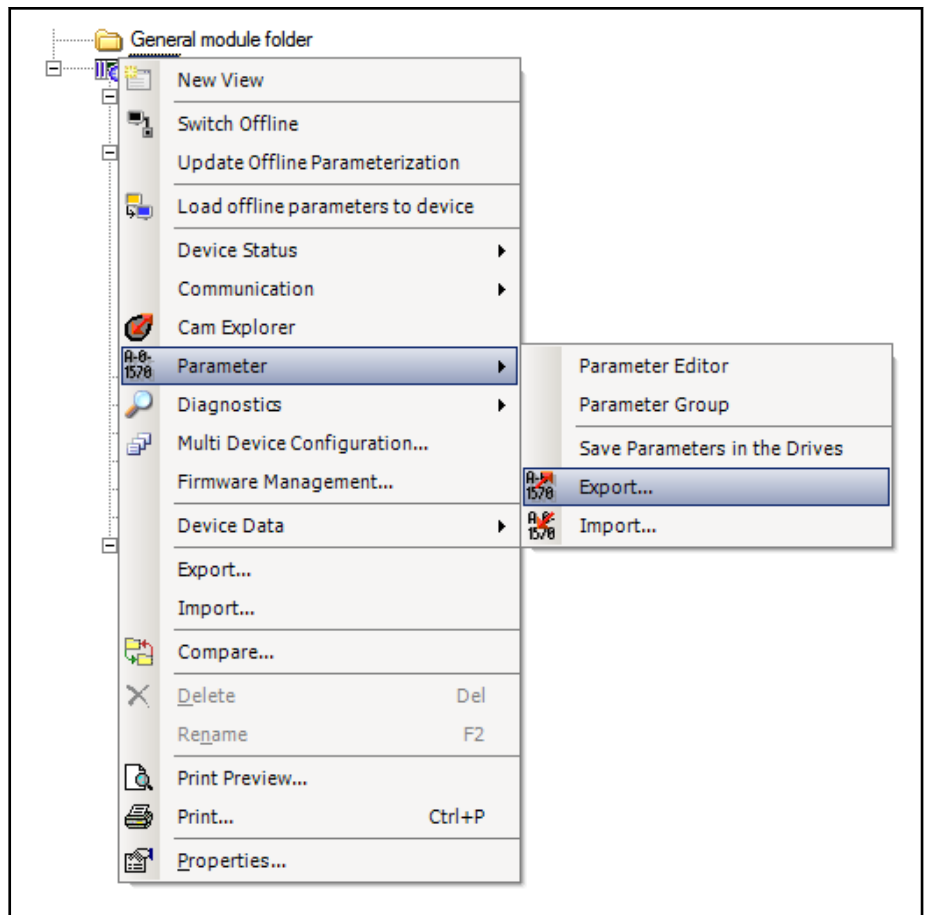


Fig. 6-60: Context menu items "Parameter => Export..." of the control

Availability

This function is enabled in

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

Structure

This dialog displays all devices/axes in the online project.

Control

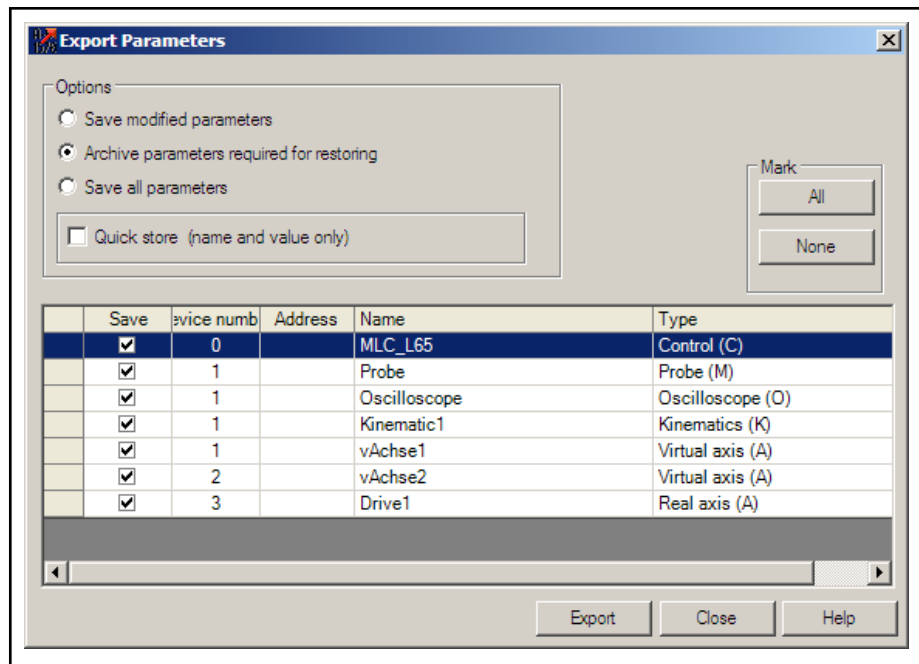


Fig.6-61: Parameter export, selecting parameters

By default, all entries are selected in the "Save" column. The device addresses of a control are shown in the "Address" column. The axis address is displayed for a drive (S- and P-parameters) or an axis (A-parameter).

"Device number" column:

The logic device number is displayed in the "Device number" column. The device address is displayed for a control, the axis address is displayed for a drive (S- and P-parameters) or an axis (A-parameter).

"Address" column:

The sercos address is displayed in the "Address" column for a drive (S- and P-parameters) or an axis (A-parameters).

"Name" column:

Displays the device name or the axis name.

"Type" column:

Displays the device type or the axis type.

It can be selected or deselected via the buttons **All** or **None**.

Export options

The export can be modified using the following options:

1. "Save modified parameters"
All parameters differing from the default value (loading according to basic parameters) are saved.
2. "Archive parameters for restoring"
Saves all parameters that can be written again to the respective object.
3. "Save all parameters"
Saves the complete data record of the object, even the parameter data that cannot be written again such as actual values.
4. "Quick store"

Control

Only name, value and attribute as well as placeholder for min./max. values and unit are saved.

The **Export** button starts the parameter export.

Parameters - Import

Saved parameters can be reloaded to the control or drives using the parameter import.

The import function can be called via the context menu of the control.

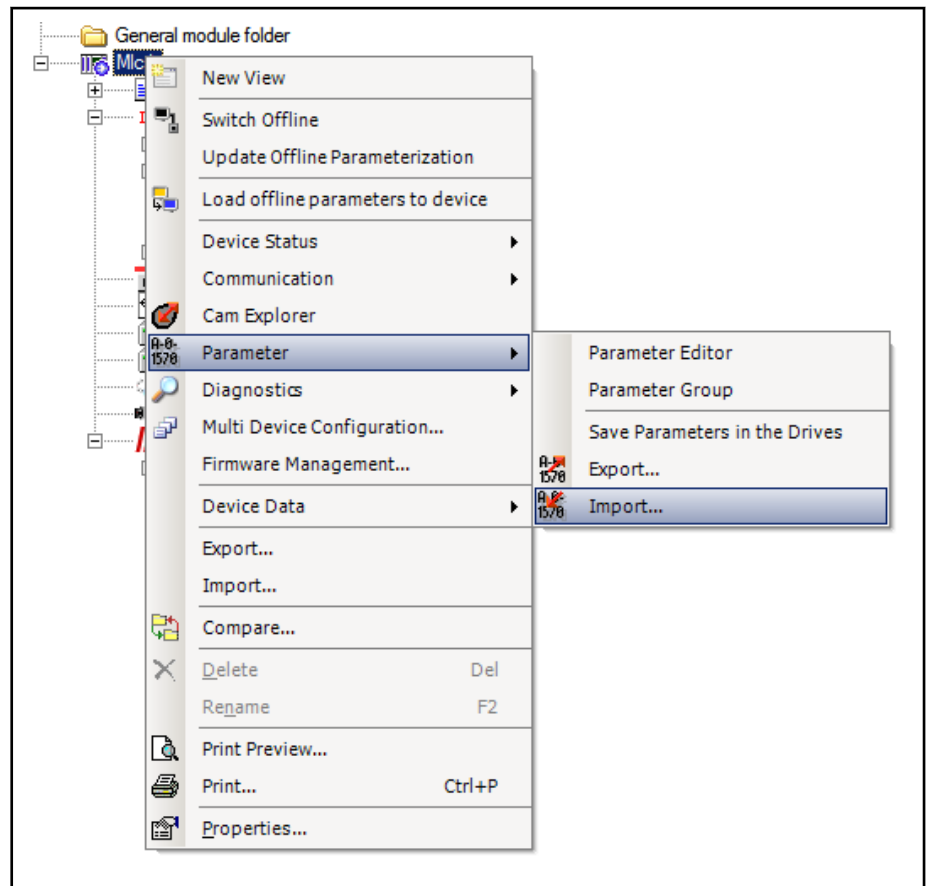


Fig. 6-62: Context menu items "Import Parameters" of the control

Availability This function is enabled in

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

Structure Settings for the parameter import can be made in the import dialog.

Control

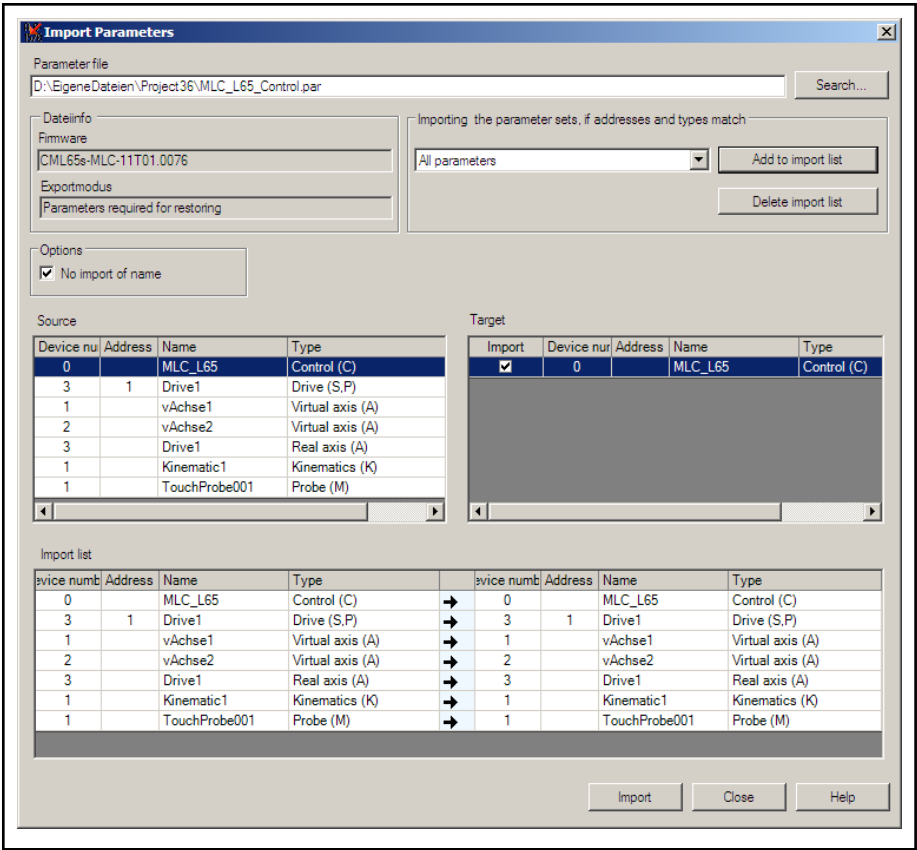


Fig. 6-63: Import settings

Parameter file:

The current parameter file previously selected when the import function was called via the "Open File" dialog is displayed.

The user has the option of loading another parameter file via **Search**.

File info:

- **Firmware**
The firmware release that was on the device at the time of export is displayed for the device selected in the source window
- **Export mode**
Provides the user with information on the export option used to save the parameters

Source:

This table lists all objects from the parameter file.

Target:

This table lists all objects that are available online with regard to the selected source line.

Import list:

By selecting one or several objects in the "Target" table, these are included in the table "Import list".



Axes of different types (e.g. from real to virtual) can also imported at once. However, only the parameters present for both types are read in.

For deactivated axes, only the A-parameters can be reached. The S-parameters and P-parameters cannot be reached.

The parameter import can be modified using the following options:

1. "Importing parameters if address and type match"

An import list considering address and type is automatically created by selecting the respective parameter(s) and clicking on the **Add to import list** button.

The complete import list can be deleted via the **Delete import list** button.

2. "No import of name"

Means that the names of the individual objects are not overwritten, e.g. for multiple instances of identical machine components in one system.

The **Import** button starts the parameter import.

6.4.7 Diagnostics

Device Status

The dialog box provides an overview on the essential properties of the current control.

Device Overview - Mlc1

Mlc1

Device Address	IP Address	Control name	Control Mode
0	10.110.241.82	Mlc1	RUN

System diagnostics: BB RUN, Motion ready, PLC in run

User name: karlwait

Firmware: CML65e-MLC-11T03.0139

System time of control: 11/16/2010 2:35:50 PM

Axis states for all axes

	Axis N.	Ad...	Type	Name	Diagnostics	Position	Motion	Status bit
bb	1	1	Real axis	RA1	Axis is in 'StandStill'	4.4573	Standstill	S-V--
	2	-	Virtual axis	VA1	Axis is in 'StandStill'	0.0000	Standstill	S--R-
	3	-	Virtual axis	VA2	Axis is in 'StandStill'	0.0000	Standstill	S--R-

Fig. 6-64: Example: "Device Status" dialog of an IndraMotion MLC

Delete all present errors of the control with the **Clear Error** button. The current status is displayed in the "System diagnostics" field.

The display shows events according to the priority of the events, i.e. if several events are present, the event with the highest priority is displayed. Therefore, the following rule applies (descending priority):

- Error class 9 ...⇒ ... Error class 0 ⇒ Warnings ⇒ Messages

Double-click on the display field of the current system diagnostics and the dialog [Error/Diagnostic Memory](#), page 144 opens.

Control

The button **More details** opens the window "Extended Properties" ([Advanced Properties, page 143](#))

The button **Set System Time** transfers the system time of the PC to the control.

The list of all defined axes provides an overview on the current status of these axes.

The axis number is displayed in the "Axis Number" column.

The "Addr" column indicates the address of the drive set in the drive ring (e.g. sercos).

In the "Status Bits" column, the most important feedback of the axis is indicated in an abbreviated form. The letters have the following meaning:

- S - At standstill
- P - In position
- V - In velocity
- R - In reference, homed
- Y - Synchronized

If the mouse pointer is moved across several lines and columns of the axis list, tooltips are displayed to provide further explanations.

Right click in the axis list and a context menu with additional functions opens:

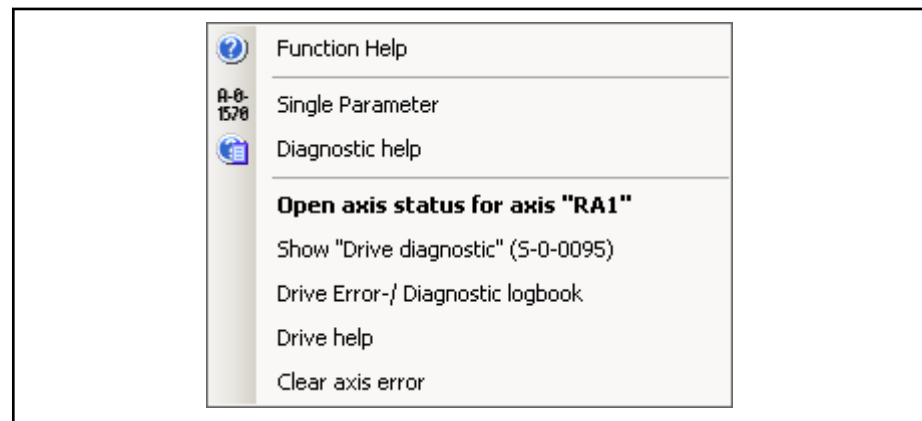


Fig.6-65:



The bold entry corresponds to the double-click on the axis.

Parameters involved:

- C-0-0010, Version number, hardware
- C-0-0012, Version number, firmware
- C-0-0023, System status
- C-0-0030, Control name
- C-0-0031, Control address
- C-0-0050, System time
- C-0-0203, Engineering, user name
- C-0-0626, System diagnostics

In the axis list, further parameters are used for each configured axis:

- A-0-0002, Axis name
- A-0-0020, Axis diagnostic message

- A-0-0021, Axis status
- A-0-0022, Extended axis status
- A-0-0100, Actual position value
- S-0-0095, Drive diagnostics (only for real axes and encoder axes)
- S-0-0390, Diagnostic number (only for real axes and encoder axes)

Advanced Properties

The "Advanced Properties" dialog contains additional details on the hardware and firmware of the control. This data is particularly required for the support:

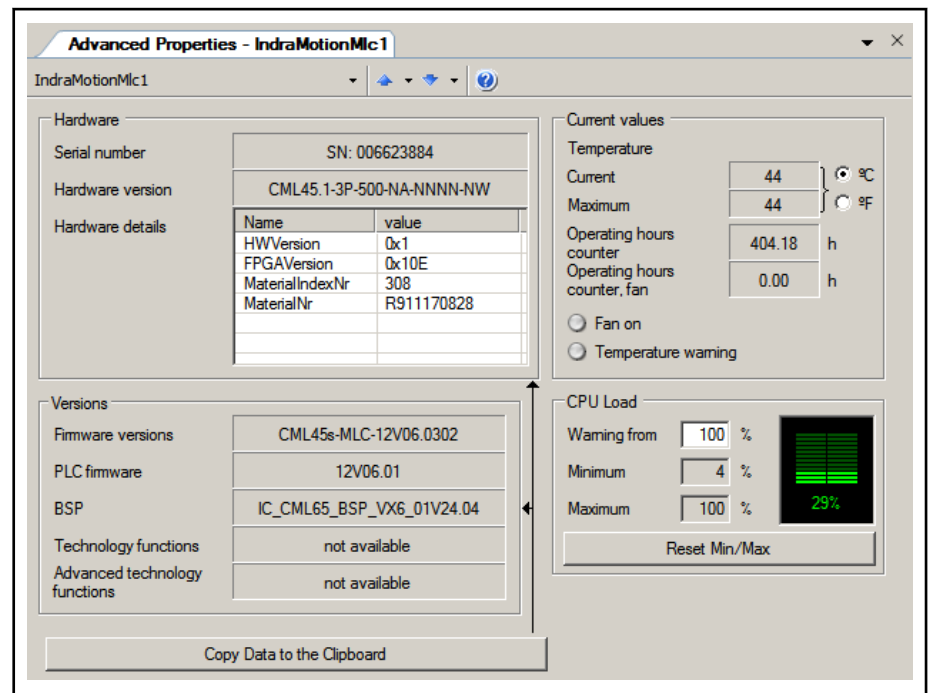


Fig. 6-66: Example: "Properties" dialog of an IndraMotion MLC

In the "Current Values" range, some additional state values of the control are displayed. The temperature warning is switched on when 60 °C is reached and switched off again if the temperature falls below 55 °C. The "Maximum temperature" value refers to the measuring period since the last reset or the last switch-on of the control.

Parameters involved

- C-0-0010, Version number, hardware
- C-0-0011, Version number, PLC firmware
- C-0-0012, Version number, firmware
- C-0-0013, Version number, BSP
- C-0-0015, Version number, technology functions
- C-0-0016, Version number, extended technology functions
- C-0-0021, Hardware, details
- C-0-0023, System status
- C-0-0032, Serial number of the control
- C-0-0051, Operating hour counter
- C-0-0415, CPU load, current
- C-0-0416, CPU load, maximum

Control

- C-0-0417, CPU load, minimum
- C-0-0418, CPU load, threshold value for warning
- C-0-1051, Command: Reset Parameter C-0-0416 and C-0-0417

Clearing Errors

The command "C-0-1030, Clear all control errors" is executed. All errors in the drives are also deleted.



Warnings of the control are also deleted if this command is executed.

Clear Error can be opened from different locations:

- In the context menu of the control via **Clear Diagnostics/Error**
- In the dialog [Device Status, page 141](#)
- In the dialog [Error/Diagnostic Memory, page 144](#) and
- In the [Toolbar of the Control, page 94](#)

Error/Diagnostic Memory

The "Error/Diagnostic Memory" dialog box provides an overview on the error states of the control and its drives. It analyzes the logbook of the control. It is accessed via the parameters described below. Error memory/logbook are organized as batch. The current status is shown on top. It has the highest index (top line, see [fig. 6-68 "Startup, error-free" on page 145](#)).

A maximum of the latest 1000 messages is saved in the control. These messages are also not lost after power failure.

Due to the high number, the display can be limited. Possible settings: 10/20/50/all.

A differentiation is made between errors, warnings and messages.

In case of errors and warnings, incoming and outgoing is stored. Messages are only present once in the logbook.

Due to reasons of storage location, the status messages following e.g. on an error deletion are not always indicated/registered completely if the deletion to **BB RUN** is carried out normally.

It can be additionally set how long the messages (e.g. A0110002 Axis was created) are shown on the display of the control. This setting does not refer to the display period of warnings and errors. These are displayed as long as they are deleted.

Categories	Remark
 Message	Status information of type "Message"
 Error	Error comes in, error reaction is triggered, error is active
 Error	Error passive, error cause still exists, but the error was reset
 Error	Error goes out, the error cause does not exist any longer and the error was reset

Categories	Remark
 Warning	Warning comes in, the warning reaction is triggered, the warning is active
 Warning	Warning passive, the warning cause still exists, the warning was reset
 Warning	Warning goes out, warning cause does not exist any longer and the warning was reset

Fig.6-67: Status information

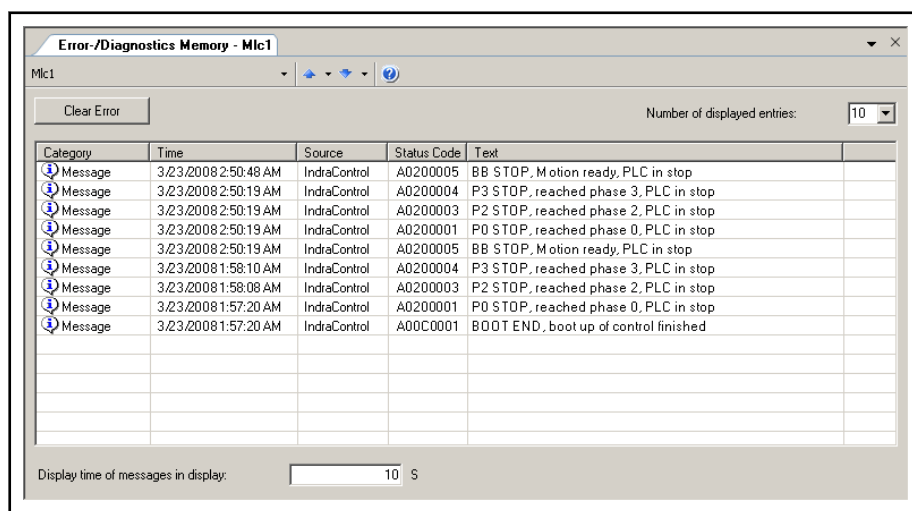
In addition to the category, the time of the log on/off, the source - whether it is information received from the control or from an axis -, the status code and its explanatory text are also indicated.



The status code and the text are documented in the "Notes on Troubleshooting" or in the online help.

Error-free startup

The work with the diagnostics/error memory and the interpretation of its display are to be considered first for an error-free startup:



Category	Time	Source	Status Code	Text
Message	3/23/2008 2:50:48 AM	IndraControl	A0200005	BB STOP, Motion ready, PLC in stop
Message	3/23/2008 2:50:19 AM	IndraControl	A0200004	P3 STOP, reached phase 3, PLC in stop
Message	3/23/2008 2:50:19 AM	IndraControl	A0200003	P2 STOP, reached phase 2, PLC in stop
Message	3/23/2008 2:50:19 AM	IndraControl	A0200001	P0 STOP, reached phase 0, PLC in stop
Message	3/23/2008 2:50:19 AM	IndraControl	A0200005	BB STOP, Motion ready, PLC in stop
Message	3/23/2008 1:58:10 AM	IndraControl	A0200004	P3 STOP, reached phase 3, PLC in stop
Message	3/23/2008 1:58:08 AM	IndraControl	A0200003	P2 STOP, reached phase 2, PLC in stop
Message	3/23/2008 1:57:20 AM	IndraControl	A0200001	P0 STOP, reached phase 0, PLC in stop
Message	3/23/2008 1:57:20 AM	IndraControl	A00C0001	BOOT END, boot up of control finished

Fig.6-68: Startup, error-free

The startup is documented starting with the bottom line:

"Message **BOOT END**, boot up of application finished".

The control runs automatically up to **BB STOP** (5th line from the bottom).

The user switches the control from offline to online. The control runs up to **P2 STOP** (7th line from the bottom). After booting, parameterizations are common. However, this is not necessary now, thus proceed in the startup from **P2** to **BB**. The control runs to **BB STOP**.

Diagnostics in case of an error

"F4034, E-Stop of the Drive of Axis 2" is considered as error.



In the "E-Stop Function" dialog, the "Analysis via Digital Input" has to be enabled and the "Interpretation as an Error Message F4034" has to be selected.

Control

The drive provides a function used to monitor an E-Stop input (connection of an external hardware switch). The axis is set to standstill as described in "P-0-0119, Optimum standstill procedure".

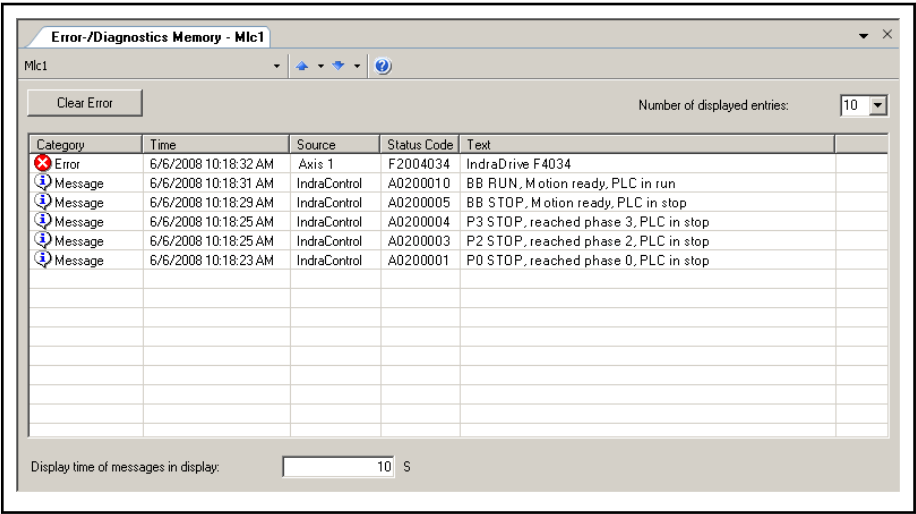


Fig.6-69: Error F4034 comes in

If the source of the warning or the error is located in a drive/its supply, its original error (warning) is indicated in the text. The status code consists of the original number, supplemented by the group information "Drive": F4034 -> F200**4034**.

In the toolbar, an active error is indicated. The axis reporting the error is highlighted in the Project Explorer. If the error affects a drive directly, the latter can - subject to the severity of the error - be set to a standstill according to the chosen strategy.

Double-click on the white cross on red background to activate the diagnostic memory and error memory window if it is not opened in the foreground.

The drive display indicates: **2 F4034**,
 sercos address **2**, error **F4034**. Its signals that the E-Stop input has responded.

The control display indicates: **F2004034**, axis 2. For the control, this is an axis error of severity "F2" which disables the individual axis.

The diagnostic and error memory window combines both information, see above.

Acknowledging the error Remove the malfunction which has triggered the E-Stop and delete the error (Esc key at the control display or error deletion command). Switch on the power supply again and clarify why the E-Stop was triggered.

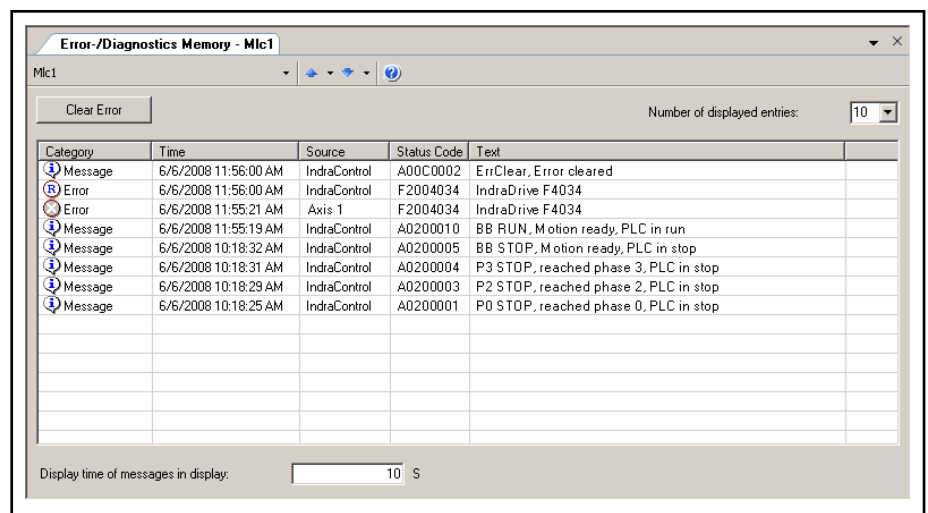


Fig. 6-70: Acknowledging the error after eliminating the cause

Acknowledging the error on the drive triggers the first three diagnostic lines.

The diagnostic messages up to **BB RUN** are not required if the procedure runs normally.

Parameters involved:

- C-0-0620, Diagnostic memory index
- C-0-0621, Diagnostics, indexed message
- C-0-0622, Diagnostics, number of messages in the memory
- C-0-0623, Diagnostics, last reset message
- C-0-0624, Diagnostics, absolute indices
- C-0-0625, Diagnostics, display message
- C-0-0660, Display, display period of messages
- C-0-1030, Command: Clear all control errors



The specified parameters should not be used directly. They were developed for the handshake with this dialog. Use the following "public" parameters instead if necessary:

- C-0-0626, Diagnostics
- C-0-0627, Diagnostic number

Invalid Parameters

The dialog box displays the current system diagnostics of the control and lists all invalid parameters:

- For the control, the parameters included in "C-0-0111, List of all invalid C-parameters"
- For each axis the parameters included in "A-0-0011, List of all invalid A-parameters", or "A-0-0014, List of non-transmitted A-parameters"
- For real axes and encoder axes, the parameters included in "S-0-0021, IDN list of invalid operating data for phase 2" and "S-0-0022, IDN list of invalid operating data for phase 3" are also evaluated

The use of the dialog is explained by means of an example:

The "positive velocity limit value" is to be increased. In doing so, the maximum admissible input value is exceeded.

Control

The user is in the operating mode **BB**. The input is immediately transferred to the drive that responds with an error message.

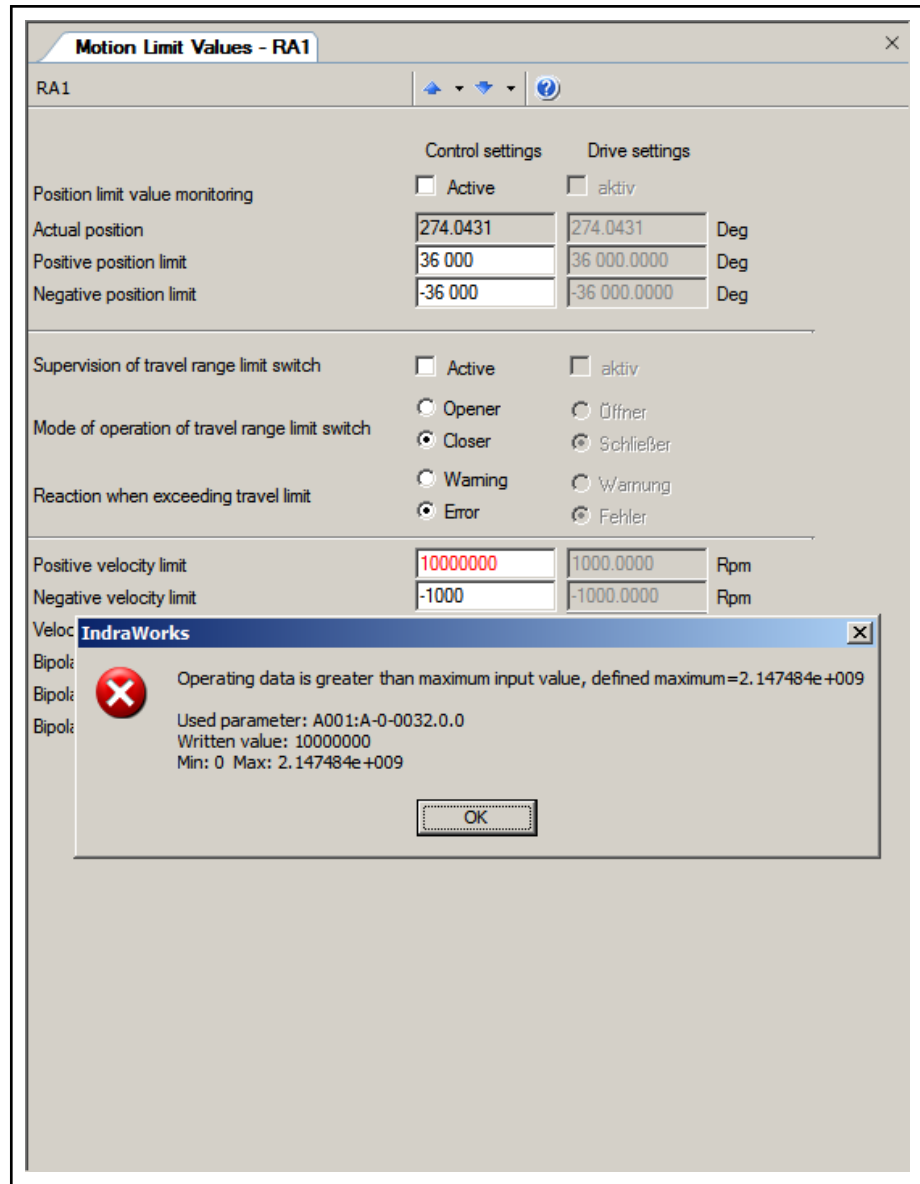


Fig.6-71: Incorrect entry, value not within the valid value range

If the change is made in the parameterization mode **P2**, the transfer does not take place before switching from **P2** to **BB**. Since several incorrect entries might have been made, it is advisable to list all these errors. Therefore, the "Invalid Parameters" window is automatically opened.

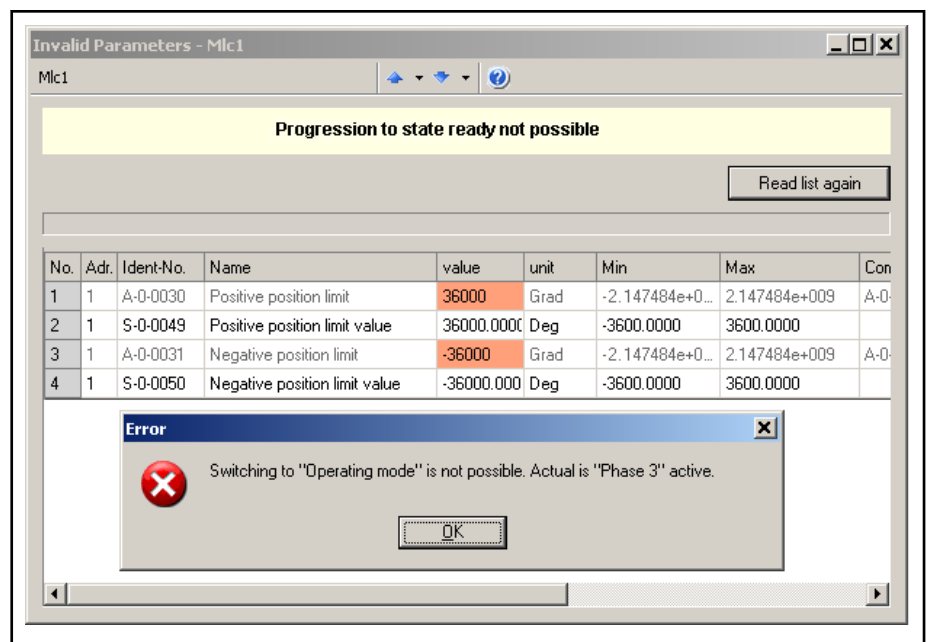


Fig. 6-72: Invalid parameters when switching from **P2** to **BB**.

It is recommended to make the appropriate correction(s) - here in the lines 1 and 3 (position limit values) - and to repeat the attempt to switch from **P2** to **BB**.



Frequently, lines are listed which seem to be identical at first glance. This is due to the fact that the S- and P-parameters determined as invalid are often overwritten with A-parameters. In this case, it is necessary to write on the corresponding A-parameter. Writing on the S- or P-parameter has no effect. Therefore, the parameters to be changed are highlighted in red. In the comment column, this is described for each case.

Parameters involved

- C-0-0111, List of all invalid C-parameters
- C-0-0626, Diagnostics

and the parameters required for each axis

- A-0-0011, List of all invalid A-parameters
- A-0-0014, List of non-transferred A-parameters

and the parameters required for each real axis and for each encoder axis

- S-0-0021, IDN list of invalid operating data for phase 2
- S-0-0022, IDN list of invalid operating data for phase 3
- S-0-0390, Diagnostic number

Task List/Configuration

The data of all tasks can be seen in the task list. The task list can also be used to prioritize the tasks.

This dialog is described in [chapter 7.4.6 "Task List/Configuration"](#) on page 210.

Control

Task Execution Viewer

The Task Execution Viewer displays the activity of tasks in a task-time diagram

This dialog is described in [chapter 7.4.7 "Task Execution Viewer" on page 212](#).

6.4.8 Multi-Device

Introduction

For "big machines" with several machine modules of the same type, the controls often have similar tasks within the complete machine. Apart from identical or comparable hardware connected to one single control, the PLC programs executed on the control are often identical except for the initial values minus the PLC variables – or they can theoretically be replaced by exactly one PLC program with control-specific initial values. The configuration of these controls is often very similar.

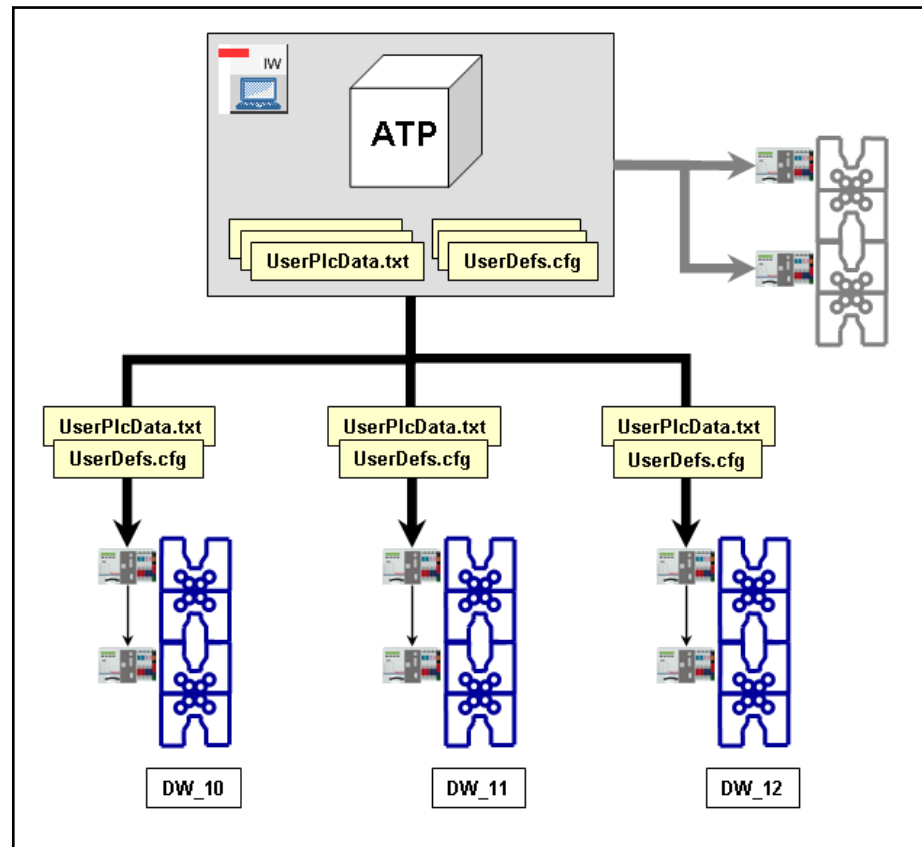


Fig. 6-73: An IndraWorks project ("ATP") downloaded together with device-specific files to several controls.

To use an application and their parameters for several physical controls, further controls can be added to the IndraWorks project in IndraWorks via the command "Multi-Device Configuration" of the control node and the **multi-device table** described in the following. These controls are contained in the multi-device table and can be individually selected and enabled to go online (Motion) or to log in (PLC).

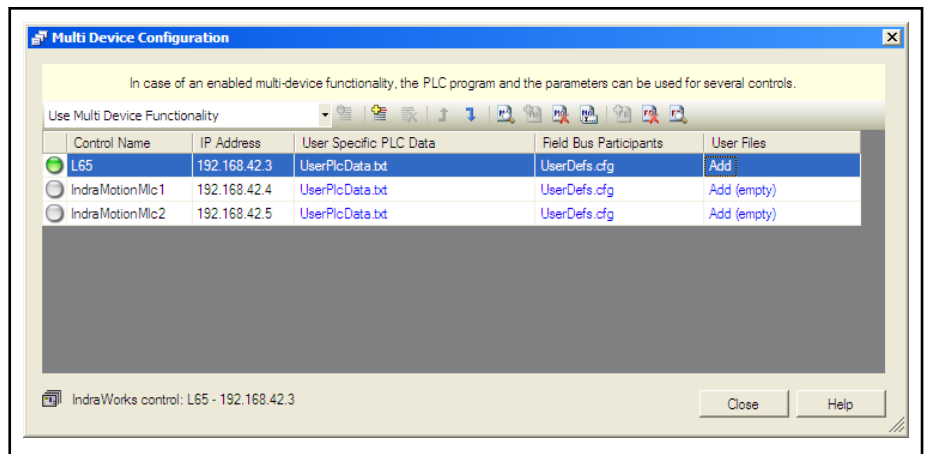


Fig.6-74: The "Multi-Device Configuration" dialog with the activated multi-device functionality and the multi-device table

Brief Description of the Multi-Device Table

Opening the multi-device table

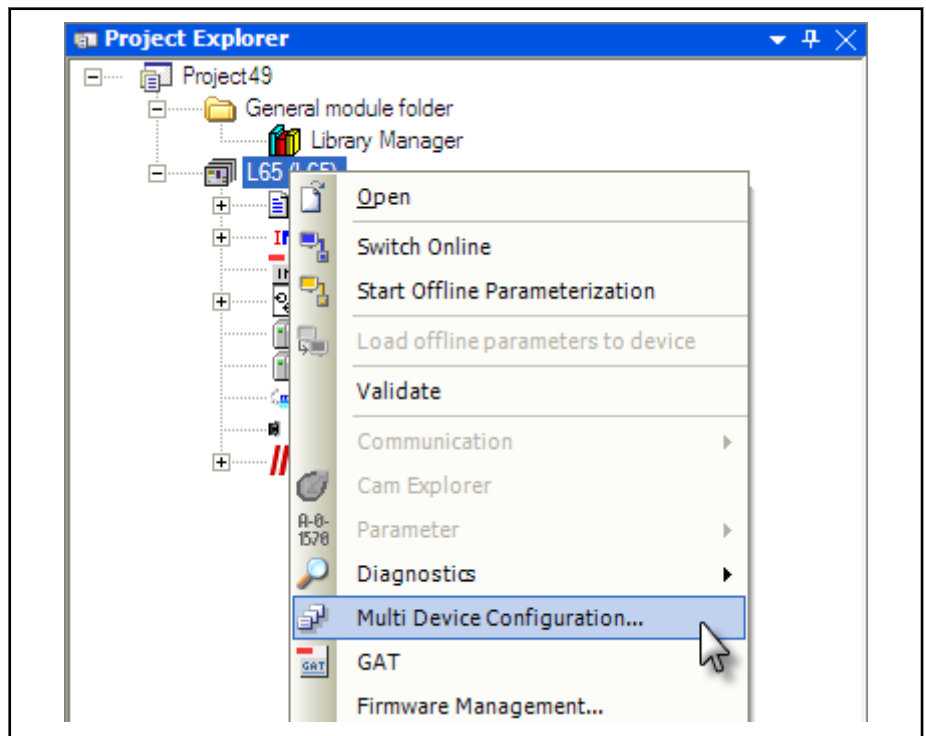


Fig.6-75: Opening the dialog with the multi-device table

In IndraWorks, exactly one multi-device table can be assigned to each control node (XLC/MLC) and edited with the help of the "Multi-Device Configuration" dialog. This dialog is opened via the context menu of the control (right-click on the control node in the Project Explorer). Click on "Multi-Device Configuration" and the dialog with the identical name opens with the multi-device table. To edit the table, the IndraWorks project may neither be online (Motion) nor logged in (PLC). Furthermore, the multi-device functionality has to be enabled in the dialog "Multi-Device Configuration".

Control

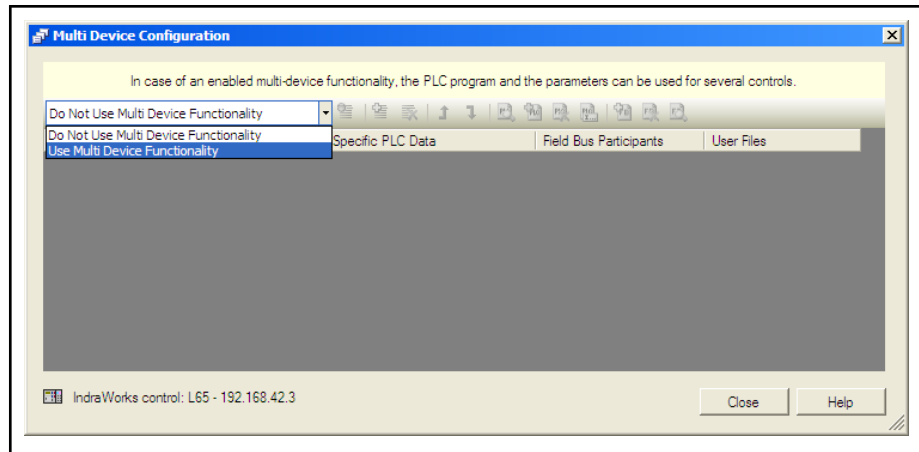
Activating/deactivating the multi-device functionality

Fig.6-76: Activating the multi-device functionality

Using the combo box in the toolbar of the "Multi-Device Configuration" dialog, the multi-device functionality can be activated or deactivated ("Use multi-device functionality" or "Do not use multi-device functionality").

As long as the multi-device functionality has not been activated, the multi-device table is empty. When the multi-device functionality is activated for the first time, a device with the name and IP address of the current control is added once to the multi-device table. If the multi-device table has been activated once, it always contains at least one device.

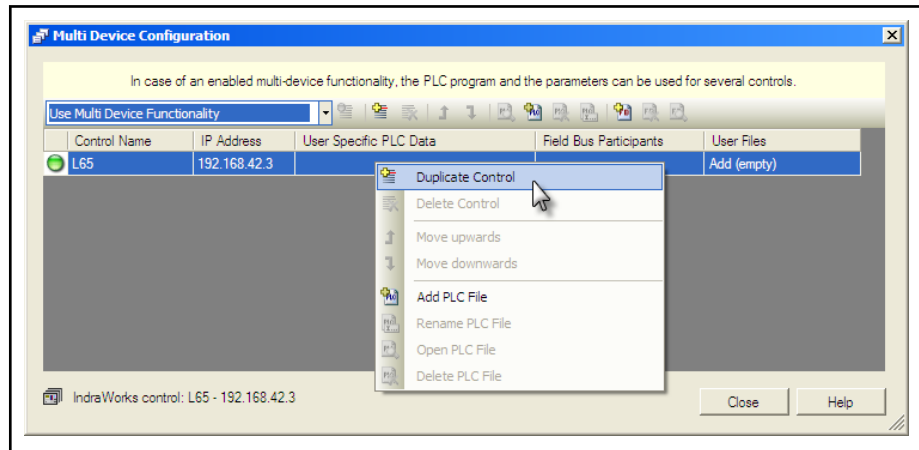
Adding/deleting more controls

Fig.6-77: Adding another control based on the selected control (duplicating)

More controls with individual device names, individual IP address, etc. can be added to the multi-device table via the context menu (right-click). Of all the controls in the multi-device table, only one control is active at a time. This control is marked by a green LED. Its properties are used in IndraWorks.

Apart from adding controls, they can also be deleted from the multi-device table. The active device cannot be deleted. Thus, if the multi-device table has been activated once, it contains at least one device.

Multi-device Functionality in Detail**Version Control System (VCS)**

The data of the multi-device table is part of the "Control" node in IndraWorks. The following results: If the IndraWorks project including the "Control" node is version-controlled, the data in the multi-device table is also version-controlled.

Difference: Multi-Device Deactivated ↔ Activated

Multi-device deactivated

If the multi-device functionality is disabled, the IP address of the control is not version-controlled. That means that if the IndraWorks project is under version control and the multi-device functionality is disabled, the IP address of the control can be changed without checking out the control.

Furthermore, if the multi-device functionality is deactivated when logging in to the PLC, no "user-specific PLC data" and no "user data" are transferred.

Multi-device activated

If the multi-device functionality is activated, the IP address of the control is part of the multi-device table data. Since the multi-device table as part of control data is under version control, the IP address is also version-controlled. This means that if the IndraWorks project is under version control and the IP address of the control is to be changed, the control node has to be checked out.

Furthermore, the following is transferred to the Compact Flash card of the control ("User partition", directory "/ata0b") at each login to the PLC

- User-specific PLC data file (if available)
- Configuration file to disable the Profibus DP slaves (if available)
- user files (if available)

Additionally, the following parameter is write-protected during a parameter import:

- C-0-0030, "Control name"

If a function module is used for the cross communication, the following parameter is additionally write-protected:

- C-0-0031, "Control address"

If the function module used for the cross communication is of type "Cross-Comm sercos III (CFL01.1-R3)", the following parameters are also write-protected:

- C-0-0700, "Link - cycle time (Tcyc) command value"
- C-0-0701, "Link - configuration function modes"
- C-0-0778, "Command configuration of the link"

Basic Behavior of the Multi-Device Table

Editing the table

For editing, click or double-click on the cells of the multi-device table.

If there is an invalid entry within a cell, a small exclamation mark is displayed in the cell with the incorrect entry after leaving the cell. Only if there is no error within a line, the data of this line is internally applied in IndraWorks.

Control

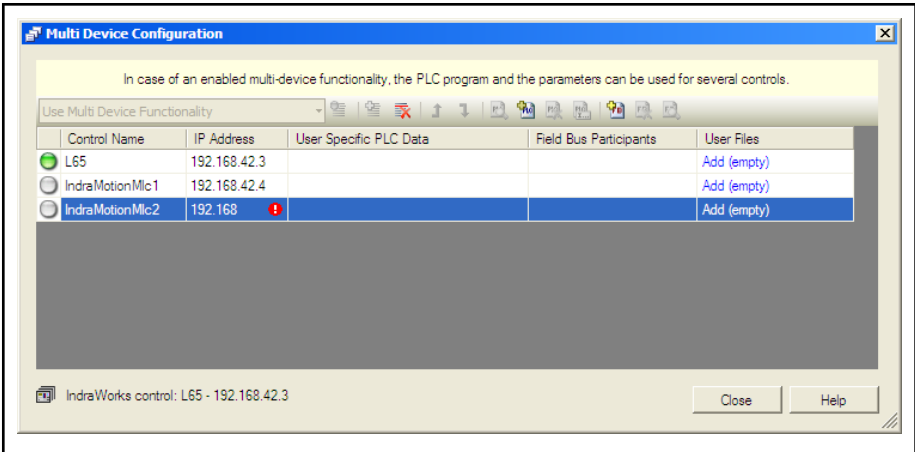


Fig.6-78: Note on invalid entry

Commands via toolbar or context menu

The commands of a multi-device table like adding a new device for example, shifting a line within the table etc., can be called via the toolbar buttons of the multi-device dialog or via the context menu of the table.

The functionality of the individual columns of the multi-device table is described in detail in the following.

Describing Columns of the Multi-Device Table

Column "Control Name"

IEC-compliant control name

This column contains the control names. Only IEC-compliant names are accepted. The control name within a multi-device table has to be unique.

Column "IP Address"

This column contains the IP addresses of the individual controls. In contrast to the control names, the same IP address can be used multiple times. Thus, it is for example possible to use different user-defined data files (see [chapter "Column User-Specific PLC Data" on page 155](#)) for one physical device.

Column "Addr."

sercos III cross communication, device address

This column is only visible, as the "Function" column, if a function module of type "CrossComm sercos III (CFL01.1-R3)" is used for the cross communication of different controls among each other. The cells of the column contain the cross communication address of the respective control. The address can be directly changed in the cell.

The command "sercos III cross communication" can also be called via the context menu of this column. This command can open the "Properties" dialog of the "CrossComm sercos III (CFL01.1-R3)" function module.

Column "Function"

sercos III cross communication, master/slave

This column is only visible, as the "Addr." column, if a function module of type "CrossComm sercos III (CFL01.1-R3)" is used for the cross communication of different controls among each other. These cells specify whether the current control responds like a cross communications master or slave.

The command "sercos III cross communication" can be called via the context menu of this column. This command can open the "Properties" dialog of the "CrossComm sercos III (CFL01.1-R3)" function module.

Column "User-Specific PLC Data"

A PLC program - n user-specific PLC data files

All controls in the multi-device table use the same PLC program. Once compiled, the program can be downloaded to the respective active control if the multi-device functionality is active when logging in to the PLC.

If specific data - such as initialization values - is to be used for the individual controls within the multi-device table, this data can be provided using the user-specific PLC data file. Exactly one individual user-specific PLC data file can be assigned to each control within the multi-device table. The task of the PLC program is to analyze this user-specific PLC data file at runtime. The XLC or MLC firmware or IndraWorks do not analyze the user-specific PLC data files.

"UserPlcData.txt"

The user-specific PLC data file is added via the context menu of this column. The default file name is "UserPlcData.txt". The name of the file can be changed or the file can be deleted via the context menu. The user-specific PLC file data is saved internally in IndraWorks in different directories depending on the device. Thus, the same file name can be used multiple times in the multi-device table. The file opens with the editor registered for text files on the target system when clicking on the file name in the cell.

Transfer at login to the PLC

The user-specific PLC data file is transferred to the Compact Flash card during each login to the PLC ("user partition" "/ata0b") irrespective of whether the PLC program is loaded or an online change is executed. An existing file of the same name is overwritten. The file is not transferred if the login is aborted in IndraWorks by the user.



The user-specific PLC data file on the Compact Flash card of the control is never actively deleted. This also applies if the file is renamed or deleted in the multi-device table and thus in the IndraWorks project.

Column "Field Bus Devices"

Same hardware configuration, exception: sercos drives and Profibus DP slaves

Devices connected to or used by the controls in the multi-device table have to be identical to the configured devices with regard to type, number and eventually the plug position. This applies especially to Inline I/Os, function modules, field bus interfaces, etc.

The drives at the sercos bus and the Profibus DP slaves are an exception. A project with the maximum number of features can be used for them. That means that the maximum stage of expansion can be implemented in IndraWorks.

Configured, but not physically existing sercos drives are automatically detected when switching to online mode in IndraWorks and can be permanently deactivated by the user if requested once.

"UserDefs.cfg"

Projected, but not physically existing Profibus DP slaves can be individually deactivated for each control in the multi-device table during the PLC startup using the configuration file "UserDefs.cfg". This file is automatically read and analyzed by the runtime of the PLC when the PLC program is started.

To further illustrate this, see the two following screenshots of the IndraWorks interface.

Control

Project Planning and Configuration

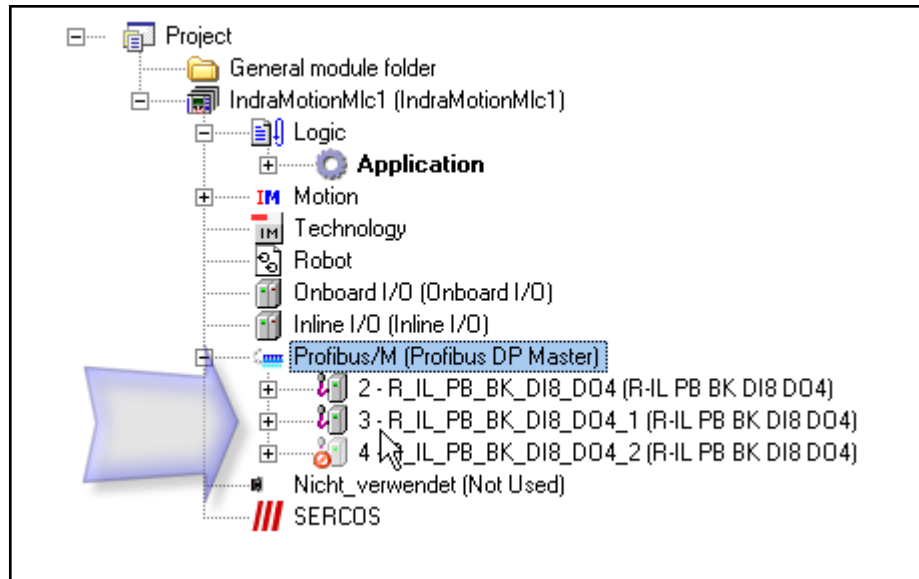


Fig.6-79: Projected Profibus DP slaves

It is logged off, the multi-device is activated and the Profibus DP slave 4 is deactivated in the IndraWorks project. The slaves 2 and 3 are configured as active, but are to be deactivated during the PLC startup via the configuration file "UserDefs.cfg".

After login and PLC startup

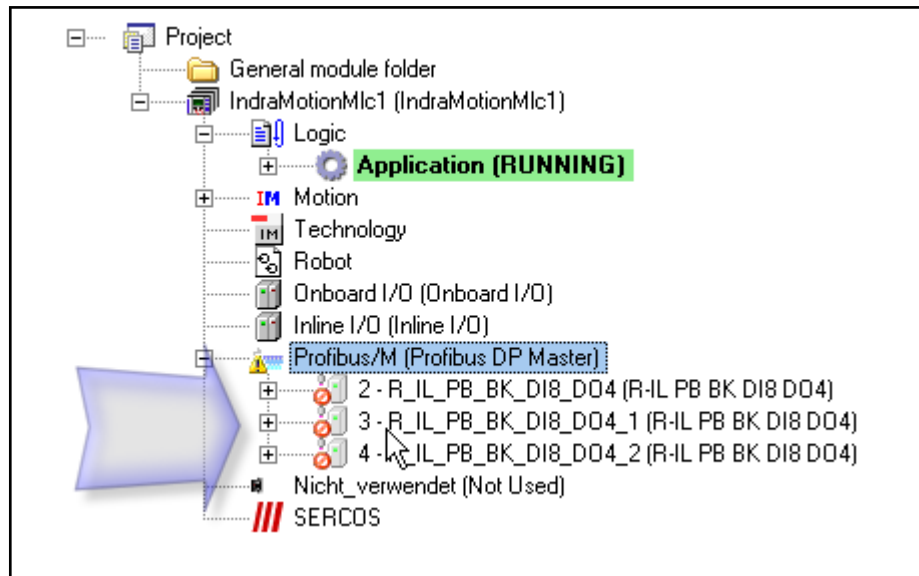


Fig.6-80: Projected Profibus DP slaves disabled

It is logged in and the PLC program was started. Profibus DP slave 4 remains deactivated, but the slaves 2 and 3 were automatically deactivated after the PLC startup and the analysis of the "UserDefs.cfg" file.

Adding "UserDefs.cfg"

The configuration file for a selective deactivation of Profibus DP slaves ("UserDefs.cfg") can be added to each control within the multi-device table via the context menu of the "Field Bus Devices" column. The configuration files are stored in different directories in the IndraWorks project and thus independent from each other. As it is the case for the user-specific PLC data file, an individual configuration file can be deleted. Click on the file name to edit the file with the standard editor for text files.

Transfer at login The configuration file to deactivate Profibus DP slaves is transferred as it is the case for the user-specific PLC data file at each login to the PLC ("user partition" "/ata0b") irrespective of whether the PLC program is loaded or an online change is executed. An existing file of the same name is overwritten. The file is not transferred if the login is aborted.

- Differences** Differences for a user-specific PLC data file:
- The configuration file to deactivate Profibus DP slaves cannot be re-named.
 - If a control in the multi-device table is not provided with a configuration file, a possibly existing file with the name "UserDefs.cfg" on the "user partition" ("/ata0b") of the file system might be deleted when logging in to the PLC of this control
 - If the Multi-Device functionality is deactivated, a possibly existing file with the name "UserDefs.cfg" on the CF card ("/ata0b") might also be deleted when logging in to the PLC

Column "User Files"

To operate a system, unit-specific information as files to other units (controls) such as special configuration files for field bus devices are often required. These files are named "user files" in the following. The following can be assigned to each control in the multi-device table:

- exactly one user-specific PLC data file
- exactly one configuration file "UserDefs.cfg"
- and several user files (described here)

The PLC program has to use these "user files" in a suitable form. The XLC or MLC firmware or IndraWorks do not use the user file or do not analyze the files.

Transfer at login to the directory "MD_UserFiles"

The user files are transferred to the "user partition" "/ata0b" of the Compact Flash card of the control to the directory "MD_UserFiles" at each login to the PLC. Existing files with the same name are overwritten in the directory. If the directory "MD_UserFiles" does not exist, it is created during login. The files are not transferred if the login is aborted.



The user files in the directory "MD_UserFiles" on the Compact Flash card of the control are never actively deleted by IndraWorks. This also applies if the user files in the "User Files" dialog - and thus in the IndraWorks project - are renamed or deleted.

The dialog "User Files" to add, delete,... user files

The user files are added to the IndraWorks project using the "User Files" dialog. Furthermore, user files can also be deleted, renamed or opened in this dialog. Click on a cell in the "User Files" column and the "User Files" dialog opens. Individual "user files" can be added to each device in the multi-device table.

Control

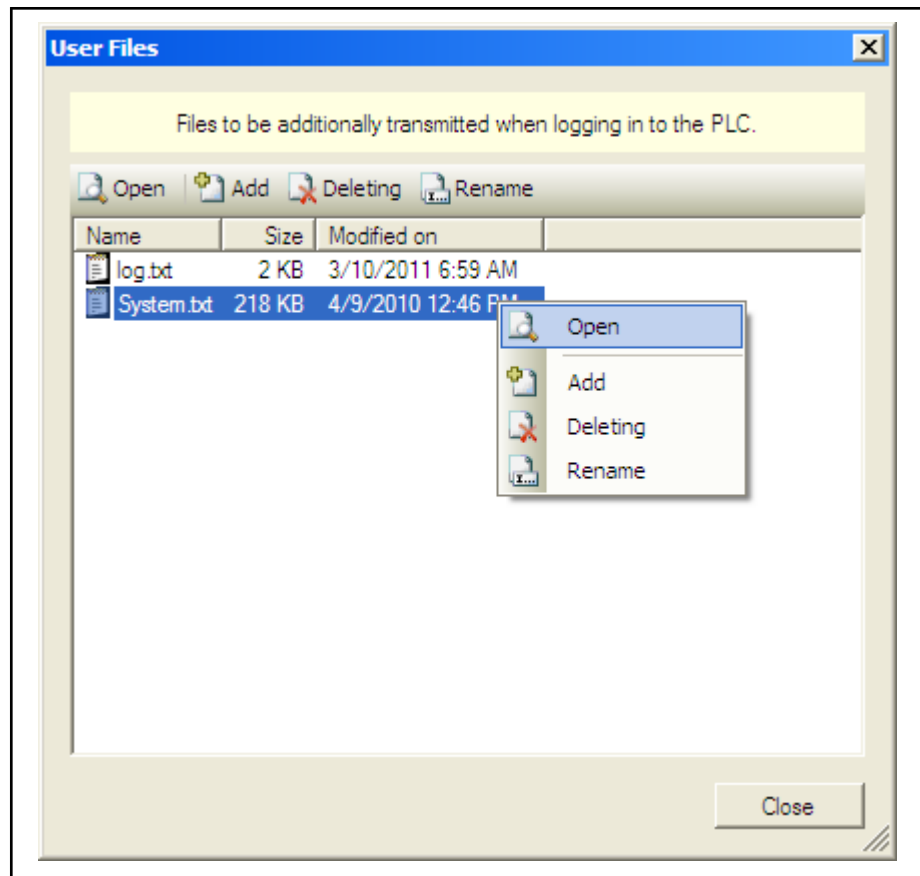


Fig. 6-81: "User Files" dialog

Configuration File to Deactivate the Profibus DP Slaves

Introduction

Deactivating configured slaves

The configuration file can be used to deactivate Profibus DP slaves configured in IndraWorks. The deactivation is carried out via a file on the Compact Flash card of the control. Thus, diagnostic messages are impeded for slaves projected in IndraWorks but that do not really exist.

Enabling the function

"UserDefs.cfg" on "/ata0b"

The function is activated due to an existing valid file "UserDefs.cfg" saved on the Compact Flash card of the control ("user partition" "/ata0b").



IndraWorks transfers and deletes the file "UserDefs.cfg" when logging in to the PLC.

When logging in to the PLC, "UserDefs.cfg" is

- ... **transferred** precisely to the Compact Flash card of the control ("user partition" "/ata0b") if the multi-device functionality of the current control is enabled and if the control marked as active in the multi-device table is provided with a "UserDefs.cfg".
- ... **deleted** precisely on the Compact Flash card of the control ("user partition" "/ata0b") if (I) the multi-device functionality of the current control is disabled or if (II) the multi-device functionality of the current control is enabled and if the control marked as active in the multi-device table is not provided with a "UserDefs.cfg".

Point in Time of Slave Deactivation

The "UserDefs.cfg" file is analyzed and the slaves entered there are disabled at the following points in time:

- Startup after power on
- Reset (warm)
- Reset (cold)
- Load program after reset (origin)

Structure of the Configuration File "UserDefs.cfg"

Elements of the Configuration File

Comments, sections as well as key/value pairs within a section can be included in a configuration file. The individual elements are explained in detail in the following.

Comments

; Comment A comment is introduced by a semicolon ";" at the beginning of a line. All characters within a comment are ignored.

Example:

```
; This is a comment
```

Sections

[Section] The configuration file is divided in one or more sections. The sections contain entries (key/pair values).

The denomination of a section is written in square brackets. A section ends with the begin of the next section or with the end of the file.

The currently valid identifier for a section are:

ProfibusDP.Master.0	First configured Profibus DP master (generally the first on-board master)
ProfibusDP.Master.1	Second configured Profibus DP master (if onboard master available, Profibus DP master on function module 1)
ProfibusDP.Master.2	...
ProfibusDP.Master.3	...
ProfibusDP.Master.4	Fifth configured Profibus DP master
ProfibusDP.Master.X	Section is not evaluated

Control

Digits 0..4 The digits 0..4 define the Profibus DP master to be used. "0" corresponds to the first configured Profibus DP master, "1" to the second Profibus DP master, etc.

Example:

-
- The onboard Profibus DP master is enabled and a Profibus DP master function module is plugged in "⇒ 0" corresponds to the onboard master, "1" corresponds to the master on the function module
 - The onboard Profibus DP master is deactivated and a Profibus DP master function module is plugged in "⇒0" corresponds to the master on the function module
 - The onboard Profibus DP master is enabled and two Profibus DP master function modules are plugged in ⇒ "0" corresponds to the onboard master, "1" corresponds to the master on the function module plugged in next to the control, "2" corresponds to the master on the Profibus DP master module that, viewed looking down on the control, is the one plugged in furthest to the left of the control.
-

Character X Due to the definition of character "X" for the index instead of a character, this section is ignored in the evaluation.

In contrast to the syntax known for example from the initialization file (.ini files), the same section name can be used multiple times.

Example:

```
; Begin of a section          (here in this example: onboard master)
[ProfibusDP.Master.0]

; Here we are inside the section

; This section is ignored
[ProfibusDP.Master.X]

; Begin of a new section  (here in this example: function module 1)
[ProfibusDP.Master.1]

; Section ends with the end of the file
```

Key/Value Pairs

Key = Value Key/value pairs can be defined within one section (key/value pairs). Key/value pairs consist of an identifier for the key, the assignment character "=" and a value as character string.

The currently valid keys within a section with the name ProfibusDP.Master.N (N = 0..4 or X) currently are:

SlaveDeactAddr Deactivates a Profibus DP slave configure in IndraWorks

In contrast to the syntax known for example from the initialization file (.ini files), the same key can be used multiple times due to a better readability within a section.

Example:

```
[ProfibusDP.Master.0]

; Deactivate slaves
SlaveDeactAddr = 2
SlaveDeactAddr = 19
SlaveDeactAddr = 104
```

Example of a "UserDefs.cfg" file

Example:

```

, *****
;
; Onboard Master
; *****

[ProfibusDP.Master.0]
SlaveDeactAddr = 5
SlaveDeactAddr = 8
SlaveDeactAddr = 25

, *****
;
; Master on function module
; *****

[ProfibusDP.Master.1]
SlaveDeactAddr = 64
SlaveDeactAddr = 30
SlaveDeactAddr = 112
SlaveDeactAddr = 3

, *****
;
; This whole section is ignored
; *****

[ProfibusDP.Master.X]
SlaveDeactAddr = 36
SlaveDeactAddr = 7
SlaveDeactAddr = 15

```

6.4.9 Firmware Management

Goal	The firmware in the control is to be exchanged without loss of data.
Prerequisites	• The control is switched on • The network settings on the control are correct. • The IndraWorks versions 04 VRS (required for an update from MLC 04VRS to MLC 12VRS) and 12VRS (higher than 12V02) are installed • IndraWorks has started and a project was loaded containing the control • The IP address of the control in the project has to match the IP address of the connected control.
Procedure	An intermediate step via firmware MLC 11VRS is required for the update of the MLC 04VRS firmware as a direct firmware update to MLC 12VRS is not supported. Note that XLC firmware variants are also provided as update option in addition to the MLC firmware. In this case, an MLC firmware version has to be selected as the functionality of the control is otherwise limited.



If the intermediate step is to be executed via a firmware 11VRS, the firmware variants XLC or MLC of 12VRS have to be taken into consideration.

Example: If the firmware XLC is to be used for 12VRS, a firmware file of version 11VRS has to be used for the intermediate step in an XLC.



Read the release notes for the new firmware first.

To be on safe side, export all parameters first (except for S- and P-parameters) (see [chapter "Parameters - Export" on page 136](#)). The control can contain old program components, parameterizations and PLC data from previous use. These are removed by clearing the memory (see [chapter 5.3.3 "Control Memory Deletion" on page 43](#)).

Control

It is assumed that the control runs in **BB RUN** with active axes.

It is recommended to follow this sequence:

1. PLC: Stop PLC, log out.
Inconsistent data caused by the PLC can be excluded.
2. Switch control offline.
3. Download the firmware via the "Firmware Management" context menu item of the control.

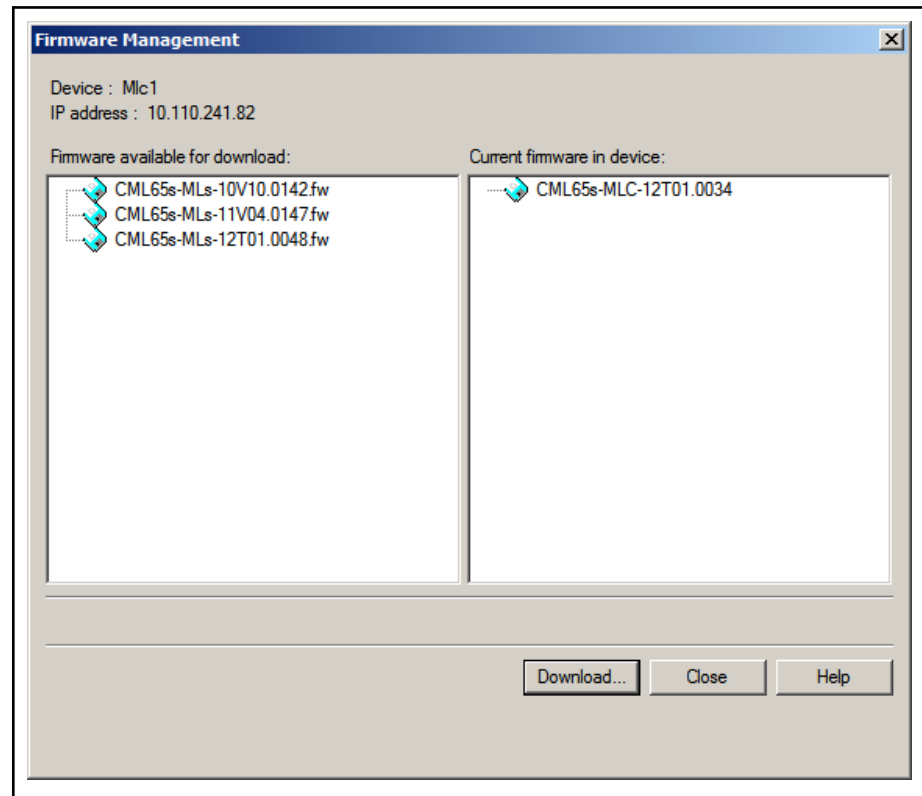


Fig.6-82: Example: IndraMotion MLC "Firmware Management" dialog

The window can be enabled in the context menu of the control.

The available firmware versions are located in the left section. Select the desired version and start the download using the **Download...** button.

The versions available for selection are located under:

<Drive>:\Programs\Rexroth\IndraWorks\<Control>\Firmware.

<Control>: "XLC" or "MLC"



The new firmware becomes only effective after the control restart.

4. The control is generally rebooted automatically. However, in some cases the control has to be restarted manually. In this case, "PWRCYCLE" is displayed on the control display. Disconnect the supply voltage for a short period and then restart the control.

After the restart, the control runs up to **P2 Stop** or **BB Stop** depending on the setting of "C-0-0450, Startup target mode Motion".

6.4.10 Device Data

Archiving

This menu item can be used to backup the control data (parameters, kinematic data) and the Compact Flash partitions of the control in the project in order to restore them on a control with the same firmware version. This function is only available if there is an online connection to the server.

Data types

There are four data types. The project has to be online for the archiving.

The following overview shows which data is included in which data type.

Data type	Control data	User partition data	OEM partition data	System partition data
Archives stored on the device			X	
Firmware				X
Control parameters	X			
Drive parameters	X			
PLC retain data				X
Compiled PLC programs			X	
Kinematic parameters	X			
RCL program transferred to the control	X			
RCL build transferred to the control	X			
Cam tables transferred to the control		X		
Cam tables loaded to the drive	X			
MotionProfiles transferred to the control		X		

Fig. 6-83: Overview on data types

Control

User interface

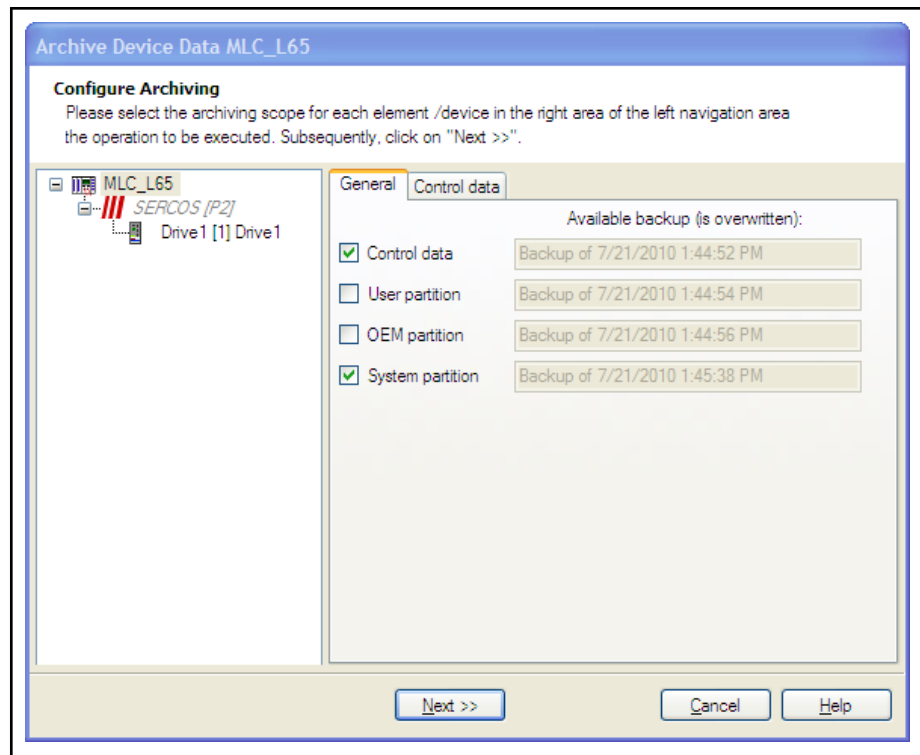


Fig.6-84: Dialog to configure the control

Specify the data type to be archived in the "General" tab. If a data type is already in the project, the archiving date is displayed in the "Available backup" area.



If a data type is already in the project, it is overwritten during archiving.

If a backup of the control data was selected, it is possible to set the parameters to be archived and whether the kinematic data is to be archived in the "Control data" tab. The "Other data" group is only available if at least one kinematics is configured in the project.

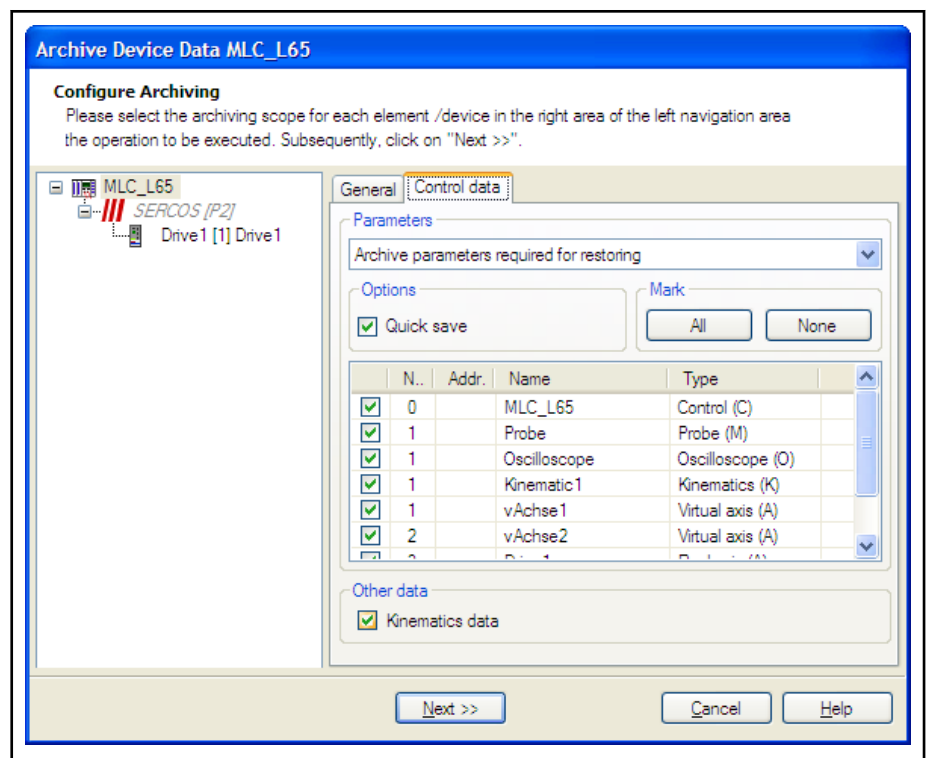


Fig.6-85: Dialog to select the control data to be archived
Data to be archived is subsequently displayed for confirmation purposes.

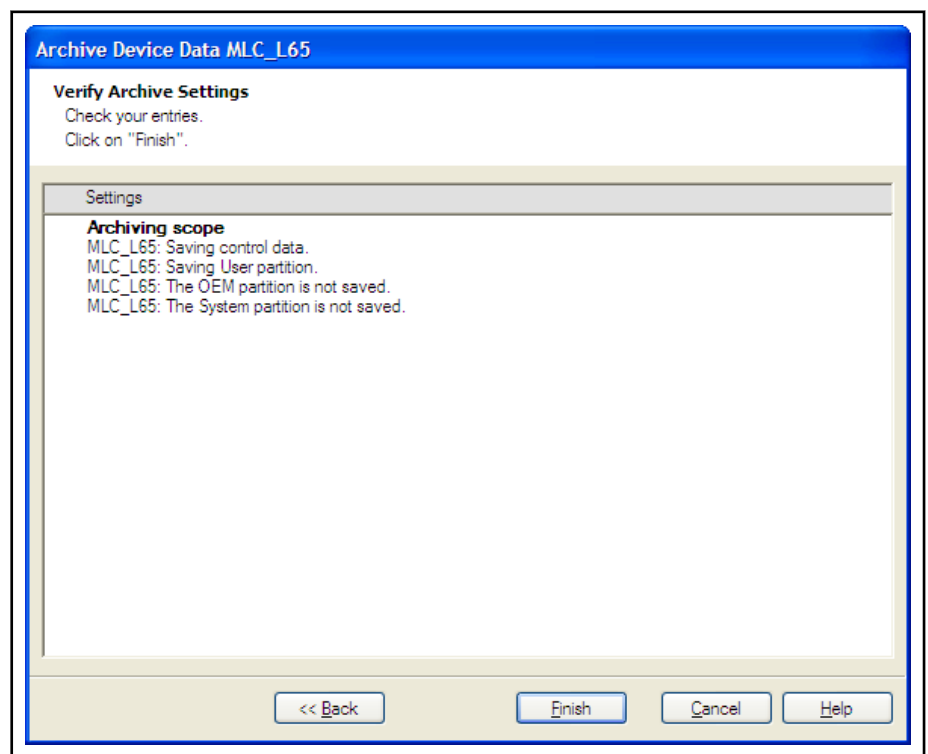


Fig.6-86: Check user inputs
Press **Finish** to start the archiving. The total progress is then displayed.

Control

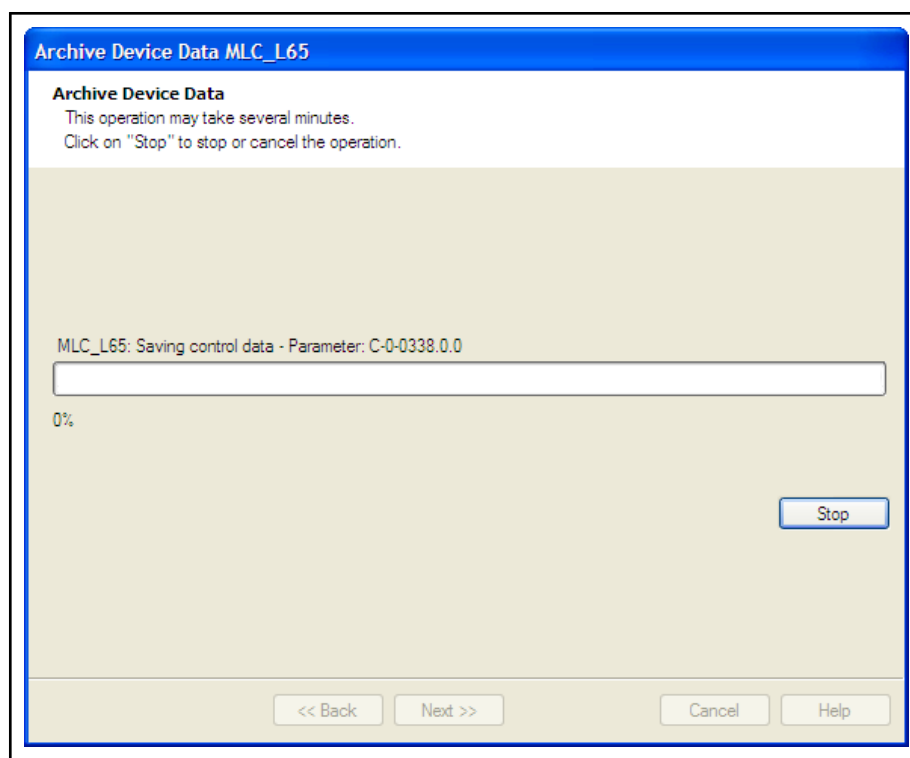


Fig.6-87: Progress bar of the archiving process

At the end, a summary is output containing the archiving results.

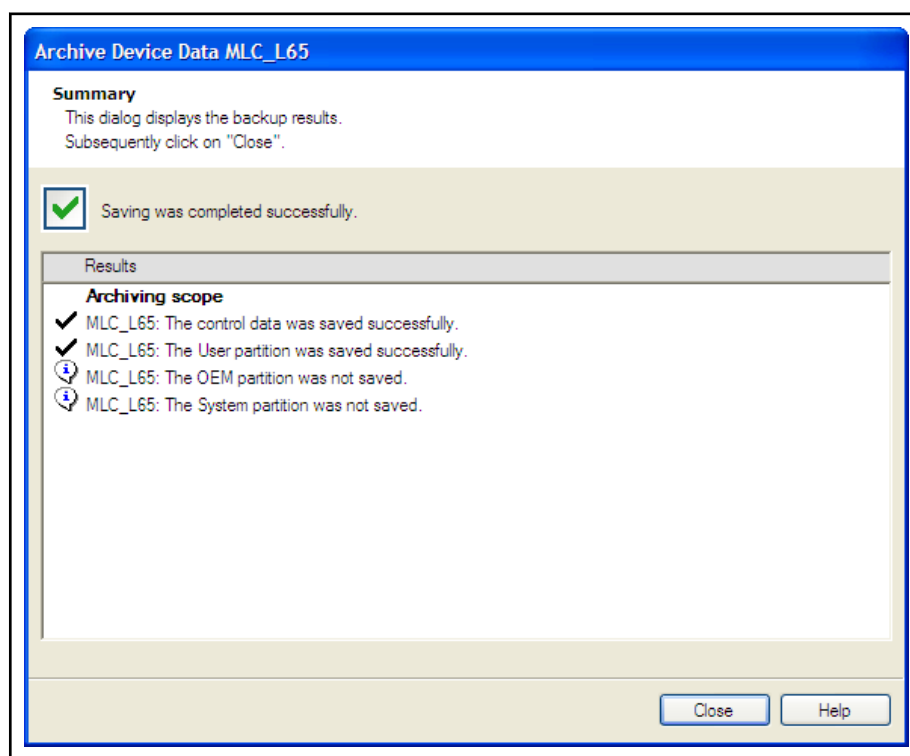


Fig.6-88: Summary of the archived device data

Complete the archiving dialog with **Close**.

Storage location The archived data is located under:

- <ProjectDirectory>\<NameOfTheControl>\ArchivedData

The individual files are named as follows:

- Parameters: Params.par
- Kinematic data: RoCo_Data.zip
- User partition data: CF_User.zip
- OEM partition data: CF_OEM.zip
- System partition data: CF_System.zip

Restoring

User interface

This menu item allows the restoration of control data (parameters, kinematic data) and Compact Flash partitions on the control. This function is only available if there is an online connection to the server.

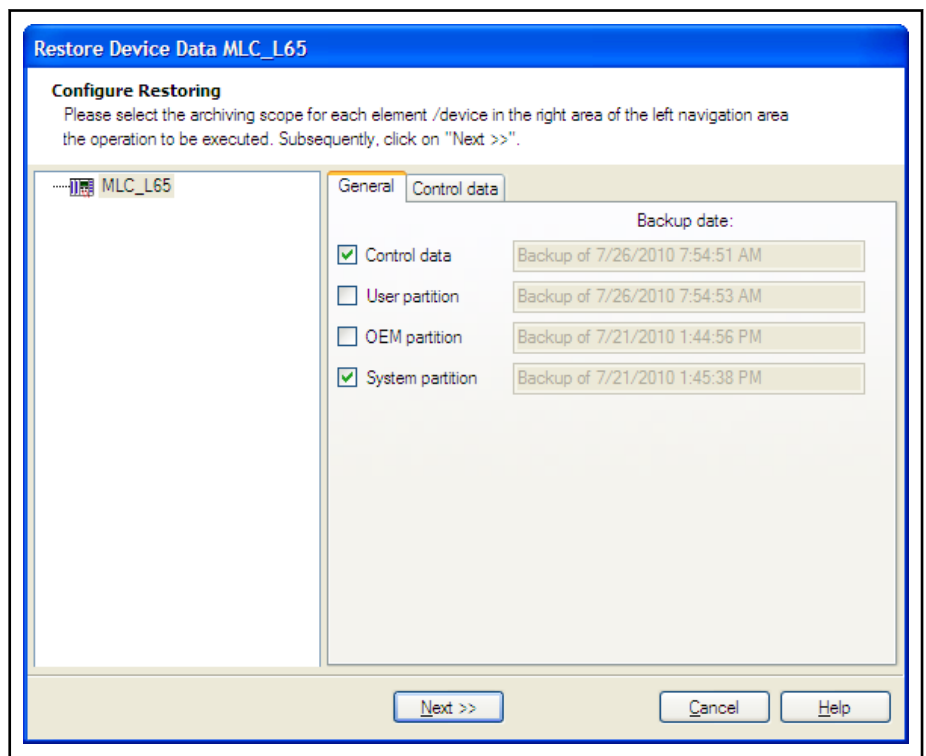


Fig. 6-89: Dialog to restore device data

The data types to be restored can be specified in the "General" tab. If a data type is in the project, the last backup data is displayed and the data type can be selected. Data types that are not located in the project cannot be selected.



Restoring the system partition overwrites the firmware currently on the control. All data and firmware settings are lost. The process may **not** be canceled. During the restoration of the system partition, the control is restarted.

If control data restoration was selected, it is possible to specify the parameters to be restored and whether the kinematic data is to be restored in the "Control data" tab. The "Other data" group is only available if at least one kinematics is configured in the project.

Control

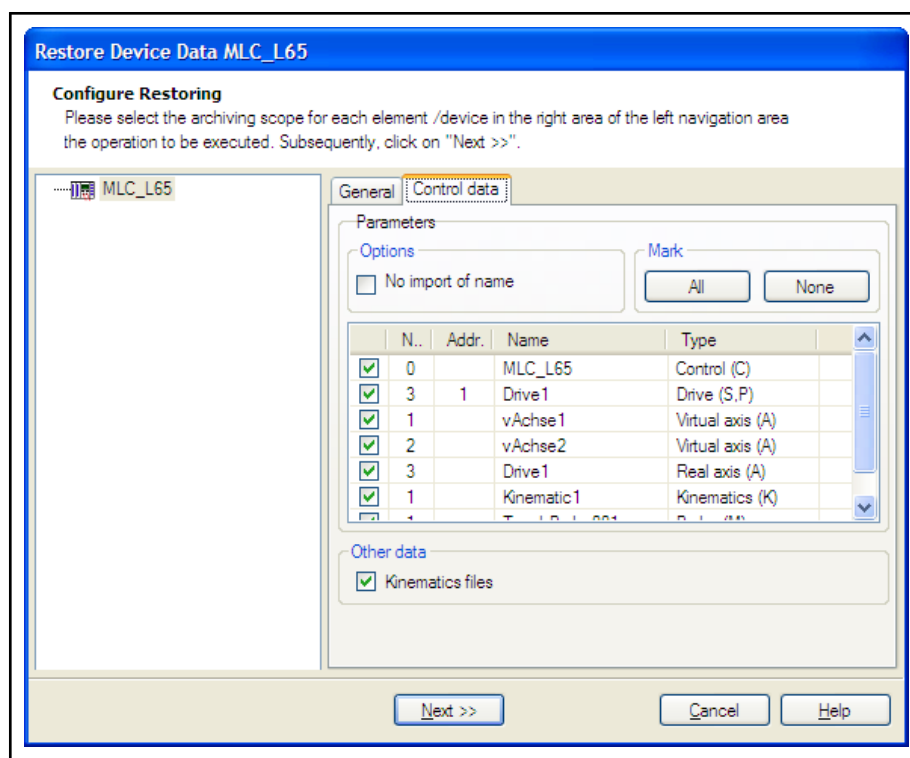


Fig.6-90: Selecting data to be restored

Finally, the data to be restored is displayed for confirmation purposes.

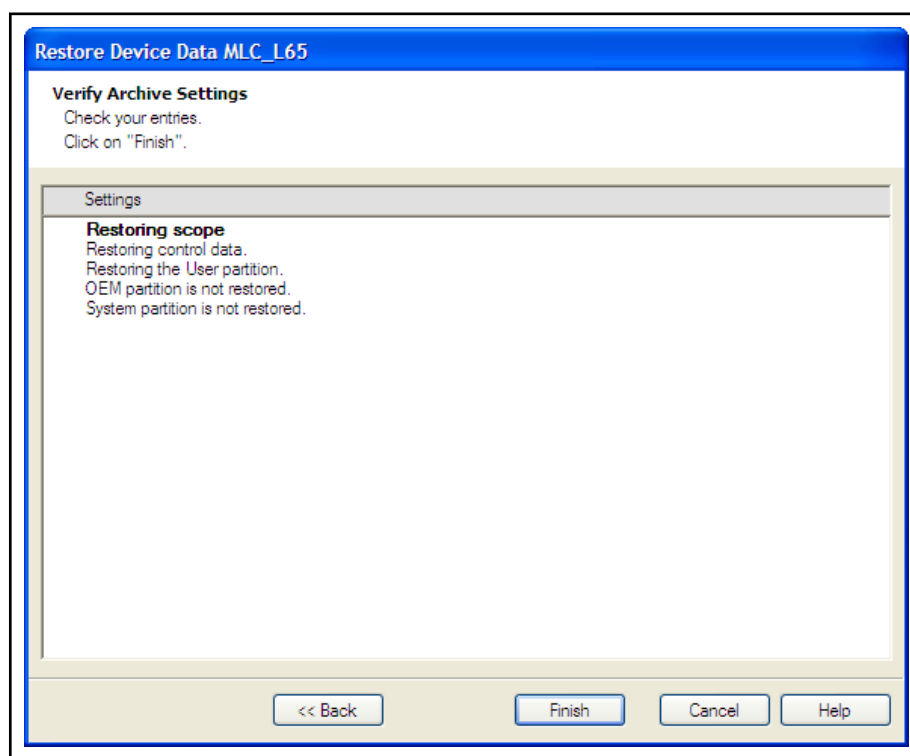


Fig.6-91: Check user inputs

Click on **Finish** to start the restoration. The total progress is then displayed.

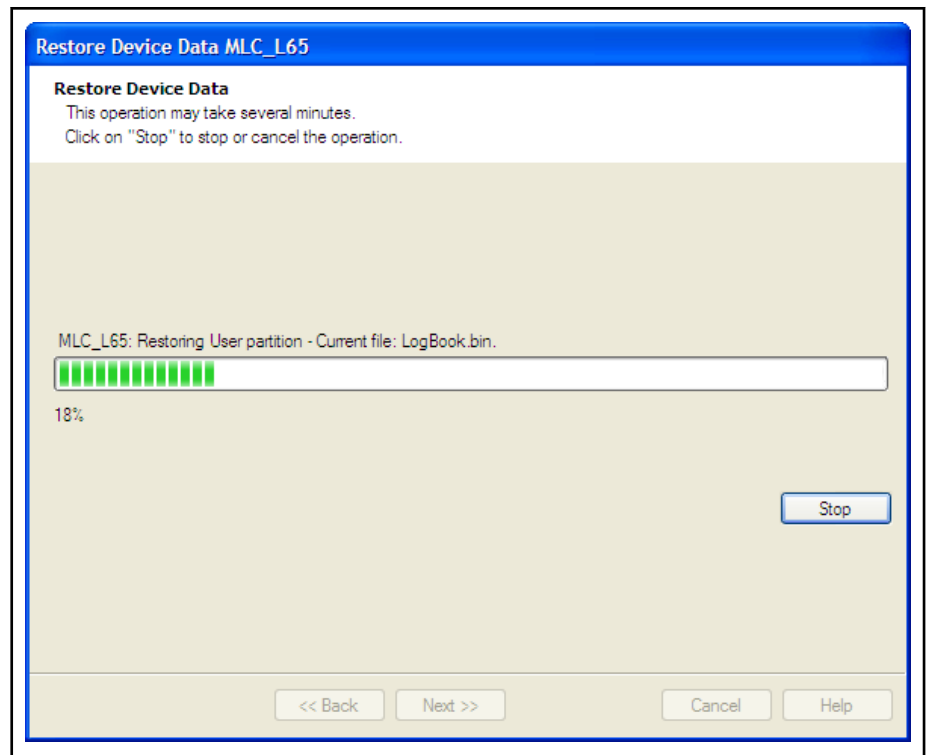


Fig. 6-92: Restoring...

At the end, a summary is output containing the restoration result.

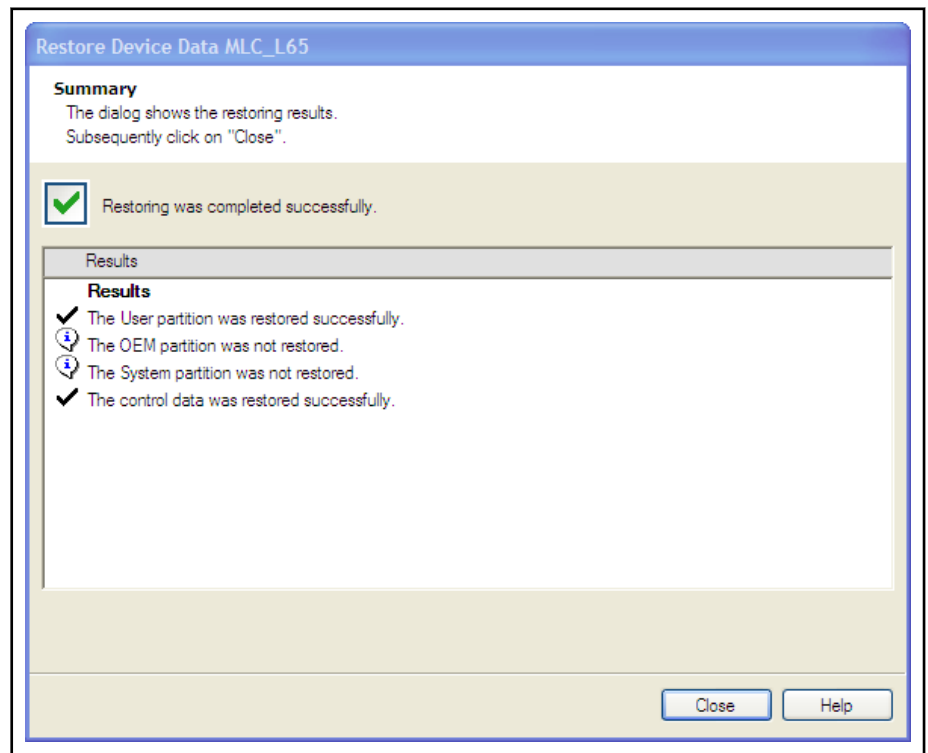


Fig. 6-93: Restoration summary

Control

Explorer

File and folders can be transferred between PC and control using the FTP Explorer. The FTP Explorer allows access to the OEM partition and USER partition of the control.

Opening the Explorer

1. Switch the control to online
2. Right-click to open the context menu of the control and select **Device Data ► Explorer**

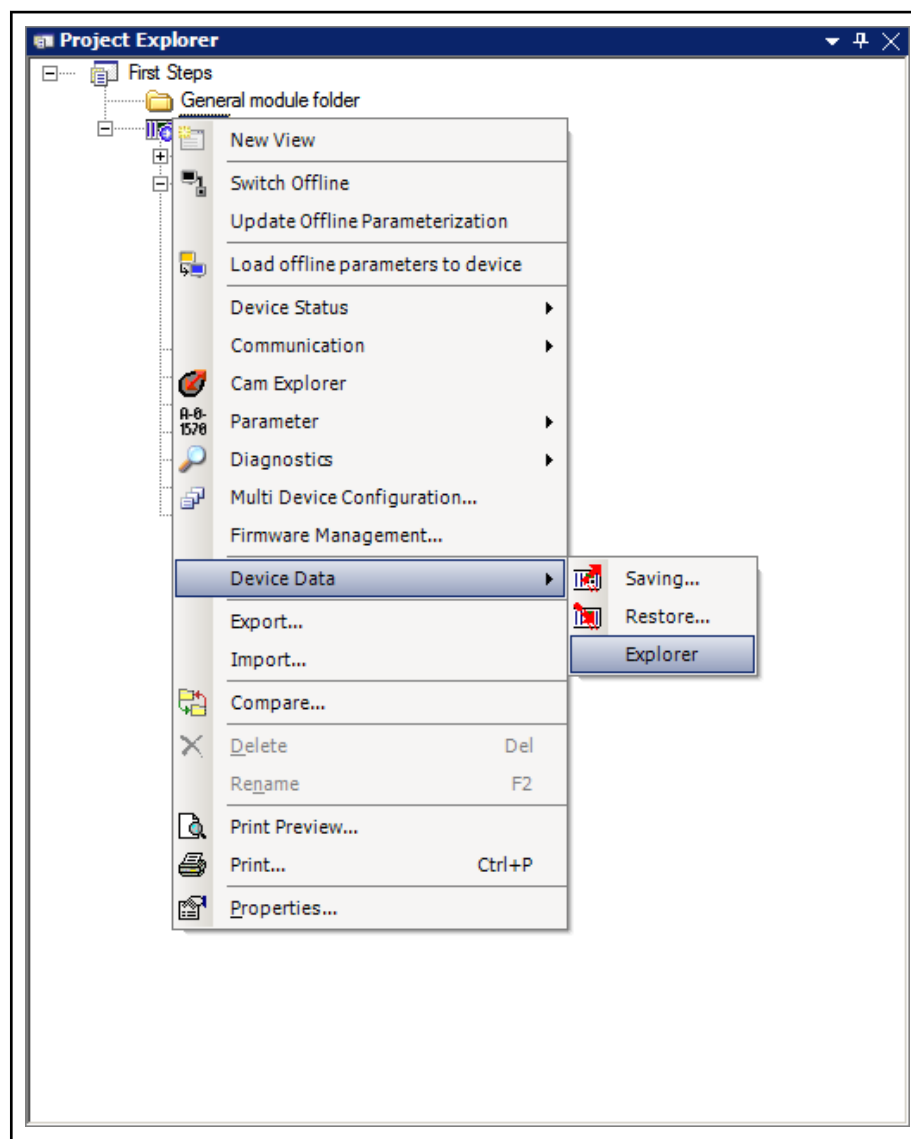


Fig.6-94: Selecting the FTP Explorer

Main window of the Explorer

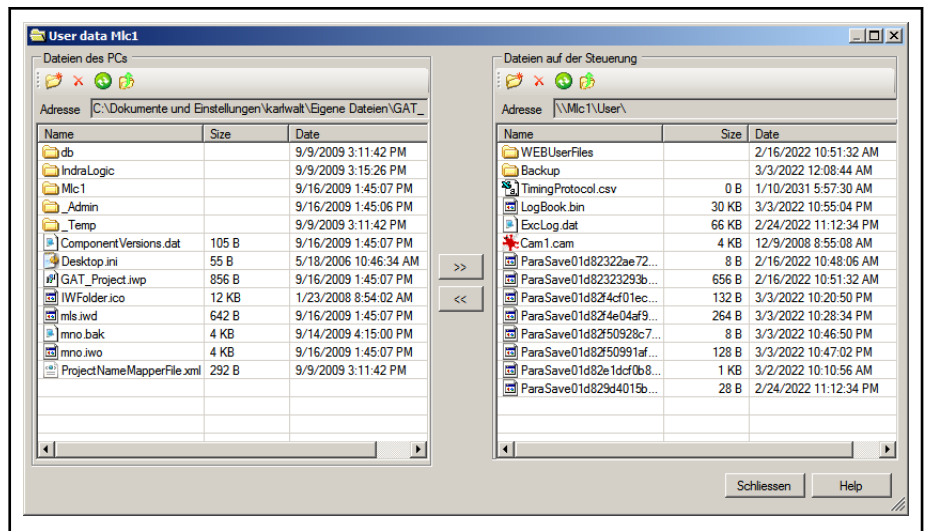


Fig. 6-95: Main window of the Explorer

The files and folders of the PC are displayed in the left section. The files and folders of the control are displayed in the right section.

FTP Explorer functions

There are two options to transfer data and folders between the PC and the control:

1. Select the files and folders in the left list and click on  to transfer the selected data to the control. Select the files and folders in the right list and click on  to transfer the selected data from the control to the PC.
2. Select files and folders and drag them to the other other list. Keep the left mouse button pressed while dragging.

Additional functions

- Create new folder:

To create a new folder, click on  in the toolbar

- Open file:

Select file and press <Enter>

or

right-click on the file and select the following menu item **Open** from the context menu

- Rename:

Select file or folder and press <F2>

or

right-click on the file or the folder and select the menu item **Rename** from the context menu (only available for files and folders of the control)

- Delete files and folders:

Select the files and folders to be deleted and click on

 in the toolbar

or

press <DEL>

or

Control

right-click to open the context menu and select the menu item **Delete**

- **Update view:**

Click on  in the toolbar

or

select the view (PC or control) and press <F5>

- **To return to previous level:**

Click on the icon  in the toolbar

or

select view and press <Back>

6.4.11 Simulation

The simulation command is available as context menu on the "Device" node and is used to switch on and off the PLC simulation mode of the programming system. In the PLC simulation mode, the active application can be loaded, started and tested on a "simulated target device". As a simulated device is always integrated into the programming system, no real target device is needed to test the runtime behavior of an application.

When the active application is logged in for the first time, a request is generated whether the application "Sim.<DeviceName>.<ApplicationName>" is to be created and loaded. No communication settings have to be entered for the simulated device. Following a successful login, the further online commands (Start, Stop, Breakpoints etc.) can be used to test the application.

To switch off the simulation mode again, first log off the control and then call the **Simulation** command again. The tick in front of the command disappears. Thus, login to the real target device is now possible.

6.4.12 Exporting

This menu item exports elements to an external file. The export is executed via a wizard.

In the first dialog, the elements to be exported, the path to save the file or an existing file are selected. In addition, the export file can be provided with a description (in the upper section of the dialog).

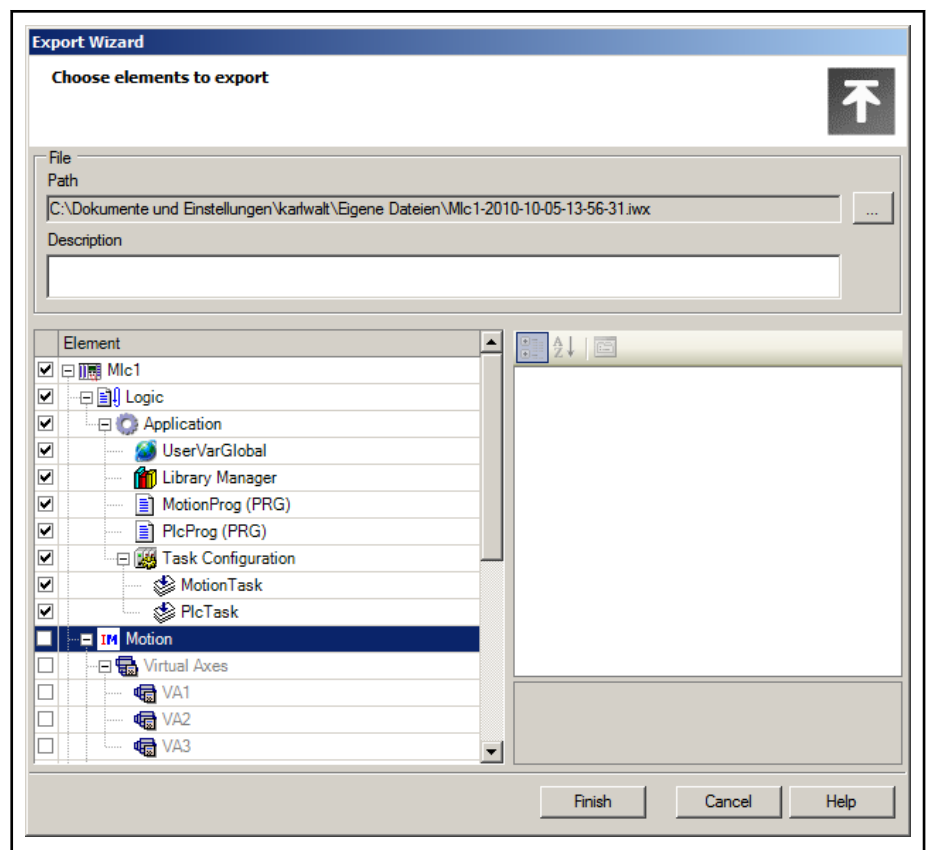


Fig. 6-96: Example: Selecting the IndraMotion MLC elements to be exported
In the last dialog, an export summary is displayed. This shows whether the export was successfully completed.

Control

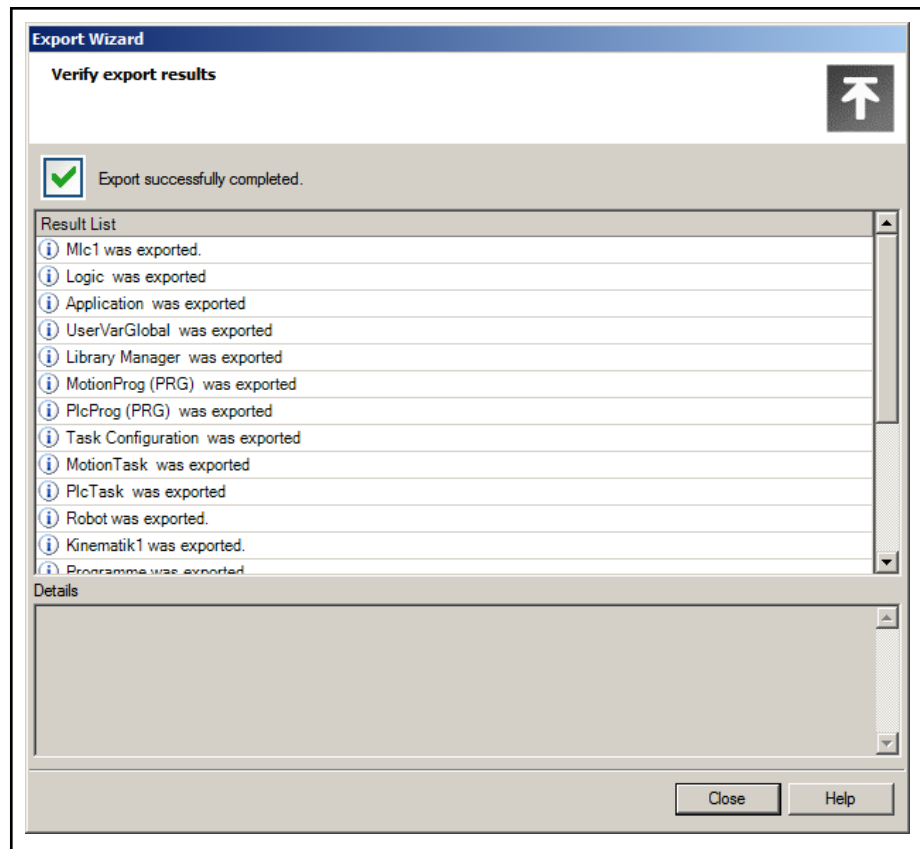


Fig. 6-97: Displaying an export procedure summary

6.4.13 Importing

Import elements from an external export file back to the project using this menu item. The import takes place via a wizard.

In the first dialog (Windows default dialog "Open"), an already existing export file (*.iwx) has to be selected for import.

In the second dialog, parts (such as nodes "Logic", "Robot") or the complete device (in the example "Mlc1") can be selected for import under "Element".

Under "Operation", select how the elements highlighted for the import are imported to the project.

There are the following options:

- **Paste**
Only available to import a complete device. If a device with the same name already exists in the project, the name of the device to be imported is renamed (in the example Mlc1 -> Mlc1_1) (see [fig. 6-100 "Displaying an import summary" on page 176](#))
- **Overwrite**
Overwrites existing elements and adds new elements from the export file to the project if required
- **Overwrite + Delete**
Overwrites existing elements, adds new elements and deletes elements not existing in the export file

Press **Details...** to open another dialog (see [fig. 6-99 "Example: Selecting the elements to be imported for the selected area" on page 175](#)) in which the el-

Control

elements for the selected area (in this example, the complete device Mlc1) can be individually deselected or selected.

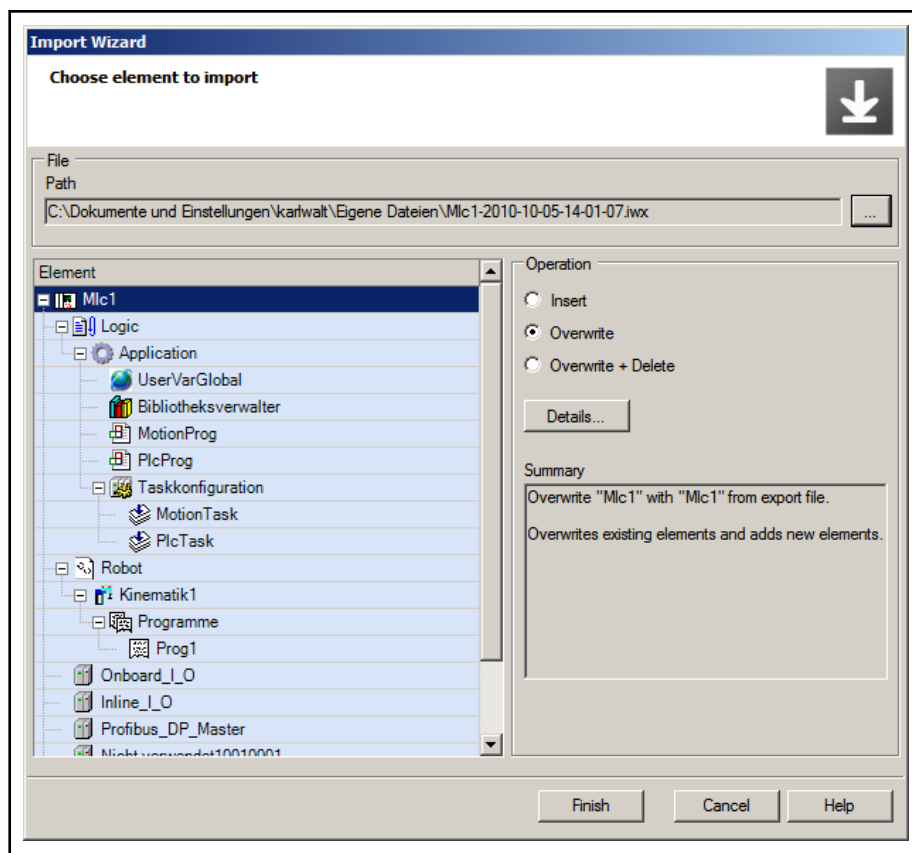


Fig.6-98: Example: Selecting the area to be imported (in the example the complete device Mlc1)

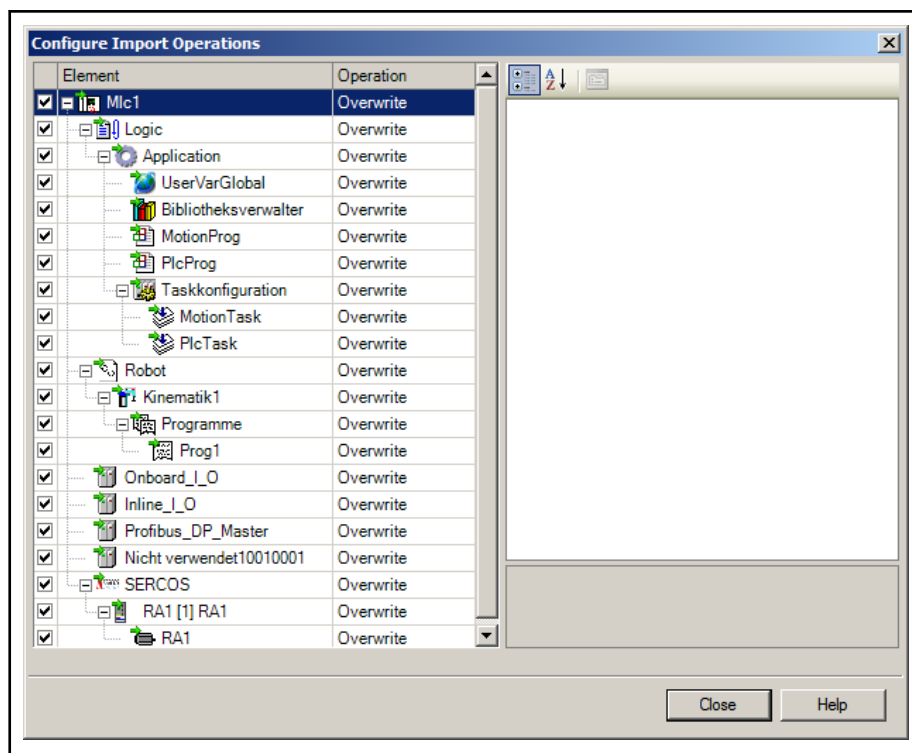


Fig.6-99: Example: Selecting the elements to be imported for the selected area

Control

The last dialog displays a summary of the import. This shows whether the import was successfully completed.

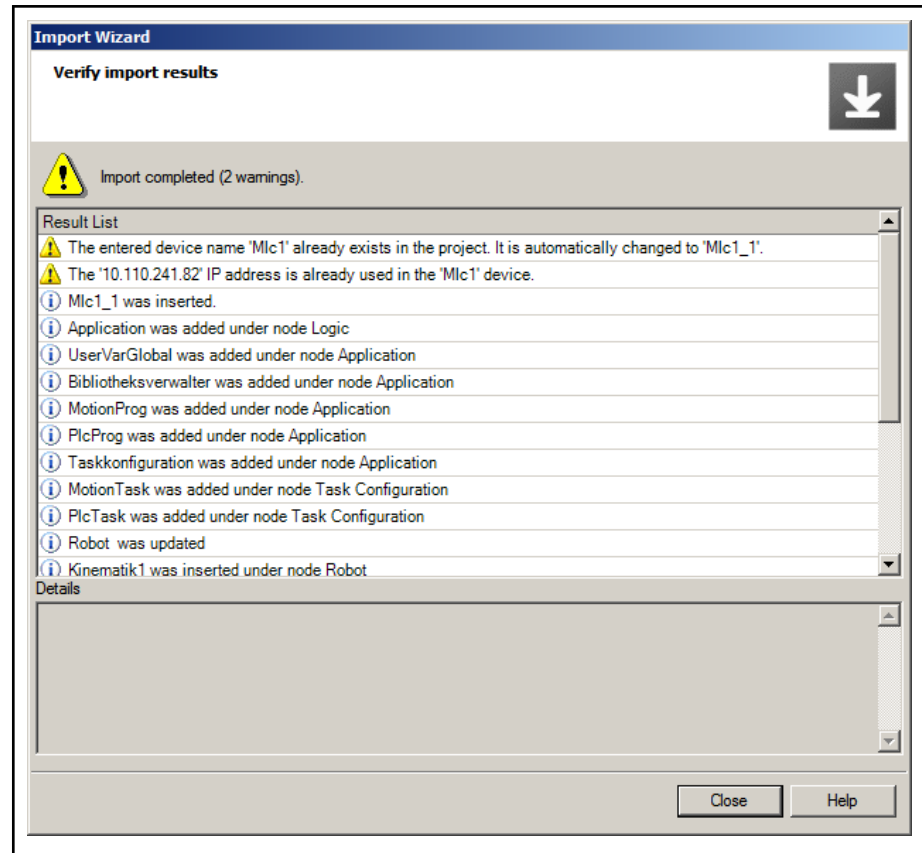


Fig. 6-100: Displaying an import summary

6.4.14 Comparing/Merging

General Information

The basic functions of the "Compare..." context menu item are identical to all other IndraWorks components (for a description refer to the "IndraWorks Engineering" manual or help, chapter "Comparing Project Data").

The special behavior of IndraLogic objects is described in the IndraLogic manual or help.

Additional deviations in the Motion objects compared to the normal behavior are described in the following.

General rules for Motion objects

Irrespective of the object type, IndraWorks ensures that the following rules are observed when merging objects:

- The value of a merged object has to be different
- Dependent properties are compared and merged
- The merging result may not cause a conflict

Control

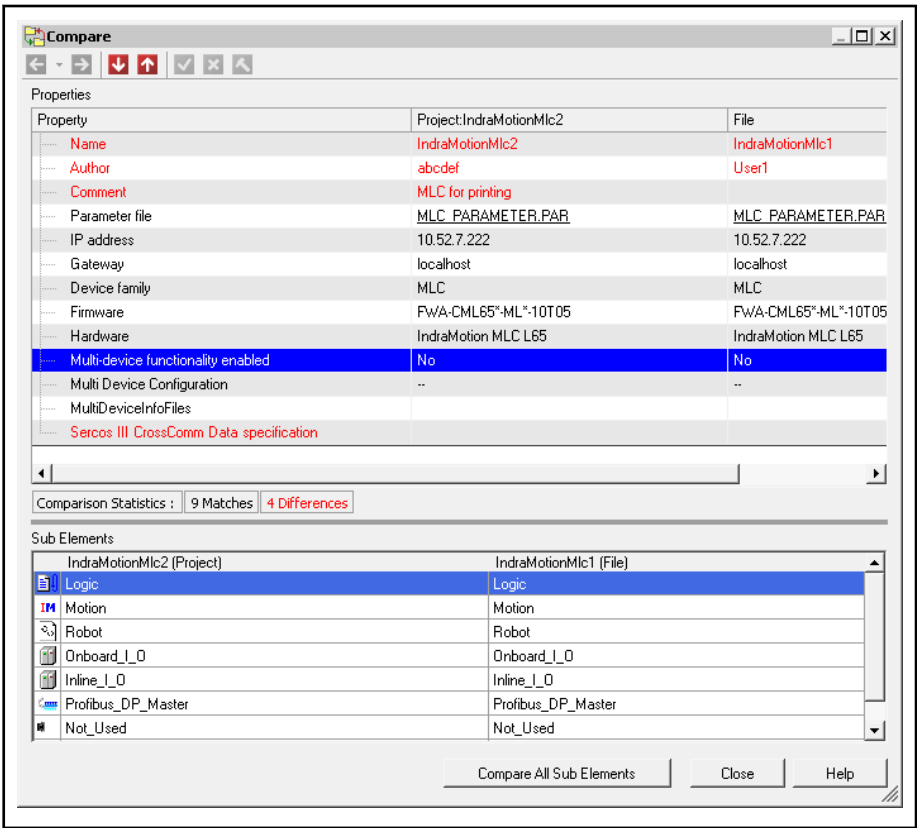


Fig.6-101: "Comparison" dialog

The file names are underlined. Click on the file to display the content. The file is opened in the editor linked to this file type.

Restrictions when merging:

- When comparing different controls (e.g. IndraMotion MLC L65 and IndraMotion MLC L45), the firmware cannot be merged

Axes

General Information

An axis can only be compared and merged with an axis of the same type.

Real axis

Comparison and merge operations cannot directly be executed for a real axis. This has to be performed with the respective drive node below the sercos node instead.

Restrictions when merging:

- Neither the sercos address nor the axis number can be merged

Encoder axis

Restrictions when merging:

- The respective drive and drive address are always merged

Behavior of all other axis types

Restrictions when merging:

- The axis number has to be unique. Each merging operation violating this rule causes an error message when executing the action.

Robot

Kinematics

Restrictions when merging:

Control

- The kinematic number has to be unique. Each merging operation violating this rule causes an error message when executing the action.
- The kinematic axes used have to be available in the project if the merging operation is executed

RCL program

Restrictions when merging:

- The RCL program, its name and .pkt file are always merged

Function Modules

General Information

A function module can only be compared or merged with a module of the same type.

CrossComm sercos III (CFL01.1-R3) function module

Restrictions when merging:

- The control number, the produced data and the "Device is master" property are always merged

6.4.15 Runtime Licenses

The "Runtime Licenses" dialog outputs all licenses used by the PLC program at runtime. Required licenses are only displayed if the function subject to license is actually used or called. The list of the used functions subject to license is deleted when switching off the control.

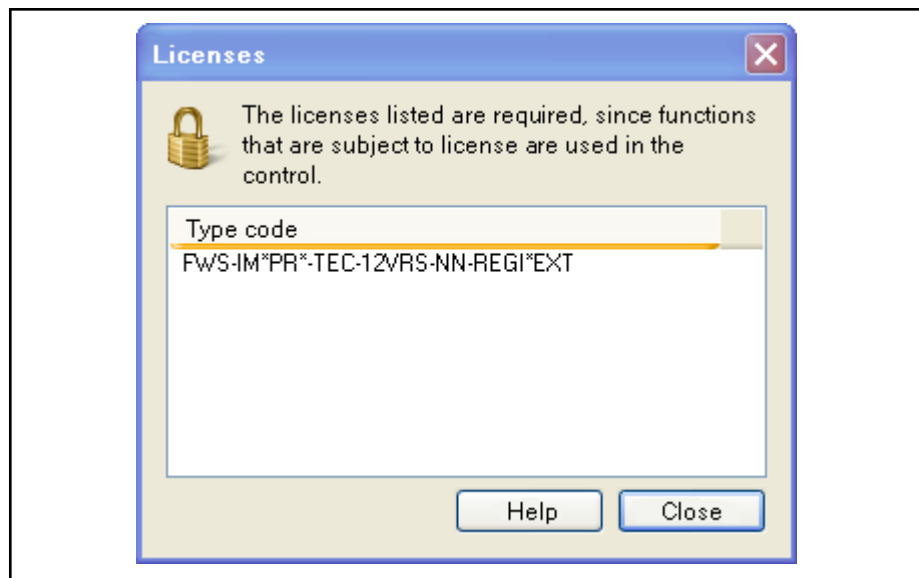
The dialog has only a display function. Values cannot be changed.

Availability

This function is available in online mode.

Structure

Open the "Runtime Licenses" via the context menu of the control.



Type code: Type code of used function that is subject to license

Fig. 6-102: Calling the "Runtime License" dialog

6.4.16 Print Preview

This menu item can be used for a print preview of the following data of a control:

- Properties
- Configuration
- Parameters

In the first dialog box, the control as well as the data for the print preview can be selected.

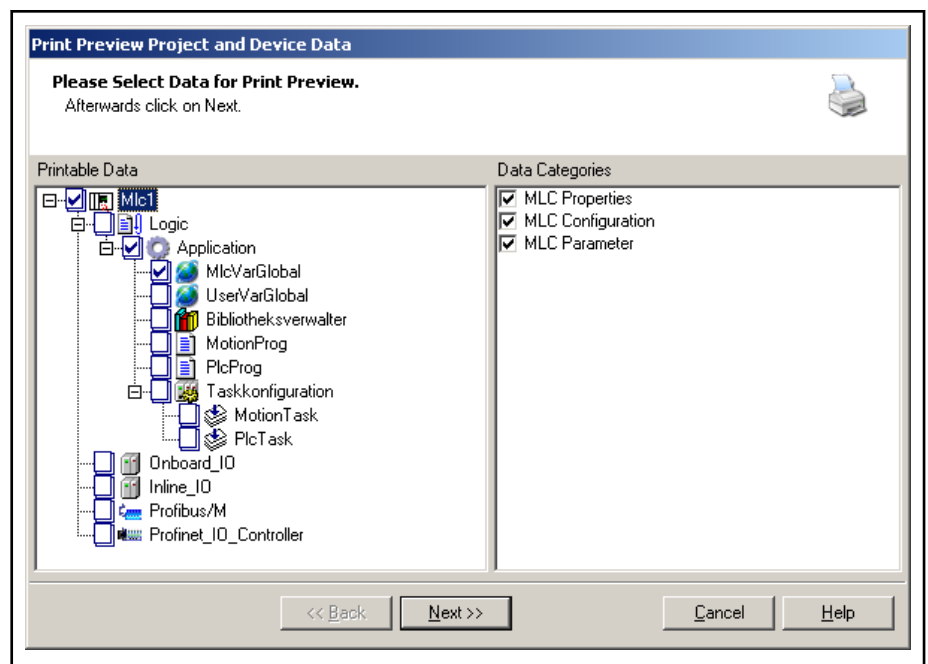


Fig.6-103: Example: Selecting data for print preview for an IndraMotion MLC

MLC properties

Includes project details such as project name, creation date, device names, general data, firmware versions and communication settings.

MLC configuration

Includes project details such as project name, creation date, device names, function modules and axis configuration.

MLC parameters

Lists all parameters, project name, creation date and device name.

In the following dialog box, check the previously selected settings and make the print settings (printer selection, layout, paper size and page setup) with "Page Setup...". The **Finish** button starts the print preview.

Control

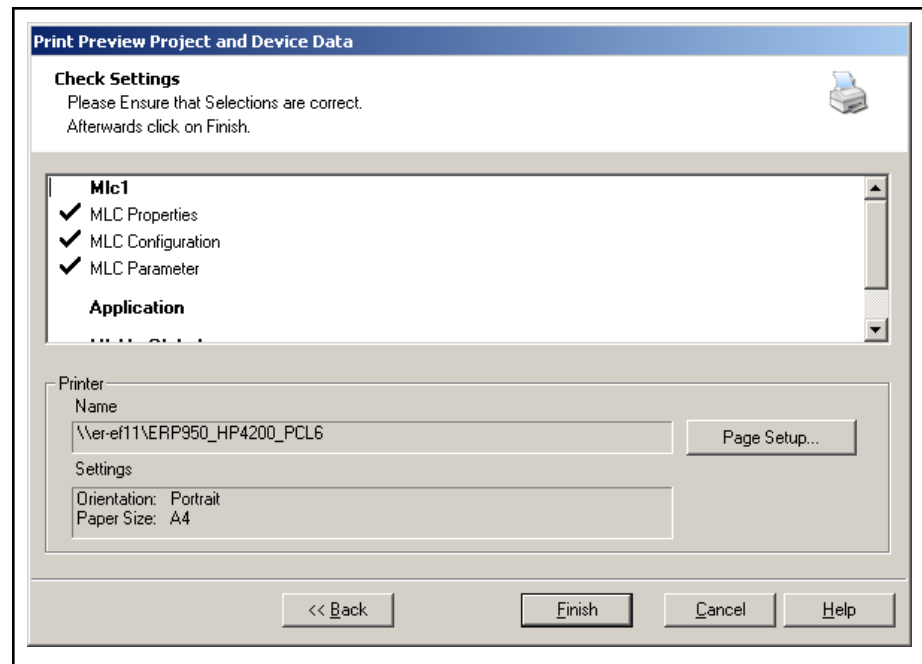


Fig.6-104: Example: Checking print preview settings for an IndraMotion MLC

The print preview can be displayed in the parameterization mode as well as in the operating mode. It can, particularly in the second case, take one or even several minutes.



Within the print preview, data can also be printed by selecting the printer icon (top left).

6.4.17 Printing

Use this menu item for an unformatted documentation of the:

- Properties
- Configuration
- Parameters

of a control

In the first dialog box, the control as well as the data to be printed can be selected.

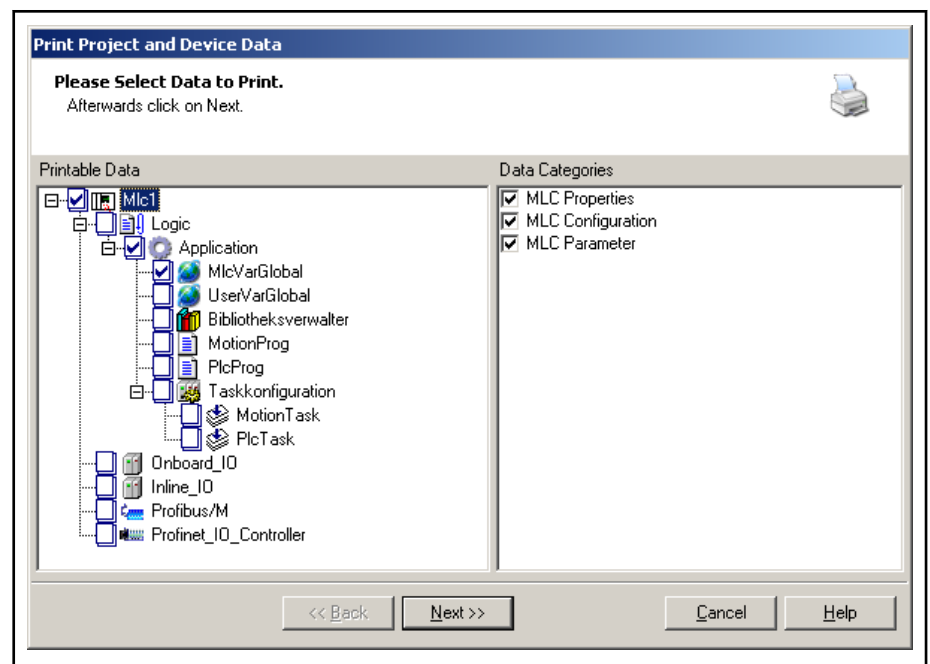


Fig.6-105: Example: Selecting data to be printed for an IndraMotion MLC

MLC properties

Includes project details such as project name, creation date, device names, general data, firmware versions and communication settings.

MLC configuration

Includes project details such as project name, creation date, device names, function modules and axis configuration.

MLC parameters

Lists all parameters, project name, creation date and device name.



Controls store C-parameters. Real/virtual/encoder/link axes store A-parameters and IndraDrive/HydraulicDrive drive store S- and P-parameters.

If an axis/a drive is disabled, only the A-parameters duplicated on the control can be printed.

In the following dialog box, check the previously selected settings and make the print settings (printer selection, layout, paper size and page setup) with "Page Setup...". Press **Finish** to start printing.

Control

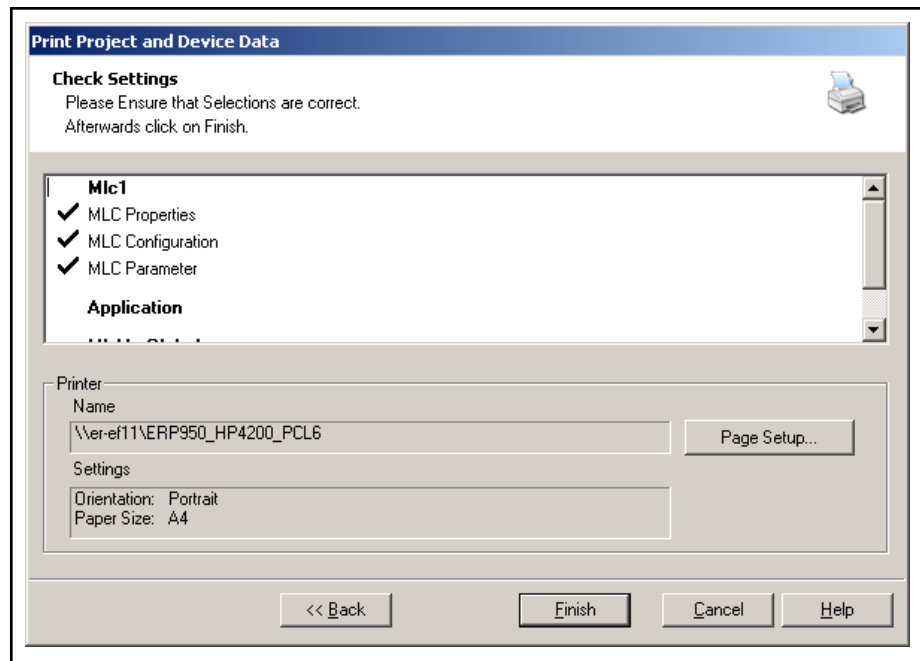


Fig.6-106: Example: Checking print settings for an IndraMotion MLC

It can be printed in the parameterization mode as well as in the operating mode. It can, particularly in the second case, take one or even several minutes.

6.5 Version Control of the Control

6.5.1 Introduction

All of the controls in IndraWorks listed below support version control:

- IndraMotion MLC L25
- IndraMotion MLC L45
- IndraMotion MLC L65

For the documentation of the version control in IndraWorks, refer to "Rexroth IndraWorks 12VRS Engineering" (see /5/, [Required and Supplementing Documentations, page 14](#))

6.5.2 Special Features

Project Management

There are some special features in the version control of MotionLogic devices.

CrossComm function module

The IndraMotion MLC and the CrossComm function module (sercos II and sercos III) are handled together. This avoids that several inconsistent versions are created due to simultaneous editing (by several users).

MlcVarGlobal

The "MlcVarGlobal" object is not version-controlled. This global variable list is automatically generated according to the Motion and Robot configuration.

Technology

The "Technology" node is not version-controlled. Instances of supported technology functions of the PLC program are displayed in the technology node. This node does not contain data to be version-controlled.

Real Axes

The axes under "Real Axes" are not version-controlled. One drive in the "sercos" node of the control belongs to the each of these axes. When editing a real axis, the respective sercos device is automatically checked out.

Offline parameterization	Some parameters of the "offline parameterization" are version-controlled. It is absolutely required to check out the control before changing the offline parameterization. After the check in and the update of the workspace, the changed parameters are also available to other users working in that team project.
Exchanging a drive	When using the "Exchange Drive" interface function, the version history of the exchanged drive is lost since the object is deleted and created again.

Working with Controls

Motion Connection with Version-controlled Project

If several users work on a team project with a common control, only one user can be connected online.

If each user is to have an own control, the IP address can be adapted without checking out the control. The IP address does not belong to the version-controlled data of the control in order to be independent from other users when working with different controls.

PLC Connection with Version-Controlled Project

The data connection between IndraWorks and the PLC of the control, allows only one user (like the Motion connection) to be connected. Each user can set his individual IP address without checking out the control.

Note that alternating login to the PLC by different users (with different work copies of the team project) cannot be performed by online change. For this change, a new program is downloaded in which the PLC has to be stopped.

Using Multi-Device Configurations

The operating mode of the multi-device configuration of the control is documented in [chapter 6.4.8 "Multi-Device" on page 150](#).

For version-controlled projects, in which the multi-device configuration is used, not the following:

- The multi-device table is version-controlled. A change can only be made by checking out or hijacking the control.
- It can be switched between different controls via the multi-device configuration without checking out the control. Thus, each editor of the team project can change his control irrespective of the other users.

6.6 Complete Data Backup

6.6.1 General Information

Control data is completely backed up via archiving. Depending on the project size and the number of axes, the data backup can take a few minutes.



Depending on the project size and the number of axes, the data backup can take a few minutes.

The following data is archived:

- Configuration
- Control parameters
- Drive parameters
- Kinematic data
- Libraries, device descriptions

Control

- Compact Flash contents

6.6.2 Complete Backup

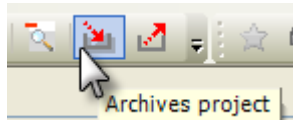


To ensure the archiving of the **current control data**, switch to **on-line**.

When working offline, only the control data last saved - if available - can be included in the archive.

The complete backup is as follows:

1.



Select the **Archives project** icon in the IndraWorks toolbar.

2. Enter the name of the archive as well as the optional input of a comment and/or password:

Fig.6-107: Enter the archive name

3. Select the storage location (file system and/or FTP server):

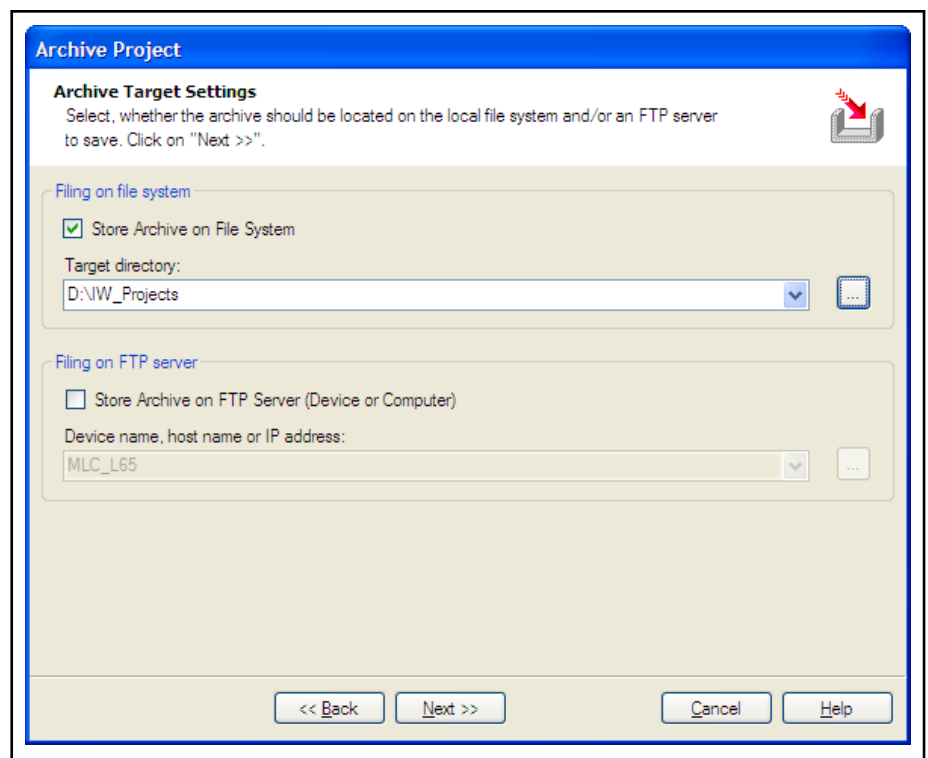


Fig.6-108: Select the storage location

4. Select the data to be archived. It is possible to specify on the "General" tab whether the control data is to be backed up and which partitions of the control are to be backed up. The three partitions contain the following data:
 - User partition: Cam tables transferred to the control, MotionProfile as well as any personal files
 - OEM partition: Archives saved on the device, compiled PLC programs
 - System partition: Firmware, PLC retain data

Control

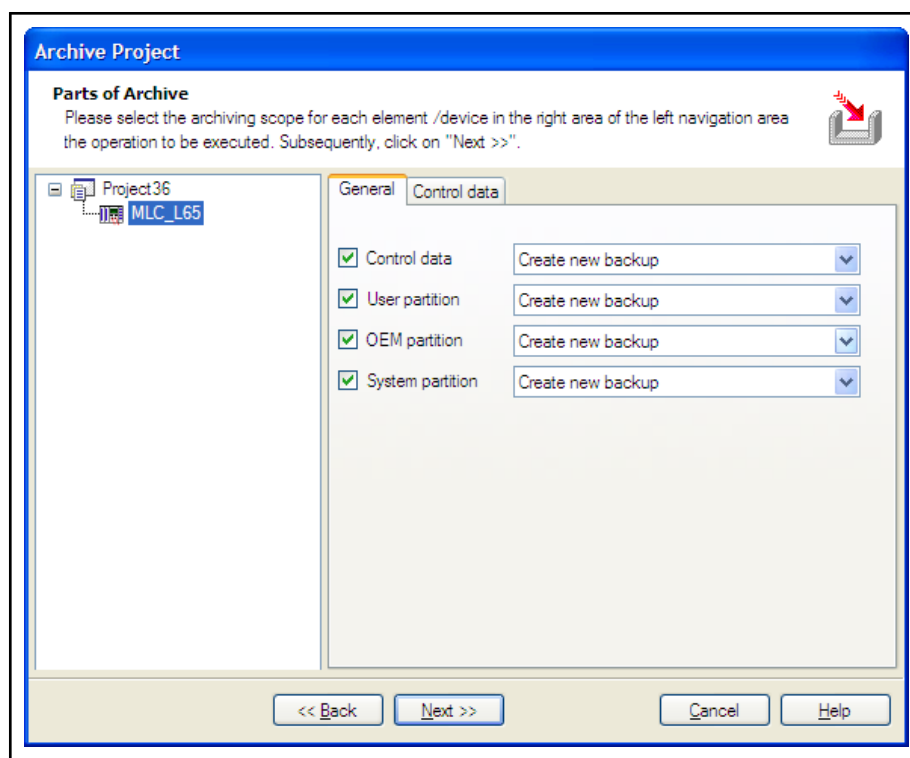


Fig.6-109: Selecting components to be archived (1)

On the "Control data" tab, the parameters to be saved can be specified and whether kinematic data is to be saved.

It is only possible to make an entry on this page if "Create new backup" was selected on the "General" tab at "Control data".

If not at least one kinematics is configured in the control, the "Other data" area is not displayed.

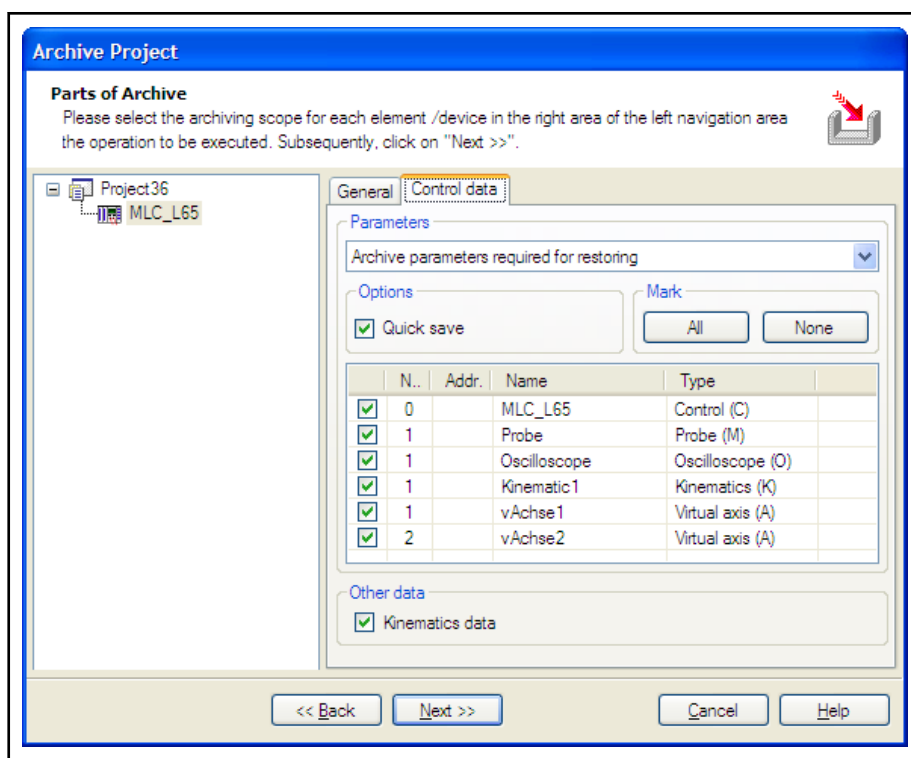


Fig.6-110: Selecting components to be archived (2)

5. Check user inputs:

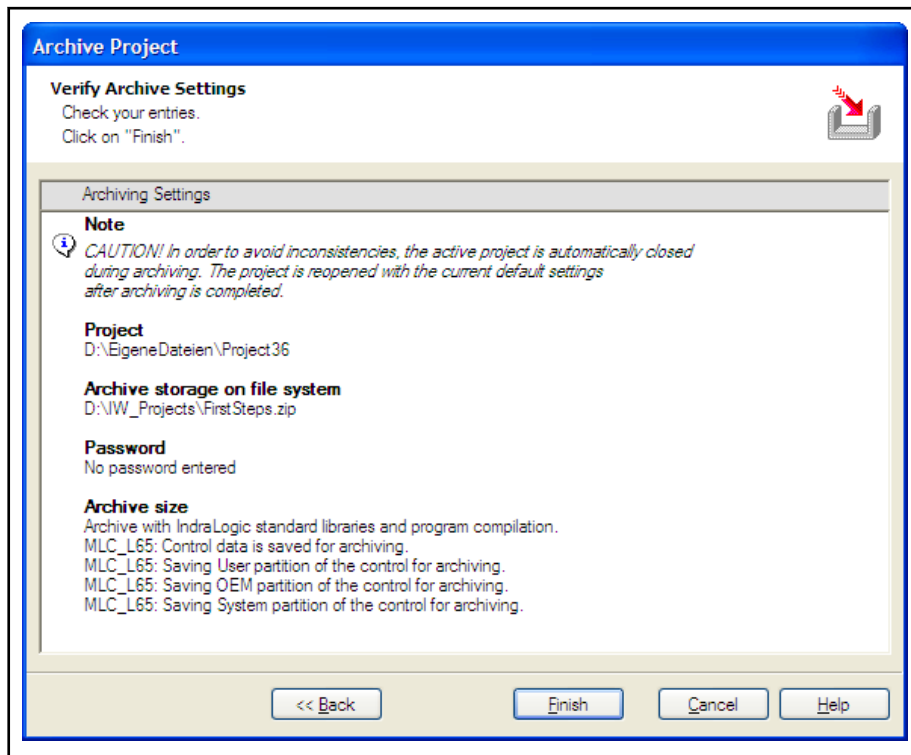


Fig.6-111: Check user inputs

6. Archiving with progress bar:

Control

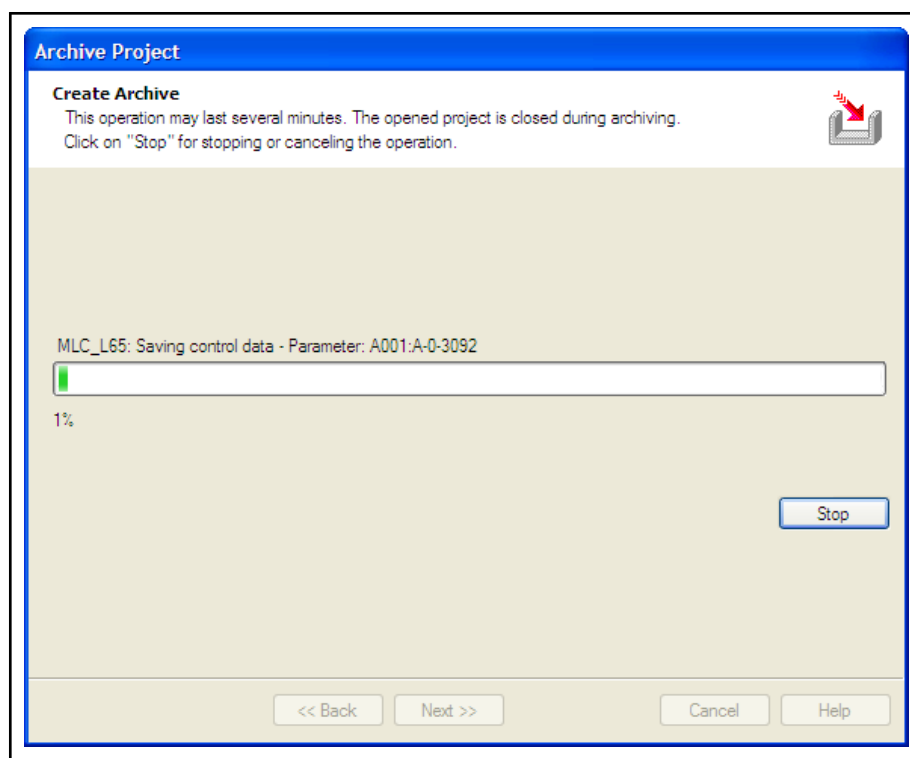


Fig.6-112: Archiving...

While creating the archive, this additional dialog is displayed to save drive parameters in the non-volatile drive memory.



Fig.6-113: Dialog to archive drive parameters

7. Display of a summary:

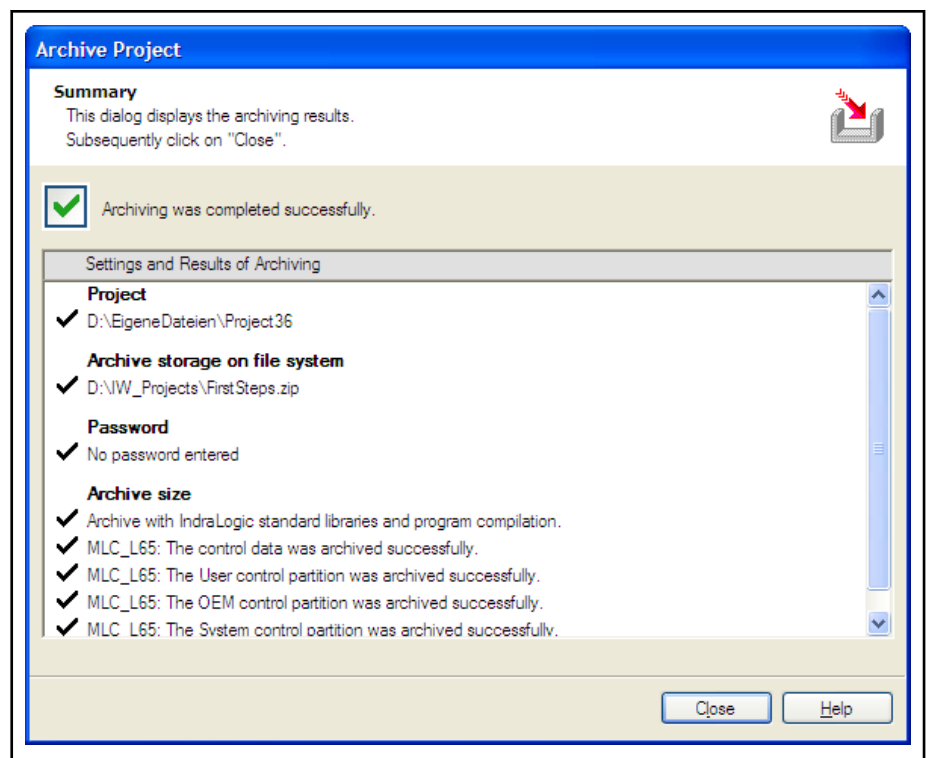
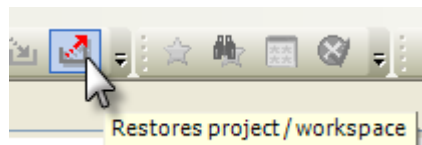


Fig.6-114: Summary of the project backup

6.6.3 Restoring a Project

The project is restored as follows:

1.



Select the
toolbar.

icon in the IndraWorks

2. Specify the restoration type, that is the source (file system or FTP server) of the project to be restored:

Control

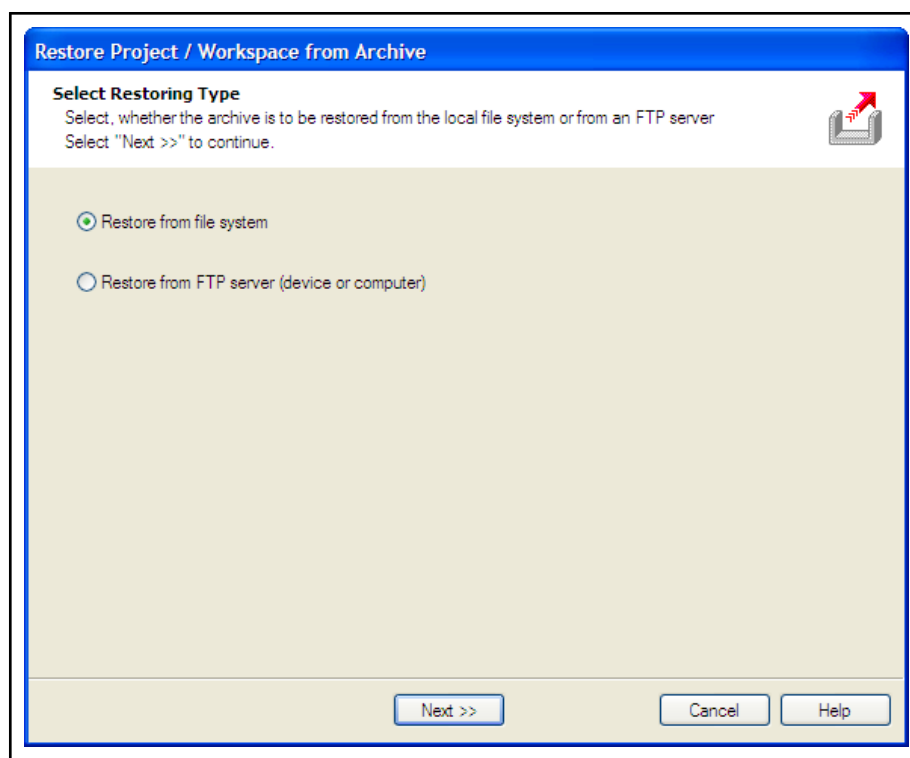


Fig.6-115: Specifying the restoration type

3. Select the archive to be restored:

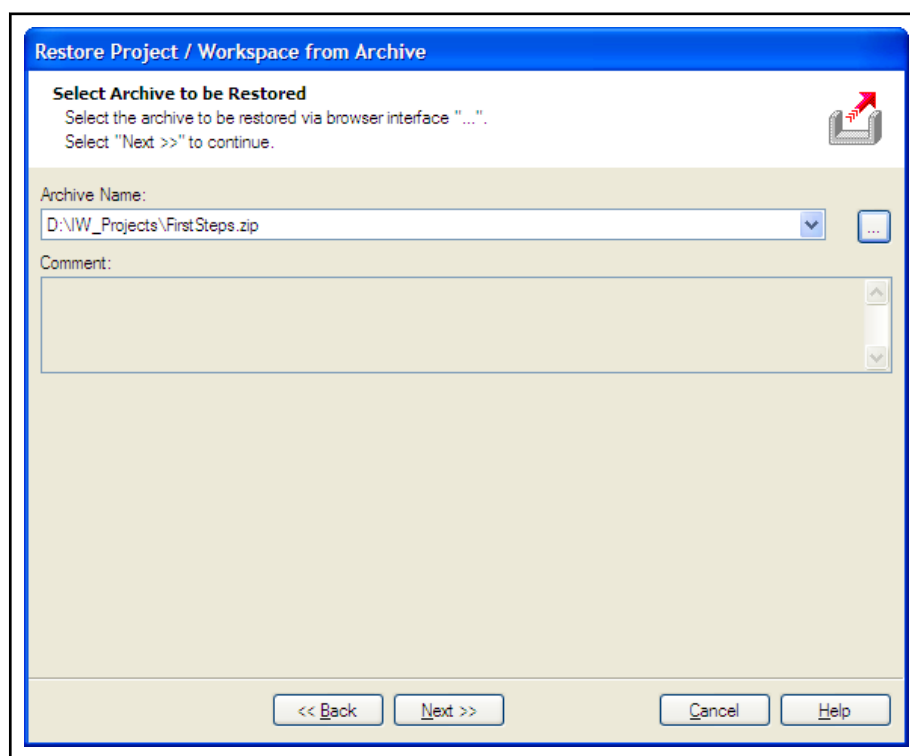


Fig.6-116: Select the archive to be restored

4. Select the directory in which the archive is to be restored. A folder with the project name is automatically created.

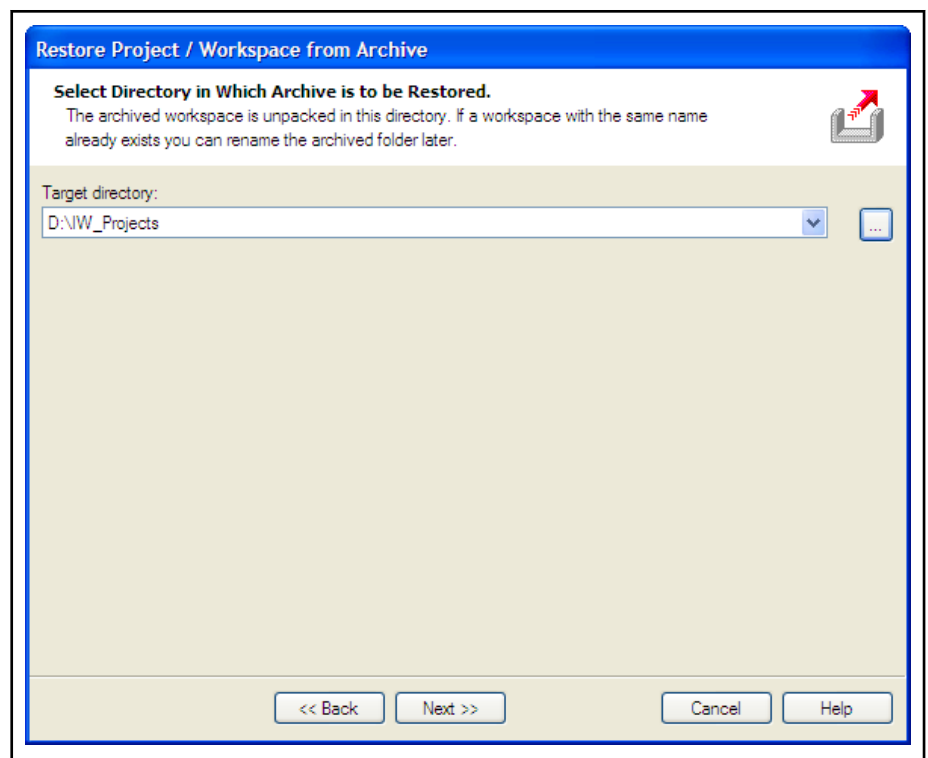


Fig.6-117: Selecting the directory...

5. Check the previous specifications:

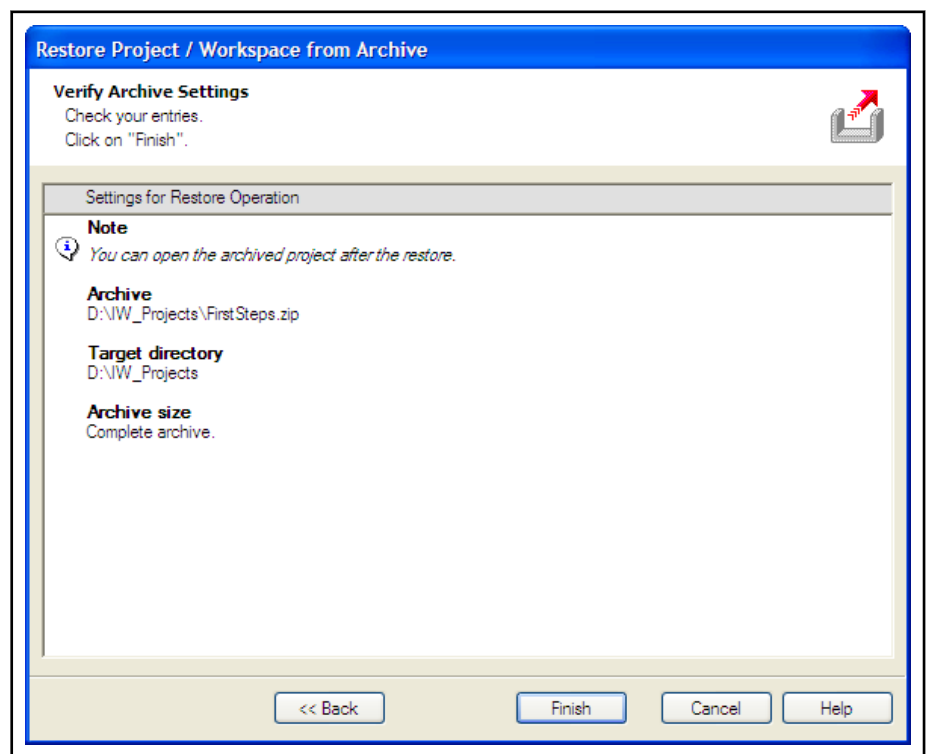


Fig.6-118: Check the specifications made

If a password was specified when the project was archived, a dialog to enter the password opens:

Control

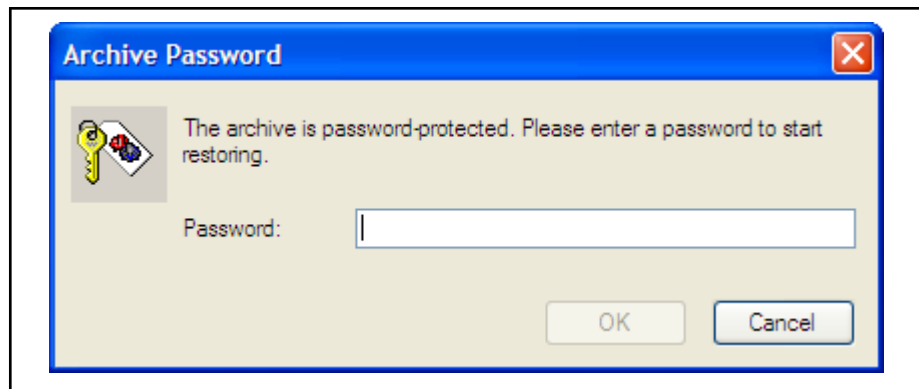


Fig.6-119: Input dialog for the password

6. A summary of the project restoration is displayed. The selected archive, the target directory, the project directory and the archive size are displayed:

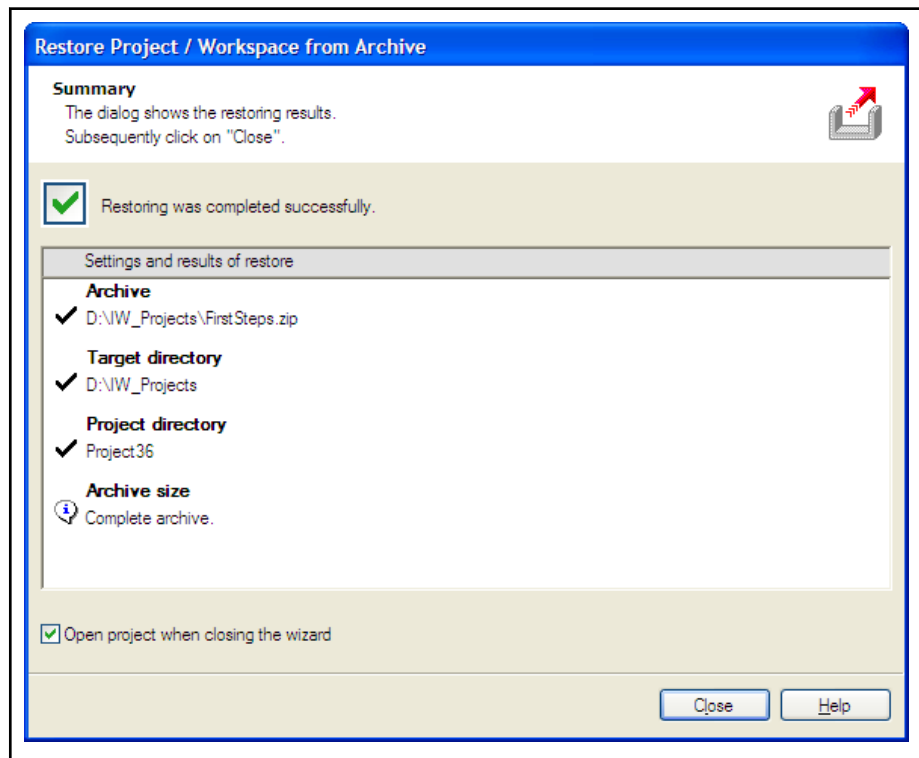


Fig.6-120: Displaying a summary

7. The project was restored. If device data is present in the project, the following message is displayed when opening the project for the first time:

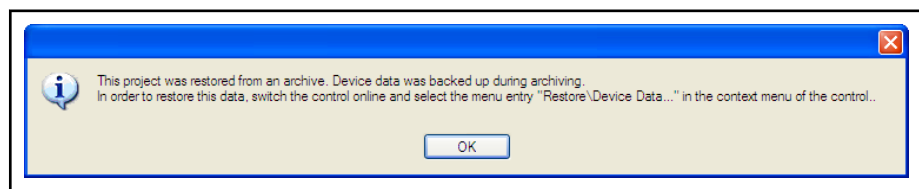


Fig.6-121: Message to restore device data

The menu item **Restore Device Data** is located in the context menu of the control (see [chapter "Restoring" on page 167](#)).

7 PLC Programming

7.1 General Information

This chapter provides a brief overview on the PLC programming on the IndraMotion MLC system. Following a short introduction, the chapter deals specifically with the library manager and the task system. The chapter then deals with I/O accesses by the PLC. The chapter ends with a section on the memory organization and memory access from the PLC.

For a detailed functional description of the IndraLogic PLC, refer to /3/, [Required and Supplementing Documentations](#), page 14.

7.2 Creating PLC Program

7.2.1 Opening "Plc_Main Object"

Double-click to open the "Plc_Main" object in the Project Explorer. Alternatively, click on "Open" to open the "Plc_Main" object via the context menu. The "Plc_Main" object workspace opens. The upper area is used to edit the variables (declaration part) and the lower area to edit the instruction (instruction part).

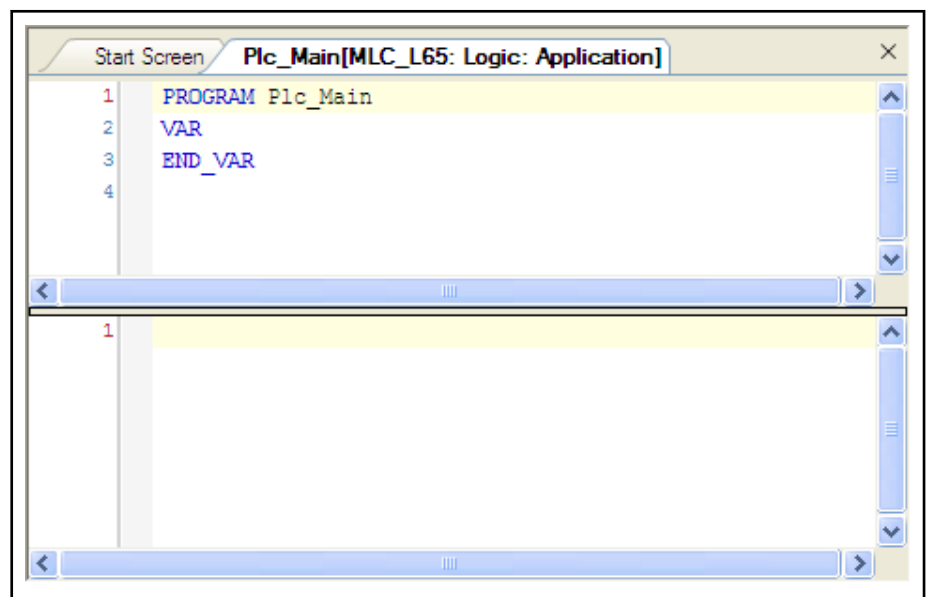


Fig. 7-1: Plc_Main

7.2.2 Declaring Variables

Enter the line "i:=i+1;" in the code section (lower area) and confirm the entry with "Return". The "Auto Declare" input field opens automatically to declare the variable "i".

PLC Programming

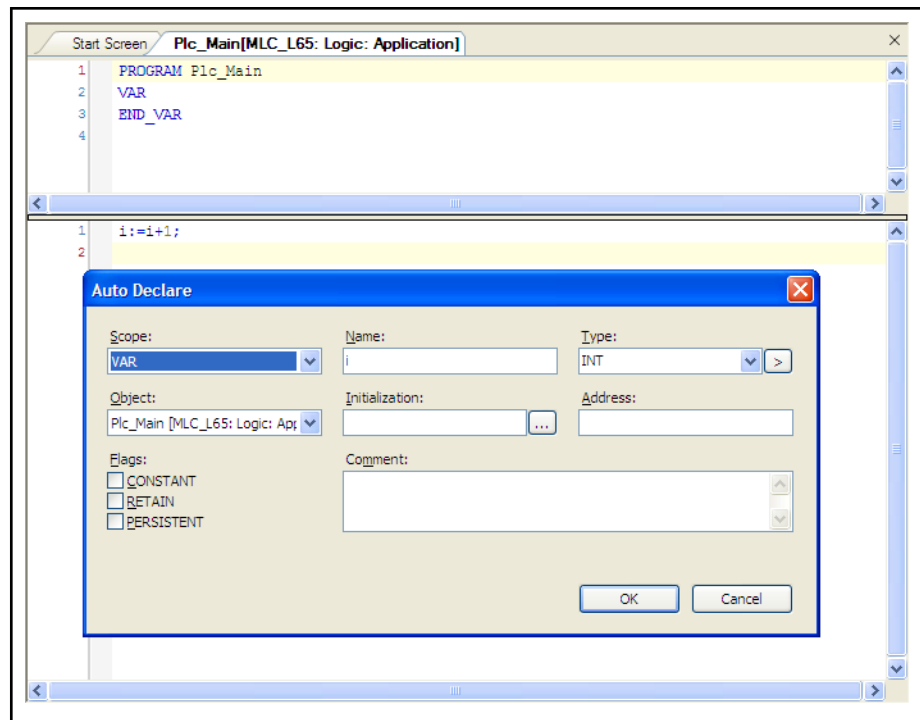


Fig. 7-2: Declaring variables

Confirm the specified values with **OK**. The variable "i" of the "integer" type is now declared.

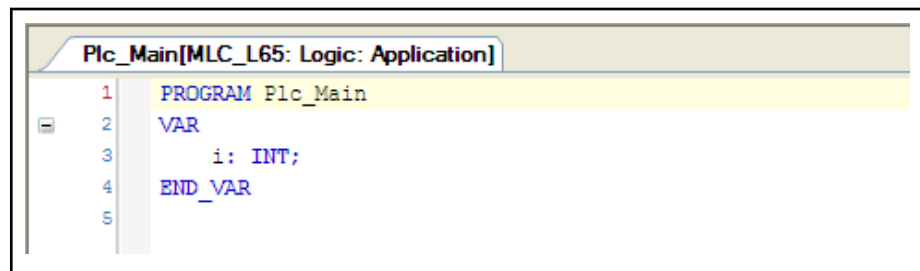


Fig. 7-3: Declared variable



The variable "i" can also be declared in the upper workspace by manual entry.

7.2.3 Editing Instruction

It is possible to edit instructions in the lower area of the window. There are six programming languages available:

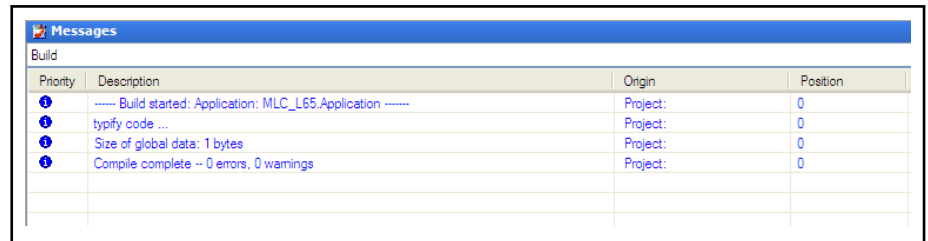
- ST (structured text)
- LD
- IL (instruction lists)
- CFC
- FBD
- SFC

For a detailed description, refer to the online help under IndraLogic Editors.

7.2.4 Compiling Program

To compile the project, select **Build ► Application[BRC_Control: PLC logic]**.

IndraLogic then starts the compilation and outputs the result (in this case 0 errors, 0 warnings) in the task list. The task list contains all messages from the most recent compilation, check or comparison.



Priority	Description	Origin	Position
Build			
1	Build started: Application: MLC_L65.Application -----	Project:	0
1	typify code ...	Project:	0
1	Size of global data: 1 bytes	Project:	0
1	Compile complete -- 0 errors, 0 warnings	Project:	0

Fig.7-4: Compiling a program

7.2.5 Loading the PLC Program to the Control

To load the project to the control, select **Debug ► Login on 'Application [BRC_Control: PLC logic]**.

The following dialog is displayed.

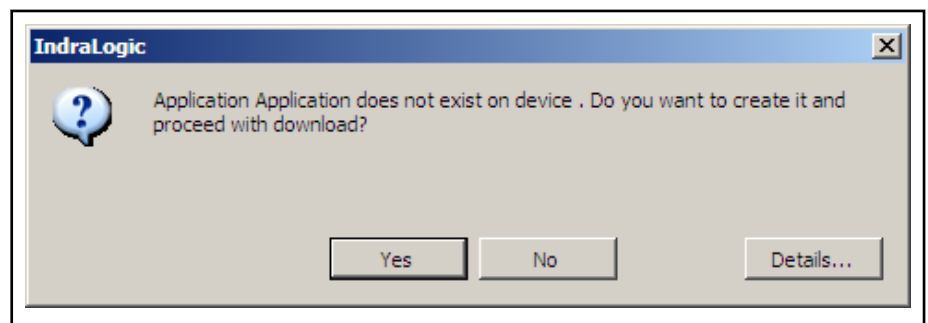


Fig.7-5: "Load program" dialog

Confirm the dialog with **Yes**. You are connected online to your device.

To start the PLC program via the menu bar, select **Debug ► Start 'Application [DCC_Control: PLC logic]**'.

To create a boot project using the menu bar, select **Debug ► Generate boot application**. If this command is executed online, the compiled project is stored on the control so that the control can automatically load it at restart.

To stop the PLC program via the menu bar, select **Debug ► Stop 'Application [DCC_Control: PLC logic]**'.

To log out of the control, select **Debug ► Logout from 'Application [DCC_Control: PLC logic]**.

7.3 Library Management

7.3.1 General Information

Libraries can provide functions, function blocks, data types, global variables and even visualizations which can then be used in the project.

The default extension for a library file is ".library" in contrast to ".lib" which was used for files in previous versions. Encrypted libraries have the extension "*.compiled-library".

PLC Programming

The management of libraries in a project occurs in the library manager. The previous installation on the system is performed using the "Library Repository" dialog.

The project functions for a global and local "find" and "replace" can also be used for libraries that are not encrypted.

See the following general information on:

- [chapter 7.3.2 "Library Overview" on page 196](#)
- [chapter 7.3.3 "Installing on the System and Integrating into a Project " on page 197](#)
- [chapter 7.3.4 "Referenced Libraries" on page 198](#)
- [chapter 7.3.5 "Library Versions" on page 199](#)
- [chapter 7.3.6 "Unique Access to Library Blocks \(Namespace\)" on page 199](#)
- [chapter 7.3.7 "Creating Libraries" on page 200](#)
- [chapter 7.3.9 "Integrating External Code" on page 201](#)

7.3.2 Library Overview

Libraries available for the control:

Name	Description	Documentation ¹⁾
RIL_CheckRtv	Automatic monitoring of internal range exceedances	/33/
RIL_CommonTypes	Data types of the PLCopen function blocks, system-comprehensive	/33/
RIL_Utility	General IL functions and function blocks	/33/
RIL_LoopControl	Controller function blocks	/33/
RIL_Inline	Diagnostic functions for Rexroth Inline modules	/33/ /4/
RIL_ProfibusDP_02	Profibus DP-V1 services, diagnostics, Sync and Freeze	/33/ /4/
RIL_ProfinetIO	Functions for Profinet IO	/4/
RIL_ProfinetIODevice	Functions for the control as Profinet IO device	/4/
RIL_EtherNetIP	Diagnostic and communication services for Ethernet/IP device	/33/ /4/
RIL_EtherNetIPAdapter	Functions for Ethernet/IP adapter	/4/
RIL_SercosIII_IO	Functions for sercos III I/O	/4/
RIL_FieldbusTypes	Internal library for field bus data types	/4/
RIL_NetXLoad	Internal library to operate the Profibus/Profinet interface	
RIH_CMLx	Hardware information CML45/L65 (operating hours, temperature, el. type plate, display output)	/33/
ML_Base	Basic system library	/23/
ML_PLCOpen	PLCOpen function blocks for sercos drives	/23/

¹⁾ See [chapter 1.3 "Required and Supplementing Documentations" on page 14](#)

Name	Description	Documentation ¹⁾
MP_PLCopenFieldbus	PLCopen function blocks for field bus drives	
ML_TechInterface	AxisInterface	/27/
ML_TechBase	Function blocks for basic technology functions	/30/
ML_TechMotion	Function blocks for general Motion applications	/30/
ML_TechRegi	Function blocks for register controllers	/30/
RMB_TechWinder	Function blocks for winders	/30/
MH_HydrControl	Hydraulic central controller	see "Rexroth IndraMotion MLC 12VRS MH_HydrControl" manual
MH_SynchControl	Hydraulic synchronization controller	see "Rexroth IndraMotion MLC 12VRS MH_SynchControl" manual
MH_TechHydrBase	Hydraulic basic functions	see "Rexroth IndraMotion MLC 12VRS MH_TechHydrBase" manual
MH_TechHydrMotion	Hydraulic motion functions	see "Rexroth IndraMotion MLC 12VRS MH_TechHydrMotion" manual
ML_SeqProg	Functions for sequential programming	see "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Sequential Programming" manual
RIL_SocketComm	Functions to communicate via TCP/IP and UDP	/33/
RIL_ModbusTCP	Functions for ModbusTCP-compatible HMI devices	
RIL_Diagnosis	HMI diagnostic functions	
RIL_HMI_Uilities	Function to read HMI M-keys	
IL_VCP_DP	Functions to trigger small operator terminals (VCP) via Profibus DP	/33/

Fig. 7-6: Basic Libraries

7.3.3 Installing on the System and Integrating into a Project

Library repository

Libraries can be managed on the local system in various "repositories" (directories, storage locations). Before a library can be integrated into a project, it has to be installed on the local system in such a repository. This is done in the Library Repository dialog.

¹⁾ See chapter 1.3 "Required and Supplementing Documentations" on page 14

PLC Programming

- Project information** A prerequisite for the installation is that the project information of a library project includes a title, version information and the name of the vendor (company). Optionally, a category designation can be entered that can subsequently be used in the library manager for sorting.
- If there is no category assignment in the project information, the library automatically belongs to the "Other" category. If other library categories are to be used in addition to this default category in libraries created in IndraMotion MLC, these are defined in one or more external XML file(s) "*.libcat.xml" that can also be extended and created again. Such a file can then be called in the "Project Information" dialog to select a category. For further information on the categories, refer to [Creating Libraries, page 200](#).
- Library Manager** Libraries are integrated into a project with the Library manager. In a "default project", it is automatically assigned to the default device first. But it can also be added explicitly in the Project Explorer (below a device or an application) or globally in the "General module" folder. This is done, as for other objects, in the "Add Object" dialog. Therefore, highlight the application and select **Add**
- **Library manager** in the context menu. Libraries integrated into a library are also displayed already preset in the library manager. However "hidden libraries" are also possible, see also [chapter 7.3.4 "Referenced Libraries" on page 198](#).
- If the library is not encrypted and the ".library*" file is also present, the library POU's listed in the library manager can be opened by double-clicking on the respective entry.
- Searching sequence** If a library function is addressed in the project, the libraries and repositories are searched in the order in which they are listed in the library repository, see also [chapter 7.3.6 "Unique Access to Library Blocks \(Namespace\)" on page 199](#).

7.3.4 Referenced Libraries

A library can link other libraries (referenced libraries). The nesting depth is arbitrary. If such a "father" library is integrated into a project, the libraries referenced in it are also available for use. However, only those libraries are available that are integrated into the "father" library in the global "General module" folder!

- Visibility** When creating a library project that references other libraries, it can be defined in the Properties of each integrated library how it has to react later when it is integrated into a project using the "father" library.
1. Its visibility in the library manager, indented below the "father" library, can be deactivated. Thus, "hidden libraries" can be provided in a project.
 2. If a pure "container" library is created, in other words a library project that does not define any function blocks itself, but instead only references other libraries, a subsequent access to its function blocks can be simplified. When a "container" library is integrated into a project, a whole set of libraries is integrated along with it. In this case, simplify the access to the function blocks of these libraries by defining them as "top level" libraries. Subsequently, when accessing the function blocks, the namespace for the libraries can be omitted. To do this, use the "Publish..." option in the library properties. However, this option is only to be used when creating a container library project!

See also [chapter 7.3 "Library Management" on page 195](#).

7.3.5 Library Versions

- Multiple versions of a library can be installed simultaneously on the system
- Multiple versions of a library can be simultaneously integrated into a project. The following clearly specifies on which version an application accesses in this case:

If multiple versions are located on the same level in the library manager, it depends on the definition in the Library Properties which version is to be used (a specific one or always the latest one). If multiple versions are located on different levels (with [referenced, page 198](#) libraries for example), enter the corresponding namespace for a unique access to library blocks (refer to the next section).

7.3.6 Unique Access to Library Blocks (Namespace)

- Basically, the following applies: If several function blocks with the same name are available in the project, access to a function block component has to be unique. Otherwise, compiler errors result. This applies to both local project function blocks and to function blocks available in integrated and in these referenced libraries. In such cases, the uniqueness is achieved by adding the namespace in front of the function block name.
- The default namespace for a library is defined in the Library properties. If it is not explicitly defined, the library name is automatically used. However, when creating a library project, another default namespace can be entered into the "Properties" dialog. Later, when a library is already integrated into the library manager of a project, the namespace can always be changed locally - also in the "Properties" dialog.
- In the following examples, the "namespace" of the library "Lib1" is added to the library properties with "Lib1". In the right column, there are the namespaces for unique accesses to the variable "var 1" defined in the function block "module1" and in the function block "POU1".

	The variable "var1" is defined in the positions (1) to (5) in the project:	Unique access to "var1" using the corresponding namespace information...
(1)	In the library "Lib1" in the global library manager in the "General module" folder	"Lib1.module1.var1"
(2)	In the library "Lib1" in the library manager below an application "App1" of a control "Dev1"	"Dev1.App1.Lib1.module1.var1"
(3)	In the library "Lib1" integrated into the library "F_Lib" (referenced) in the global library manager in the "General module" folder	Preset: (Option "Publish..." is disabled in the library properties of Lib1 when "Lib1" is added to "F_Lib"): "F_Lib.Lib1.module1.var1" If the option "Publish..." were activated, "module1" would be treated like a component of a library integrated at top level. Subsequently, access without entering the namespace of the "father" library "F_Lib" is normally possible: "Lib1.module1.var1" or "module1.var1"). In the present example, however, this leads to a compiler error since the call is no longer unique; see points (1) and (4).

PLC Programming

(4)	In the object "module1" that is defined in the "General module" folder	"module1.var1"
(5)	In the object "POU1" that is defined in the "General module" folder	"POU1.var1"

Fig. 7-7: Namespaces

7.3.7 Creating Libraries

A library can be created and saved from the global "General module" folder. To do this, highlight the "General module" folder and select the menu item **Create Library...** in the context menu. Alternatively, create a library via the main menu **Tools ▶ Create Library....** Consider the following when creating a library:

- To generate a library, some project information has to be specified in the "Library Info" object. Highlight the global "General module" folder and select **Add ▶ Add Library Info Object** in the context menu to add the "Library Info" object. Alternatively, use the mouse to drag the "Library Info" object from the "PLC Objects" library to add it to the global "General module" folder. Double-click the "Library Info" object or use **Open** in the context menu to open the "Project information" dialog.

In Project Information on the "Summary" tab, enter a "title", a "version" and the "company". If the "default namespace" is different to the library name, it can be defined here. Using the option to enter a "category" is also recommended, since it is used later in the library repository dialog and in the library manager for sorting purposes.

If the library does not belong to the default category "Other", load a corresponding category description either from an XML file "*.libcat.xml" or from another library that already contains the information for such a description file. If necessary, create the category description file again or modify an existing file. The content of the description file is transferred to the library project and then to the library repository during installation. Afterwards, the categories are known in the repository. If, afterward, when another library that is installed "contains" a description file with the same identifier but with different contents, the information from the more recent file applies for the entire repository for displaying the categories.

- Note the behavior of [referenced libraries](#), [page 198](#), when the "father" library is integrated into a project. Important are version handling, namespace, visibility and access option configured in the "Properties" dialog of the individually referenced library. If the library is to reference another library depending on the target system, the "placeholder" reference can be set now during definition.
- If the library function blocks are to be protected from viewing and accessing, a library project can be saved in encrypted format (<Project-Name>.compiled-library)
- The data structures of a library can be highlighted as library-internal. These non-public objects have the attribute "hide" and are therefore not displayed in the library management, the Intellisense functionality or the input wizards.
- To easily provide library users information on a function block, a corresponding comment can be added to the declaration of a function block parameter when creating a library project. Later, when the library is integrated into a project, it is displayed in the library manager on a "Documentation" tab

7.3.8 Converting IndraLogic 1.x Libraries

- Libraries created with IndraLogic V1.x (*.lib) and earlier versions are still supported
- An old library project (*.lib) can be read via the library import of the "General module folder" (**General Module Folder ▶ Library Import**) After the import, the library can be saved via **General Module Folder ▶ Create Library...** in the new *.library format
- If an old IndraWorks or IndraLogic project that references old libraries is converted via the menu item **Project ▶ Data Transfer**, it can be selected whether these references are to be retained, replaced or deleted. If they are to be retained, the corresponding libraries are converted into the new format and automatically installed in the system library repository. Rexroth system libraries are automatically detected and converted here. If they do not contain the necessary "project information", it can be immediately added. The mapping, by which an old library was once handled while converting an old project, can be saved in the project options so that the same library does not need to be explicitly processed every time in future project conversions.

For a procedure description to convert projects and libraries, refer to the on-line help under "Data Transfer".

7.3.9 Integrating External Code

External and internal libraries or library function blocks, late linking

- An "external library", in contrast to an internal (IndraMotion MLC library), is a library file that is programmed outside of IndraMotion MLC in another programming language, e.g. C. It has to be on the target system and is only implemented if the application is running there
- As in IndraLogic 1.x, it is also possible to link an IndraMotion MLC library as an external library later on, i.e. when the application is first operated on the runtime system. In addition, it is now even possible to define the late linking individually for every library function block. For this purpose, the property can be enabled in the object properties ("external implementation") of one or all function blocks.

7.4 Task System

7.4.1 General Information

The control can execute several tasks simultaneously. A "job" is called "task". For instance, controlling an axis by a task while another task is performing complex calculations is possible. A PLC task can process one or multiple programs. These are then processed successively by the task in the order of their definition. The PLC programs of a task can be located below the "Application" node of the control as well as below the "General module folder".

Task types

Tasks are divided into several types:

- **Cyclic:** The task is processed cyclically with a fixedly defined time interval (e.g. every 10 ms). This is the usual case for PLC tasks
- **Free running:** The task runs permanently. Once the attached PLC program has ended, it is immediately restarted
- **Event-controlled:** The task is always started when a specific Boolean PLC variable receives a rising edge
- **Externally event-controlled:** The task is always started when a specific event occurs

PLC Programming

Interval Cyclic tasks are started at a defined interval. The runtime of the tasks is to be set smaller than the set interval. If the task is not completed when the interval elapses, it is only restarted in the next interval cycle. This case is to be avoided. It is possible to monitor the task runtime using watchdogs (see ["Watchdog" on page 203](#)).



In order not to unnecessarily increase the CPU load, the cycle times is be selected in such a ways that the application requirements are fulfilled but without executing any undesired processes.

Events An event is either a changing PLC variable (for event-controlled tasks) or an external event named as such (for externally event-controlled tasks).

The following external events can occur in the system IndraMotion MLC:

- **EVENT_OP_MODE_MOTION_CYCLIC:** This event is triggered with each Motion cycle. It is used to synchronize a PLC program with the calculation of axis motions. The "MotionTask" created by default uses this event.
- **EVENT_OP_MODE_SERCOS_CYCLIC:** This event is triggered with every sercos cycle. It can be used to synchronize with sercos I/Os.
- **EVENT_Local_Input_Bitx:** This event is triggered by the onboard inputs of the control. Every input is mapped on Bitx according to its number. A rising edge at the input triggers the respective event (see also [chapter "Onboard and Fast I/Os" on page 216](#)).

Task priority

Only one task can run simultaneously as only one processor is available on the control. Task priority decides about the processing order if several tasks are ready for processing. A low number in the task priority means that this task has a high priority and is thus preferred.

If a task with a higher priority is ready for processing, any other running low-priority tasks are interrupted until the task with the higher priority is completed.

The following figure shows the processing of tasks with different priorities:

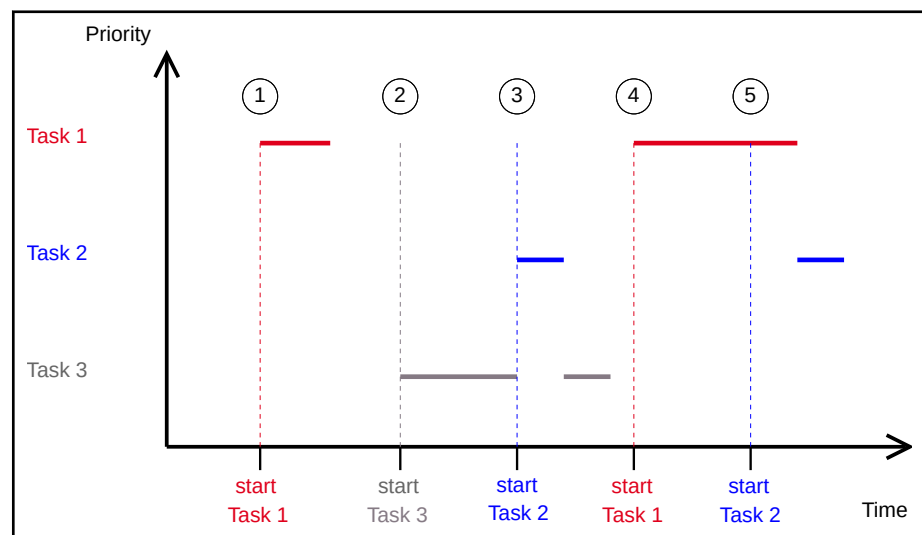


Fig. 7-8: Processing different task priorities

The horizontal lines in the figure above represent the times at which the individual tasks are running.

- ① Task 1 is started and starts to run immediately as it is the only task that is ready for processing at this point.
- ② Task 3 is started and immediately starts to run as well since no other task is running.
- ③ If task 2 is started, task 3 interrupts, as task 2 has a higher priority. Subsequently, task 3 runs until completion.
- ④ Task 1 is started and starts to run immediately as it is the only executable task at this point.
- ⑤ Task 2 is started and cannot start running as the higher priority task 1 is still running. Once task 1 is completed, task 2 can run.



Also ensure that PLC tasks are adjusted to this application. The priority of the different tasks is to be selected according to their processing relevance. This prevents that tasks whose job is irrelevant with regard to time block important tasks that need to be processed in specified cycles.

If several competing tasks have the same priority, the so-called round-robin method is applied. A running task is interrupted after 1 ms and the next task with the same priority becomes active. This takes place for all tasks with the same priority. This method is applied providing no higher-priority task is started or the starting condition of such a task is met.

The following figure shows the processing of several tasks with identical priority:

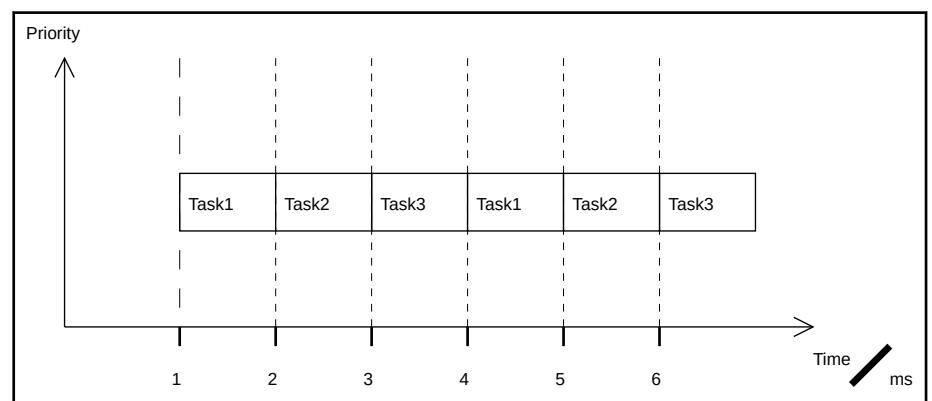


Fig. 7-9: Round-robin method

Tasks 1, 2 and 3 have the same priority. If a task needs more than one millisecond, it is interrupted and the next task is started. This is done until all tasks have been processed.



Assigning the same priorities is only required for a few applications. It is difficult to predict the behavior of the system and is thus to be avoided.

For tasks with the same priority and the same starting condition, call programs in one task if possible.

Watchdog

The watchdog is used to monitor the computing time of a task. It reacts if a task runs longer than defined in the watchdog. Triggering the watchdog always leads to a PLC exception and thus to the stop of the corresponding task. The watchdog is to be set in such a way that the job of the tasks is still

PLC Programming

completed on time. If the value is too small, the system stops unnecessarily. If the value is too high, the watchdog only starts at extreme events such as endless loops in the PLC program.



The watchdog should always be activated and set to the respective time for time-critical tasks.

Alternating memory access

Consistency problems can also occur in multi-tasking systems when other global objects (global variables, function blocks) are accessed by several tasks if the objects exceed the data width of the processor (structures or arrays that form a logical unit). Access to resources has thus to be protected and blocked from simultaneous access by other tasks. As a solution, functions and function blocks of the "SysSem" library system are available.

7.4.2 Task Configuration

General Information

Task configuration defines one or more tasks for controlling and executing the application program on the control.

The task configuration is automatically created by adding the control.



If the task configuration was deleted by mistake, it can be subsequently inserted into the application via the context menu using the **Add ► PLC Object ► Task Configuration** command or by dragging from the library **PLC Objects ► General Objects**.

The uppermost node of a task configuration () is always called "task configuration". Defined tasks are attached below. Each is represented by its name.

The program calls of the individual tasks are not shown in the tree.

Certain types of objects, such as a visualization object create their individual task automatically (VISU_Task).

In the task configuration, the usual commands for editing an object tree can be used to insert tasks and to copy, cut or delete. To add a new task, use the command **Add ► PLC Object ► Task** via the context menu.

Alternatively, add a task by dragging it from the library **PLC Objects ► General Objects**.

The individual tasks are configured in the [task editor, page 207](#), , which also provides an online view. The configuration options depend on the target system.

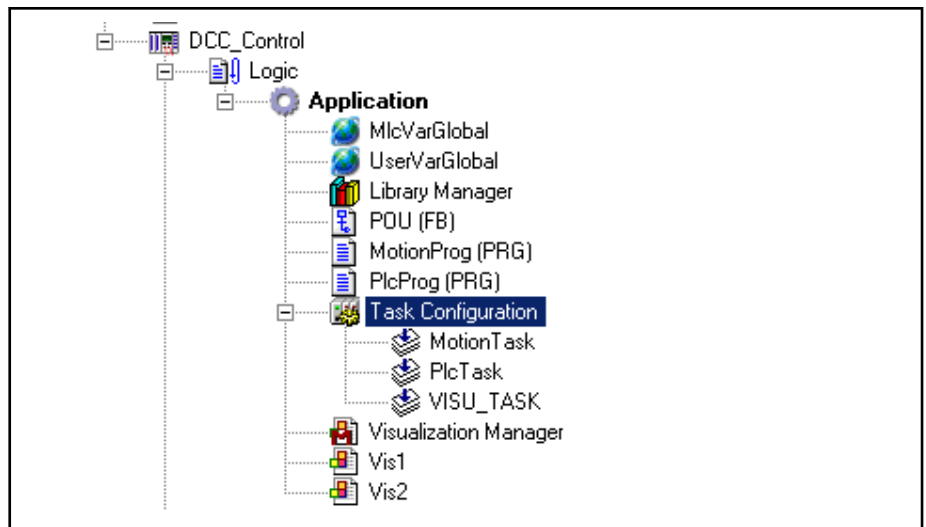


Fig. 7-10: Task configuration below an application in the Project Explorer

In online mode, the "task editor" provides a monitoring view with information on cycles, cycle times and task status.

Default tasks

Two PLC tasks are set up automatically when creating the control:

- **PlcTask:** This task is called cyclically every 10 ms. Time-critical PLC program processes cannot be implemented in this task
The cycle time and priority of this task can be adapted to the needs of the PLC program
- **MotionTask:** The MotionTask is coupled to the calculation of the axis motion. It has very high priority and is therefore only to be used for time-critical tasks that require synchronization with axis motions.
The Motion task provides dead time-optimum behavior for axis-synchronous processes (that is, all dead times related to axis motions are of a minimum).

A task runtime that is too high can lead to the error "Cycle time exceedance" (see [chapter 7.4.9 "System Tasks" on page 217](#)).

The settings of this task are not to be changed

Other PLC tasks can be created any time if required. The maximum number of tasks depends on the control used.

CML L25	CML L45	CML L65
10 tasks	20 tasks	20 tasks

Fig. 7-11: Maximum number of PLC tasks that can be configured

Configuration Examples

Preconfigured tasks are sufficient for the majority of applications. As additional computational effort is required for each task, the number of configured tasks is to be as low as possible. If additional tasks are to be created, it is important to know how they can be integrated into the existing system or how they affect the system behavior.

New PLC tasks can be created using the "Task Configuration".

The following examples illustrate how new tasks can be integrated into the system.

The following figure illustrates the default settings (see [chapter 7.4.9 "System Tasks" on page 217](#)).

PLC Programming

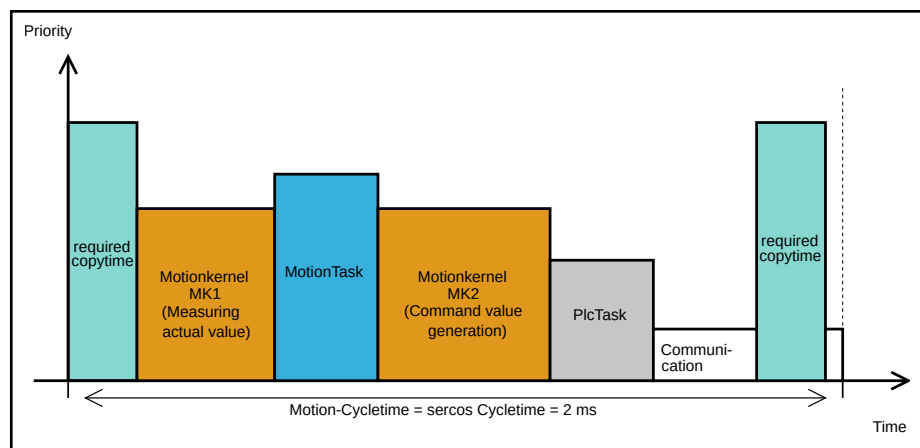


Fig. 7-12: Default setting before adding

Example:

As an example, a new PLC task with the name "NewTask" is added.

This task is configured cyclically. The cycle time is 1 ms.

The task priority is set to 15 and no watchdog is active.

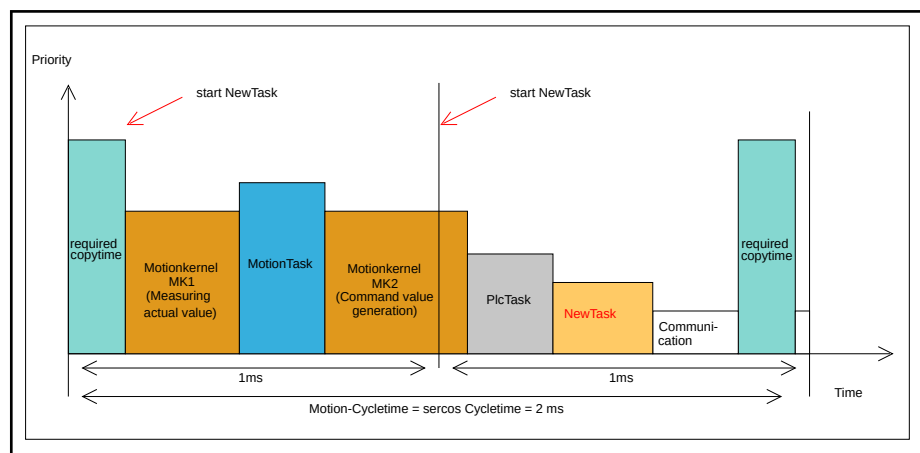


Fig. 7-13: Example 1, creating new PLC task

The "NewTask" is started each millisecond. However, tasks with a higher priority are running at this time. Thus, the "NewTask" cannot run during this period. The "NewTask" can only be processed after all tasks with a higher priority have been completely processed. The "NewTask" has already been started twice up to this point at which the NewTask is processed. Now, it is only running once.



If a PLC task is started several times without being processed, the next time, it is started only once.



If a PLC task is still running when it is supposed to be started again, the start is ignored. The task is processed completely and is not immediately started again.

Example:

The priority of the "NewTask" of example 1 is now set to "1".

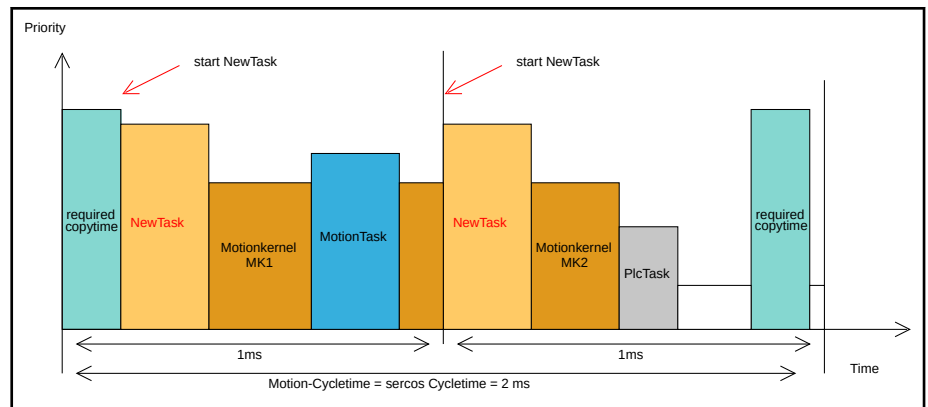


Fig.7-14: Example 2, the priority of "NewTask" of example 1 is now set to 1
The "NewTask" has now a higher priority than the other tasks. Thus, the "NewTask" is processed immediately at the starting times. During the second starting point shown in the figure, the "NewTask" interrupts MK2 until it is completely processed.



The task responsible for copying the real-time data has a higher priority than all other tasks, as it always has to be processed in each cycle. The priority of this task cannot be changed.

Example:

A PLC task with the name "sercosSyncTask" is added to the default tasks. The PLC task is event-controlled configured and has the priority 10. The external event triggered by the task is EVENT_OP_MODE_SERCOS_CYCLIC, i.e. the task is started in each sercos cycle.

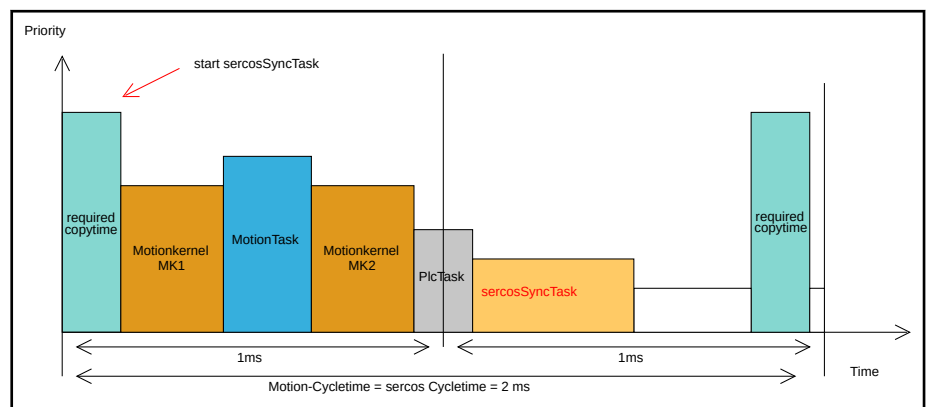


Fig.7-15: Example 3, creating another task

In this example, the sercos and the Motion cycle time are equal. The "sercos-SyncTask" is started at the beginning of the sercos cycle. However, this task can only run if all higher priority tasks have been completed.

7.4.3 Task Editor

Use the task editor to create the task configuration [Task configuration, page 204](#) for an application.

Open the task editor window by double-clicking on a task or by opening the context menu item "Open" of the task.

PLC Programming

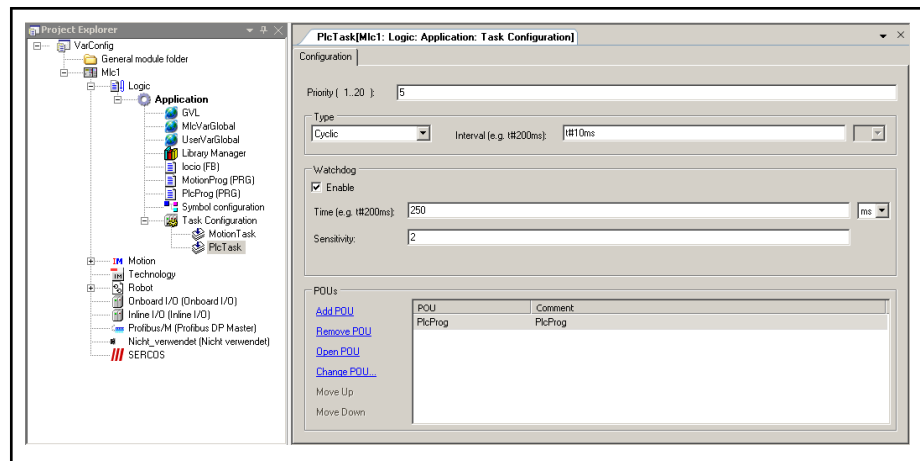


Fig. 7-16: Task editor, dialog to configure a task

The tree structure of the **task configuration** is integrated into the object tree of the Project Explorer. The higher level "task configuration" object and the respective task objects below can be inserted via the **Add ► PLC object** command.

The task editor can be opened by double-clicking on an object in the task configuration. The following configuration dialogs are arranged there as tabs:

- **Properties**, page 208, to display some basic settings of the task configuration
- **Configuration**, page 209, to configure an individual task
- Only if supported by the target system: Profiling to configure (offline mode) and display (online mode) the dynamic analysis of an application task during online operation

The configuration options available in the dialogs depend on the target system used.

7.4.4 "Properties Dialog"

Select the  node with a double-click and the **Properties** dialog opens in the **Task Editor** window.

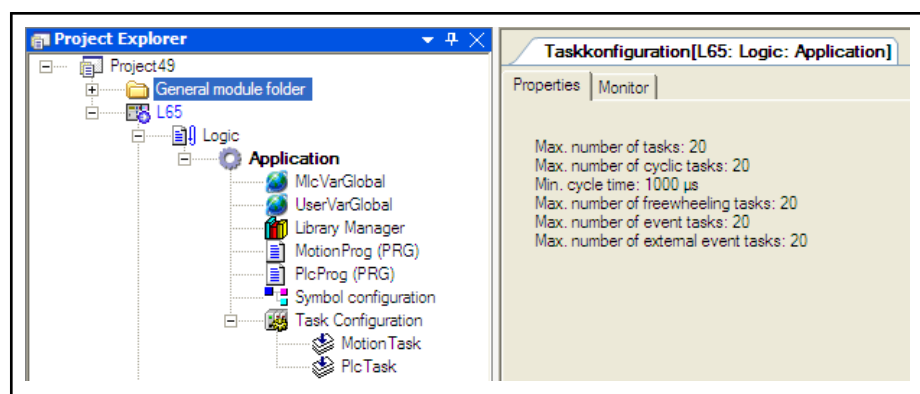


Fig. 7-17: Task configuration, "Properties" dialog, example

This is a display of general settings for task configuration as specified by the target system. For example, the maximum possible number of tasks per task type.

7.4.5 "Configuration" dialog

When a task is added to the [task configuration](#), page 204, (using **Add ► PLC object**), the "Configuration" task editor dialog opens to set task properties:

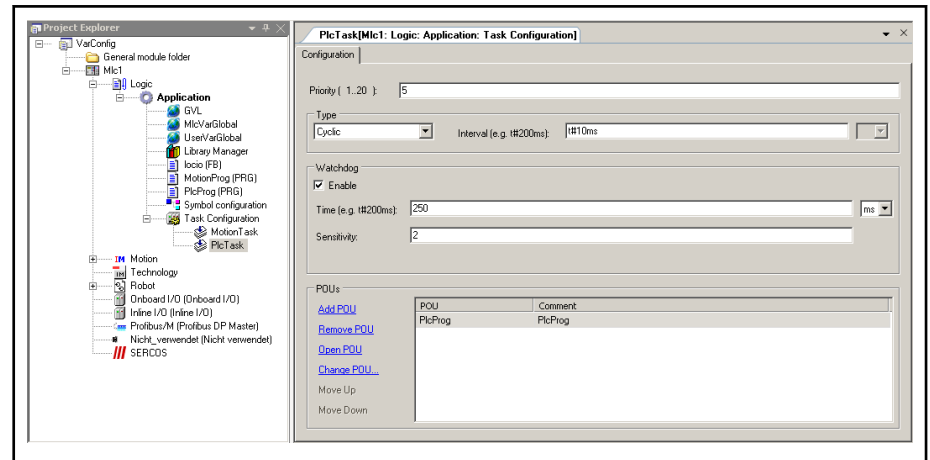


Fig. 7-18: "Configuration" dialog for a task

Define the parameters as desired:



A task can be renamed by editing its name in the Project Explorer.

Priority (1-20): (Number between 1 and 20; 1 is the highest priority, 20 is the lowest).

Type: The selection list provides the following task types:

- **Cyclic:** The task is processed in cycles with a cycle time as defined in the "Interval" field (see below):
- **Free running (FreeWheeling):** the task processing begins at program start. At the end of the run, automatic restart occurs in a continuous loop. No cycle time is defined
- **Event-controlled:** Task processing begins as soon as the variable defined in the "Event" field receives a rising edge.
- **Externally event-controlled:** The task processing begins as soon as the event defined in the "Event" field occurs. The events supported and provided in the selection list depend on the target system. (Do not confuse them with the system events)

Obligatory entries depending on the task type:

- **Interval** (obligatory entry for "cyclic" task types or for an "external event" if the event requires a time specification): Period after which the task is to be restarted. If a number is entered, the unit can be set in the selection box behind the input field: milliseconds [ms] or microseconds [µs]. If the [ms] format is selected, an entry is automatically displayed in TIME format (e.g. "#200ms") as soon as the focus is returned to the window. However, the entry can also be made directly in TIME format. Inputs in [ms] format are always displayed as number (e.g. "250")
- **Event** (for the task type "Event" or "External event"): Global variable triggering the task processing to start as soon as a rising edge is received.

Use the  button for a list of all available global variables

Watchdog:

PLC Programming

For each task, a watchdog can be configured. If the target system supports an extended watchdog configuration, it is possible that upper and lower limits and a default watchdog time are preset as well as a time specification in percent.

- **Active:** If this option is enabled (, see below), the task is aborted with error status as soon as the processing has taken longer than the period defined in the "Time" field (see below)
If the option [Update I/O in stop status, page 124](#), was enabled, the outputs are reset to the defined default values.
- **Time (e.g.: t#200ms):** Watchdog time; after this time period has been expired, the watchdog is enabled irrespective of whether the task has been completely processed. Depending on the target system, the monitoring interval might have to be entered in percent of the task interval. In this case, the drop-down list for the unit is grayed out and displays "%"
- **Sensitivity:** Number of acceptable times the watchdog time can be exceeded without outputting an error



Note that a watchdog can be deactivated for specific PLC cycles using the "CmplecTask.library" library functions. This is useful for cycles that can take longer due to initializations.

POUs:

The function blocks controlled by the task are listed in a table with the "POU" columns (function block name) and an optional "comment". There are commands for editing on the left of the table:

- To define a new program call, use the **Add Call** command to open the input assistance dialog and select the desired program from those available in the project
- To replace one call with another, highlight the entry in the table, open the input assistance with the **Input assistance** command and select another program
- To delete an entry, highlight it in the table and use the **Delete All** command
- The **Open POU** command opens the currently selected program in the respective editor

The arrangement of the function blocks in the list from top to bottom specifies the **execution sequence** of the programs in online mode. Use the **Move Up** and **Move Down** commands to move the currently selected entry in the list.



When configuring the task system, ensure that the available calculation time is used the best possible.

"Optimally" in this case means:

- that the application requirements have to be fulfilled
- that the system capacity should not be unnecessarily loaded

7.4.6 Task List/Configuration

The task list enables the data of all tasks to be seen at a glance and the tasks to be prioritized.

PLC Programming

Open this dialog via the context menu of the control. **Diagnostics ► Task List / Configuration**

Availability

The table is available in the following states; "Online", "Offline" and "Offline parameterization". For offline and offline parameterization, only the data of the PLC task is available. The data of the Motion calculation (corresponds to the "Motion kernel" in previous versions) and the C-tasks (user tasks) (tasks created via "Motion Logic Programming Interface MLPI") are only available online since they are read from the parameters of the control.

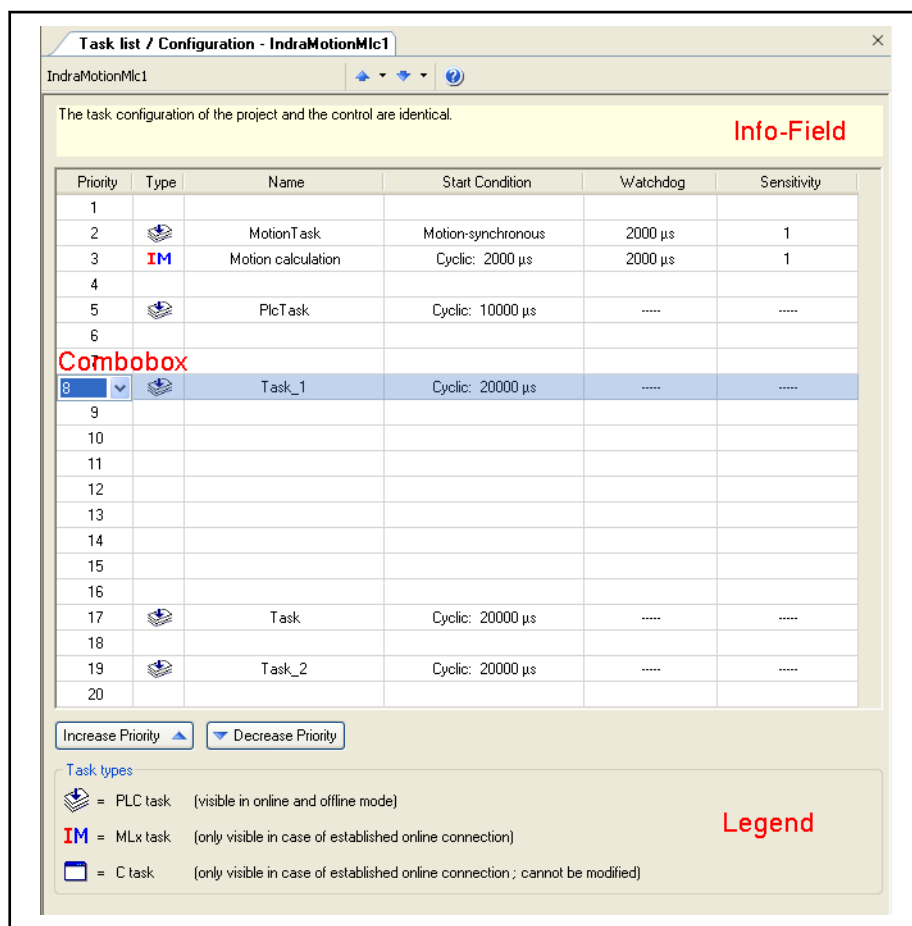


Fig.7-19: Task list

The key below the table shows symbols regarding the respective tasks and a note on visibility and changeability.

Up-to-dateness

The "Info" field above the table shows the current status of the configuration. Data can either be identical to the control data or differ. Additionally, it might not be possible to provide any statement regarding the status. This occurs if the dialog is opened without an established online connection, since the comparison is executed via the parameter.

Activation

If the PLC task configuration in the table is not identical to the control, log in onto the control to transfer this configuration. If already logged in, log out and log in again to activate the configuration. Only the current values are displayed for the C-tasks and the Motion calculation, since they can only be influenced externally or data can be immediately applied.

Changeability

Only the priority of the individual tasks can be changed directly in the table. The C-tasks are an exception, since they are externally created. Therefore, they are only displayed and cannot be changed. The other PLC task data

PLC Programming

(e.g. starting condition) has to be configured via individual dialogs of the tasks (see [chapter 7.4.3 "Task Editor" on page 207](#)). The individual dialog is only available for the PLC tasks. For the Motion calculation, the sensitivity and the cycle time can be modified via the parameters C-0-0-420 and C-0-0-400. There are two options to change the priorities in the table. The priority can be decreased or increased by one using the buttons below the table. The other option is to directly adjust the priority of a task by clicking on the respective cell. The combo box can be activated by clicking. A priority between 1 and 20 can be selected directly in this box.

Operation The combo box is only available for lines in which changeable tasks are located. The same applies for the buttons below the table. If a line without content is highlighted, neither the combo box nor the buttons can be selected. If a task is to be highlighted at the upper or lower end of the table, the respective button for increasing or decreasing the priority cannot be used.

7.4.7 Task Execution Viewer

This dialog shows the chronological sequence of task processing on the control. It can be accessed via the context menu of the control. **Diagnostics ► Task Execution Viewer**

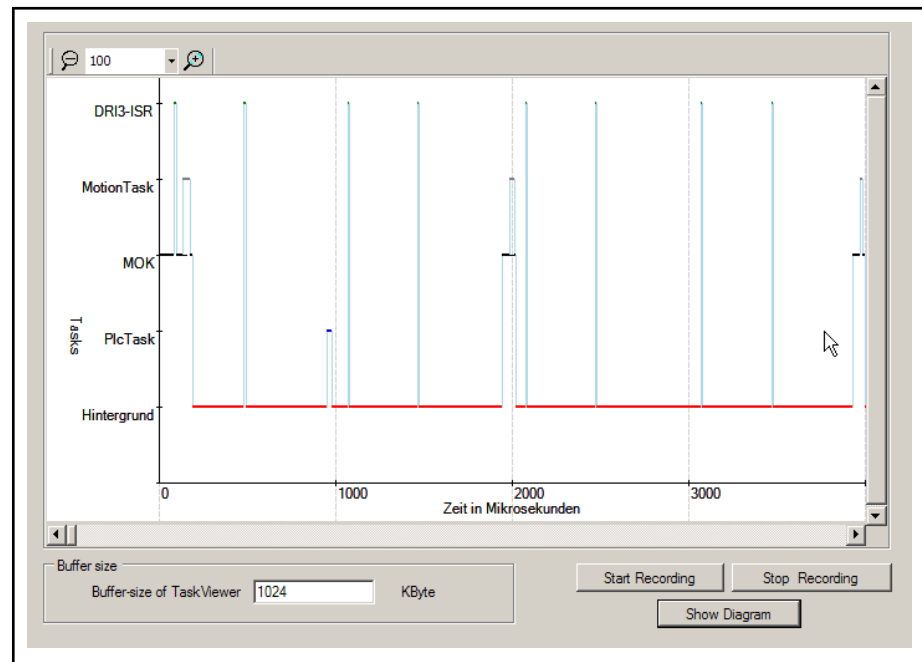


Fig. 7-20: "Task Execution Viewer" dialog

- Availability** The Task Execution Viewer is only available in "online" mode. PLC tasks are only displayed if the PLC program is loaded and running.
- Recording depth** The recording period of the diagram is defined in the "Buffer size" field. The default setting is generally sufficient. The recording is created as ring buffer. The oldest values are lost when the buffer is full.
- Task recording** The recording is started by the **Start Recording** button. The recording then runs until the **Stop Recording** button is clicked. The **Show Diagram** button is pressed to transfer the recorded sequence from the control to the interface.
- Recording analysis** The task diagram is displayed once the recording is completed. The "Task" axis represents task priority. The further up a task is displayed, the higher its priority. The time axis can be scaled via the  and  symbols. The recording can be moved horizontally if required.

Displayed tasks Active tasks are shown by a horizontal blue line. Tasks of a higher priority replace tasks of a lower priority. Once a high-priority task is ended, the next lower-priority task ready for running automatically becomes active. Internal system tasks are processed if no task is ready to run. These are shown by a red line.

The following tasks are displayed in the Task Execution Viewer.

- **PLC tasks:** These are all tasks created in the task configuration (see [chapter 7.4.2 "Task Configuration" on page 204](#)). These tasks are displayed with their name on the "Task" axis
- **MOK:** This task is the Motion kernel of the control. It calculates the axis motions
- **DRI3-ISR:** This task is responsible for the sercos III bus. It has the highest priority
- **Background information** These are all further internal tasks with low priority. These tasks are shown by a red line

7.4.8 I/O Turn-Around Times

General Information

The term "turn around time" describes the time between an event and its resulting response. I/O turn around times are a special kind of turn around times. They are defined by the time span between the setting of an input and the resulting setting of the output.

Inputs and outputs are processed by the PLC. The PLC works according to the IPO principle (Input - Process - Output). Initially, a consistent input image is generated in which all required actual states are read. Thus, the I/O states are recorded at this point. Subsequently, the PLC program is sequentially processed (e.g. setting outputs). At the end of the PLC, a consistent output image is generated and the newly calculated data becomes valid. At this point, outputs are set for example.

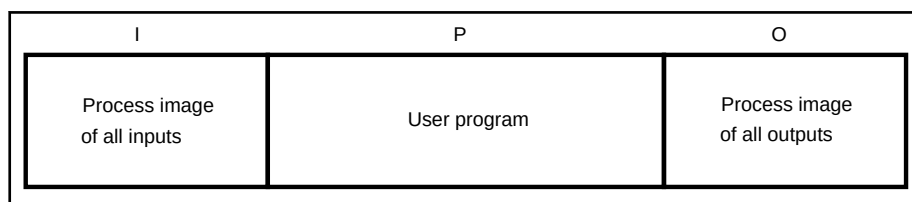


Fig. 7-21: Basic PLC task structure

[fig. 7-22 "Components of I/O turn around time" on page 214](#) shows the different components of the I/O turn around time. The entry is the time span between the setting of the input and the detection of the I/O hardware. The transmission time between the I/O hardware and the control is represented by the communication. The processing time is defined by the time required for calculating an output signal from the transmitted input signal. In the simplest case, it is the copy of the input signal to the output signal. After the output signal has been computed, it has to be transmitted for output (communication). The output time is the required time of the I/O hardware to set the output.

PLC Programming

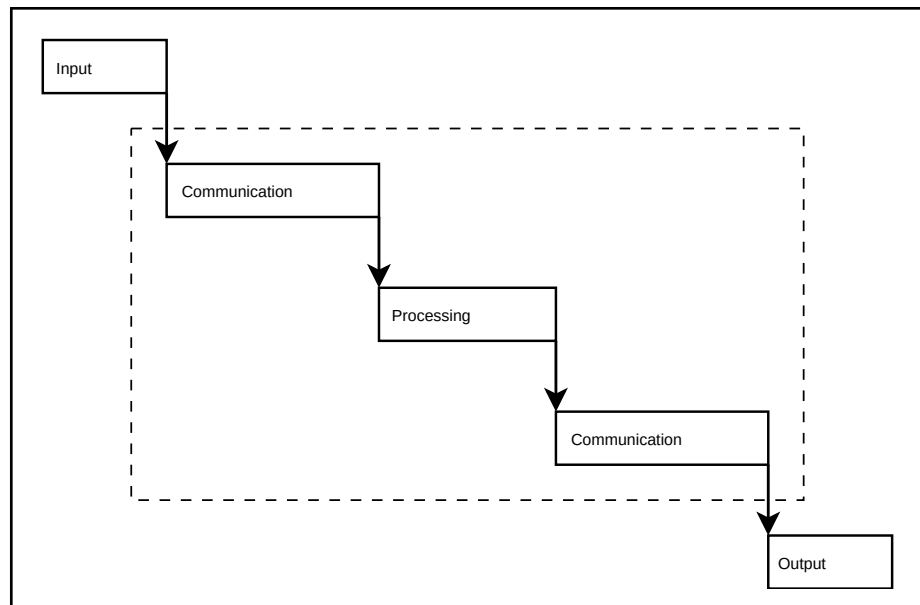


Fig. 7-22: Components of I/O turn around time

When implementing short I/O turn around times, ensure that the times are as short as possible.

However, there are defined times depending on the hardware. Input and output represent the times in the figure above. These delay times filter peak loads. It is filtered to detect only desired inputs or to set outputs. The time depends on the I/O hardware and is given in the respective hardware specification sheet.

The time required for communication depends on the connection of the I/O hardware to the control. This time can be reduced if required (e.g. sercos cycle time).

The processing time depends on how quickly the I/O data is processed by the PLC task in which it is integrated. I/Os are only processed by the PLC (see [chapter 7.4.5 "Configuration dialog" on page 209](#)). If I/O data is in the control, process it as quickly as possible.

After processing, I/O data is to be transmitted to the I/O hardware as quickly as possible.

These times have to be optimized in order to achieve the desired I/O turn around times.

Profibus, Profinet and Inline I/Os

The I/O mapping for Profibus, Profinet and Inline I/Os can be coupled to a PLC task. Inputs are then copied before the PLC task starts. Outputs are copied following the end of the PLC task. A consistent state of the respective field bus is guaranteed at the time of mapping.

There are two options to configure the I/O task:

- **Global:** In the "Configuration" dialog of the control (double-click on the control), it is possible to set the so-called "Bus cycle task" for all field buses in the "PLC Settings" tab. The selected task then executes the I/O mapping for all field buses
- **Field bus mode:** The "Bus cycle task" can be set for Profibus and Profinet in the "Configuration" dialog for the field bus (by double-clicking on the respective field bus node) in the "I/O Mapping" tab. The task set

there then creates the I/O mapping for this field bus. This setting has a higher priority than the global setting

sercos III I/Os

In order to have relatively short I/O turn around times with sercos III I/O, perform the following steps:

- Adjusting sercos cycle time (C-0-0503): The sercos cycle time has to be adjusted to the desired I/O turn around time
- A sercos-synchronous PLC task is to be used to process I/O data (type: externally event-controlled; external event: EVENT_OP_MODE_SERCOS_CYCLIC). The task is started in each sercos cycle and can thus process the I/O data between receiving and sending data. To ensure that the task can run in each sercos cycle, the priority of the task is to be adjusted (see [chapter 7.4.1 "General Information" on page 201](#))

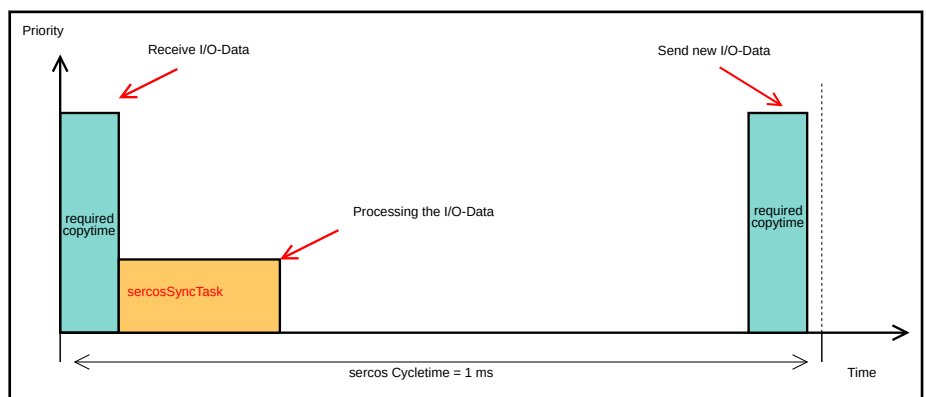


Fig.7-23: Processing I/O data

As shown in the figure, the I/O data has been received at the start of the sercos cycle. This data can now be processed by the PLC task. One or several outputs are to be set if one or several inputs are set. The processed data is sent again at the end of the sercos cycle.

The time required until the data is in the control depends on the sercos cycle time and the time of setting the input with regard to the time of latching the inputs in the I/O hardware.

The following figure illustrates this process:

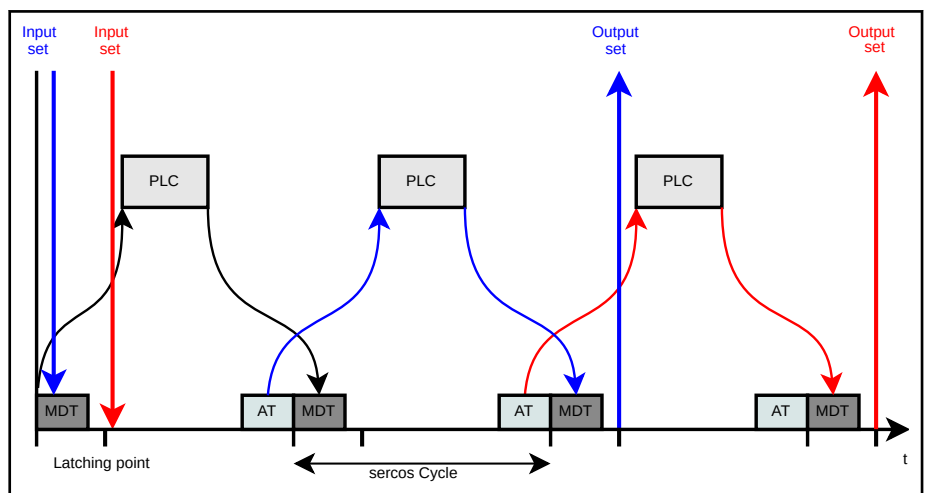


Fig.7-24: Time required for data transfer

PLC Programming

Changes of inputs and outputs become active after the **MasterDataTelegram** as shown in the figure above. There is only one I/O update per cycle. If an input is set after the MDT, it becomes active only in the next MDT and is sent in the subsequent **Acknowledgement Telegram**. This results in a maximum delay of two sercos cycles. The PLC detects that the input is set and the output is also set. The data is sent in the next MDT and the output is enabled immediately after the MDT was received by the I/O hardware. Worst case: the I/O turn around time is three sercos cycles plus filter times (for MDT and AT functionality, refer to "Data Exchange Between Control and Drives"). As a consequence, reducing the sercos cycle time also reduces the I/O turn around time.



If sercos I/Os are configured, data can be exchanged faster by reducing the sercos cycle time. Thus, it is necessary to configure a sercos-cyclic PLC task in which the I/O data is analyzed.

Onboard and Fast I/Os

The IndraMotion MLC L45/L65 already has onboard I/Os and can be extended via function modules. One of the function modules is the Fast I/O module (see /24/ [Required and Supplementing Documentations](#), page 14).

Function modules and onboard I/Os are connected to the control via PCI bus. Therefore, the time required for communication is very low. The filter times are not high either and can be taken from the specification sheet. Processing is the decisive factor for reducing the I/O turn around time.

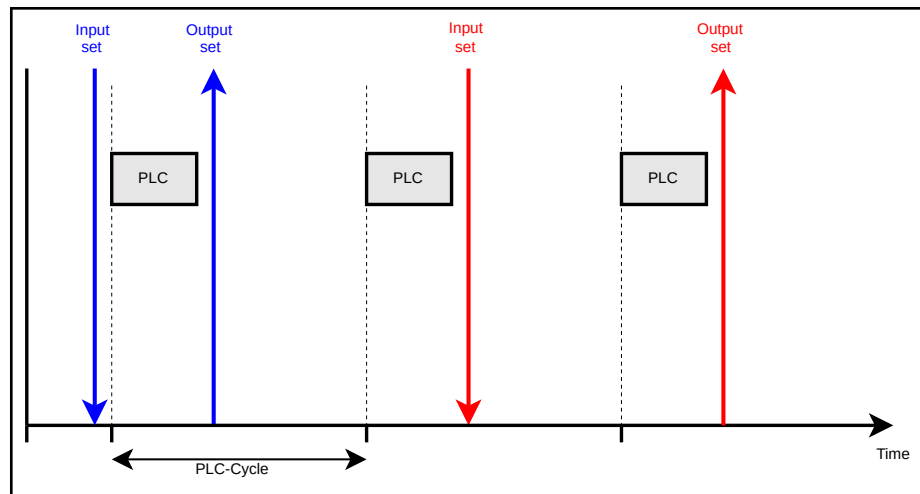


Fig. 7-25: I/O turn around time, Onboard I/O and Fast I/O

As shown in the figure above, the max I/O turn around time is one PLC cycle plus filter times plus the length of the PLC task.

With onboard I/Os, it is possible to directly start a task via an external event if an input is set. In order to start the task via an external event, an externally event-controlled task has to be configured with an external event, e.g. `EVENT_Local_Input_Bit0`. If this task has a higher priority than the other tasks, it is started immediately and the I/O data is processed immediately.

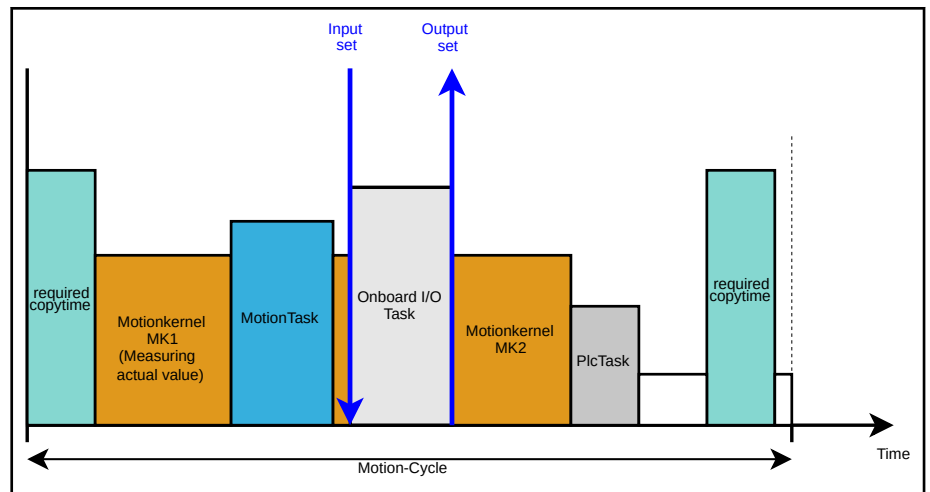


Fig. 7-26: I/O turn around time; task directly started via external event

As shown in the figure, a very short I/O turn around time can thus be achieved.

7.4.9 System Tasks

Several system tasks are used to execute different tasks of the control.

Typical functions of the tasks:

- Communication (e.g. IndraWorks)
- Handling of real-time data of sercos buses
- Collecting and generating real-time data
- Error reaction, etc.

These tasks are already configured in the control and cannot be influenced.

The task responsible for calculating the Motion is the only exception.

In the default setting, an entire Motion cycle runs as follows:

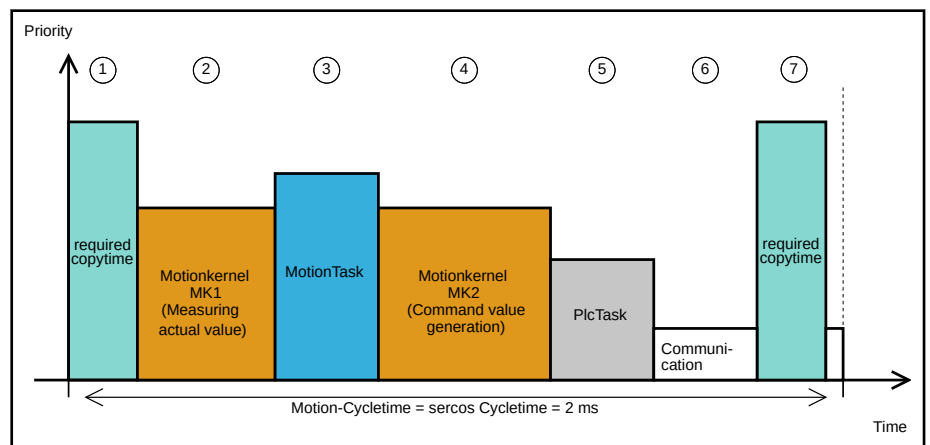


Fig. 7-27: Overview on the temporal behavior

In the default setting, the Motion cycle time (C-0-0400) is equal to the sercos cycle time (C-0-0503) = 2ms.

PLC Programming

-
- ① Data received with the drive telegram (AT telegram) is copied to the local user memory. This is carried out in each sercos cycle and requires some calculation time. The calculation time depends on the number of sercos devices.
 - ② The Motion kernel is started and the actual value detected is executed.
 - ③ The MotionTask is started and interrupts the Motion kernel.
 - ④ The MotionTask is completed. The Motion Kernel continues to run and the command value is generated.
 - ⑤ The PlcTask is processed, as all tasks with a higher priority are now completed.
 - ⑥ PlcTask is completed. Now, low-priority tasks, so-called background tasks, can be processed. These tasks are used for different tasks. One of the most important functions is to control the communication (e.g. IndraWorks).
 - ⑦ Data received with the MDT telegram is copied to the local user memory. This is carried out in each sercos cycle and requires some calculation time. The calculation time depends on the number of sercos devices. If the data has been copied, there is time left to perform background tasks until the start of the next cycle.
-



The figures illustrate this process and do not show the exact temporal behavior of the control. The required task times are considerably lower than shown in the figures!

The cycle time of this task for calculating the Motion, from now on referred to as **Motion kernel**, can be set via parameter C-0-0400. The minimum and maximum cycle times depend on the hardware used. The currently valid cycle time can be read via the parameter C-0-0410. The Motion cycle time has to be equal to or a multiple of the sercos cycle time C-0-0503. These times can be set via the context menu of the control **Cycle Times** of the Motion node (see [Cycle Times](#), page 102).

The priority of the Motion calculation can be changed via the parameter C-0-0403 in the priority table (see [Task List/Configuration](#), page 210). 20 different priorities can be assigned. The Motion kernel and the PLC task have the same priority.

As the Motion kernel is a real-time task, it has to be processed in the set cycle. Thus, it is checked in each Motion cycle if this was the case.

The sensitivity of this check can be changed via the parameter C-0-0420. The error counters can be read via the parameters C-0-0421 and C-0-0422 and can be reset via the command C-0-1052. These parameters are also displayed via the context menu of the control **Diagnostics ► Cycle Times** (this functionality is only available when using sercos III).

The two main tasks of the Motion kernel are:

- **Actual value detection** (part 1 of the Motion kernel, also referred to as MK1)
- **Command value generation** (part 2 of the Motion kernel, also referred to as MK2)

Actual value detection

Recording and conversion of the actual value data of all Motion objects is executed in the first part of the Motion kernel.

Command value generation

The statuses of all Motion objects and the pending Motion commands are generated by the Motion kernel.

For example, the axis status A-0-0021 is generated here.

The second part of the Motion kernel generates and outputs the command value.

MK2 interprets all new Motion commands and processes all Motion commands in each cycle.

If the option "Interpolation in control" is selected (in the drives' "Configuration" dialog), the command values for the drives are calculated.

For example, the operating state switching takes place if new commands were issued.

The calculation time of the Motion kernel depends on the number of configured drives. The currently required time can be read via the parameter C-0-0412. The required maximum time can be read via the parameter C-0-0413 and the required minimum time via the parameter C-0-0414. By executing the command C-0-1050, the minimum and maximum times can be reset.

The times that can be read via the parameters contain the time required to write the bus data into the local memory in each sercos cycle.

These parameters are displayed via the context menu of the control **Diagnostics ► Cycle Times**.

If the Motion kernel is not completely processed in the set Motion cycle, an error response is executed.

This can have two reasons:

1. The time the Motion kernel requires is longer than one Motion cycle.
2. A task with a higher priority was processed before the Motion kernel.

Furthermore, there are two different error types:

1. The **Actual value detection** could not be performed in one cycle.
2. The **Command value generation** could not be performed in one cycle.

In the case of the **Actual value detection**, the error "F5180006 Timeout cycle time of actual value detection" is generated.

In case of the **command value generation**, the error response depends on the set sensitivity of the Motion kernel C-0-0420. Either error F0180008 or error F5180005 is generated depending on fact whether the number is smaller or greater than the set sensitivity.

Error "Timeout cycle time of actual value detection" due to blocking by a higher priority PLC task:

PLC Programming

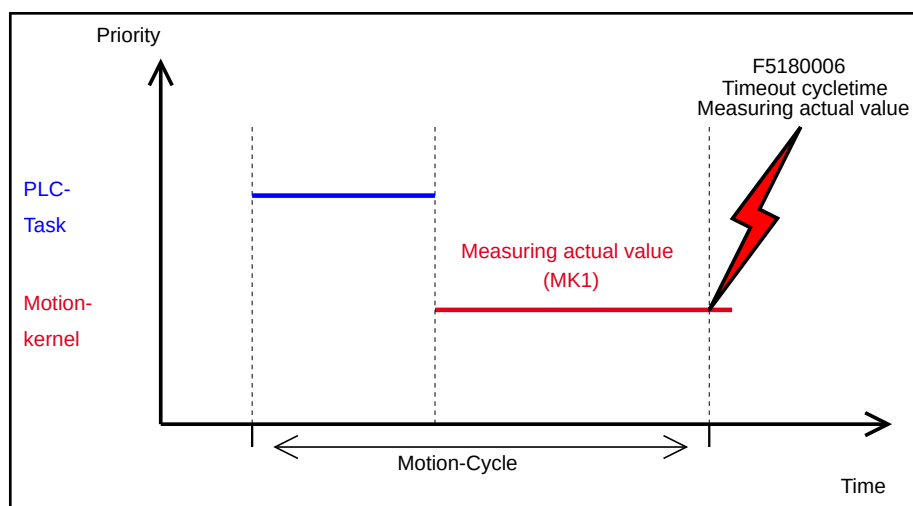


Fig. 7-28: Actual value detection blocked by a higher priority PLC task

Error "Timeout cycle time of Motion kernel" was triggered since the Motion kernel could not be completely processed:

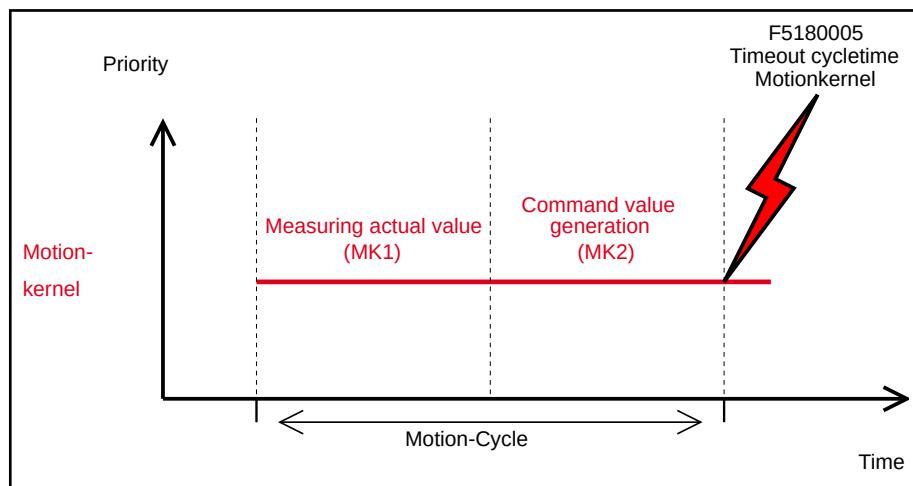


Fig. 7-29: Motion kernel not completely processed

There are several possibilities to adjust the task system to the different requirements.

A correct configuration of the Motion kernel is essential for all applications whose behavior can be influenced in different ways (see [chapter 7.4.6 "Task List/Configuration" on page 210](#)).

Adjust the following settings to the requirements of each application:

- **Cycle time:** The Motion cycle time set via the parameter C-0-0400 specifies in which temporal order the Motion kernel is processed. The shorter the time selected, the higher the CPU load. Consequently, the time in which the Motion kernel has to be processed is also reduced. Due to this reason, ensure that the actually required cycle time is set for the application
- **Priority:** In the set priority of the Motion kernel, the importance of the Motion calculation for an application can be set via the parameter C-0-0403. Each task with a higher priority than the Motion kernel blocks the running Motion kernel (system tasks). For this reason it should be ensured that only tasks whose runtime is not that long and for which it is also necessary (e.g. for short turn around times), are always configured with a higher priority than the Motion kernel.

PLC Programming

- Sensitivity: The Motion kernel sensitivity (C-0-0420) is 1 by default, i.e. an F5 error is triggered when the cycle time is exceeded and all axes are set to standstill (F5 error, error response)



If the sensitivity is increased and an error occurs during the axis motion, the drive calculates the non-existent command value. However, this value can slightly differ from the actual command value. Thus, minor inaccuracies can occur!

Selecting the correct sercos cycle time (C-0-0503) is also important. Reducing the sercos cycle time as well as reducing the Motion cycle time results in a considerable increase of the CPU load, as the processing of real-time data requires time for copying in each cycle.

If there are only drives in the sercos bus, the sercos cycle time is also equal to the Motion cycle time. A faster sercos cycle time than the cycle time of the Motion results in the data being transmitted more often. However, this data is only processed and prepared in the Motion cycle of the Motion kernel.

Configuring the Motion kernel:

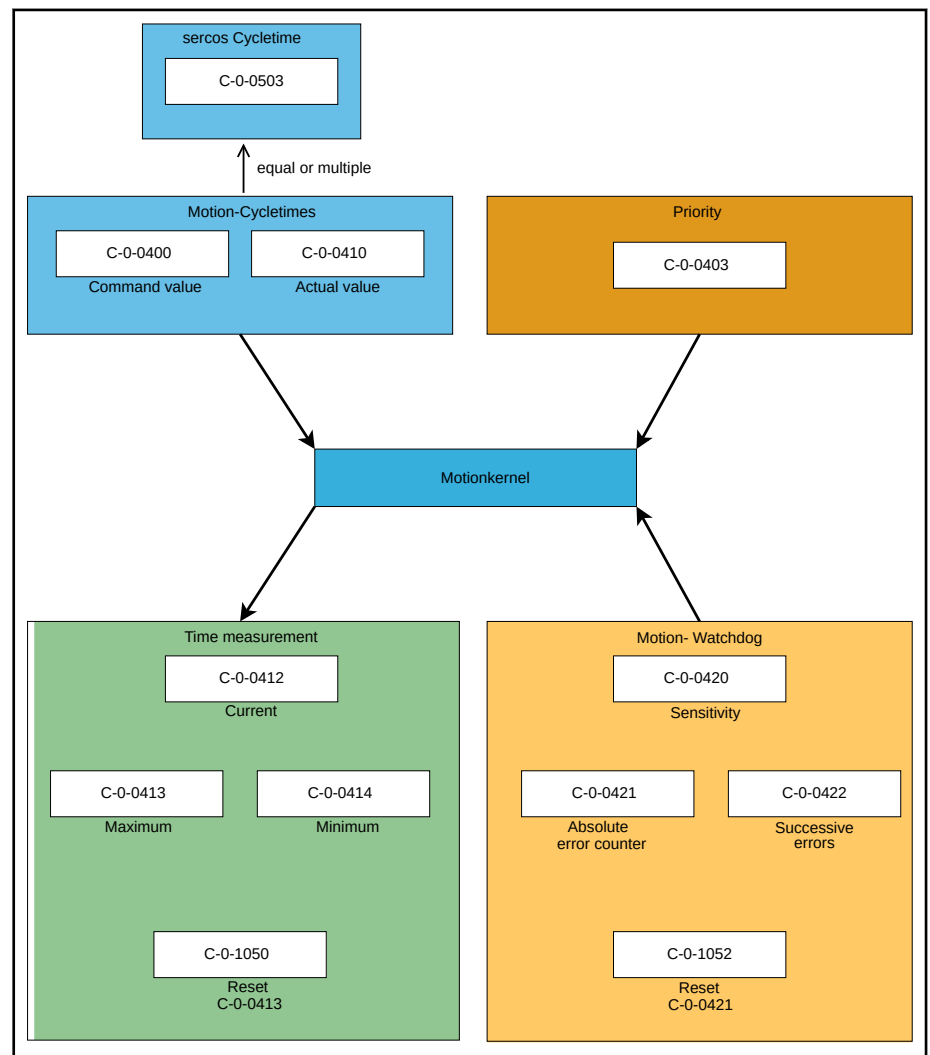


Fig.7-30: Configuring the Motion kernel

PLC Programming

7.5 I/O Accesses

A PLC variable that accesses this address has to be defined to use an I/O address. This process is known as "mapping".

The PLC allows two different methods for a PLC variable to map on an external input or output:

- I/O access via the "AT" operator
- I/O definition via the "I/O Mapping" dialog of the respective I/O device editor

Both variants map a PLC variable on the memory address of an external input or output. When the variable is accessed, the hardware is accessed. All I/O devices are entered into the memory map in the order of their definition in the project. The address of an I/O can be read in the I/O devices "I/O Mapping" dialog.

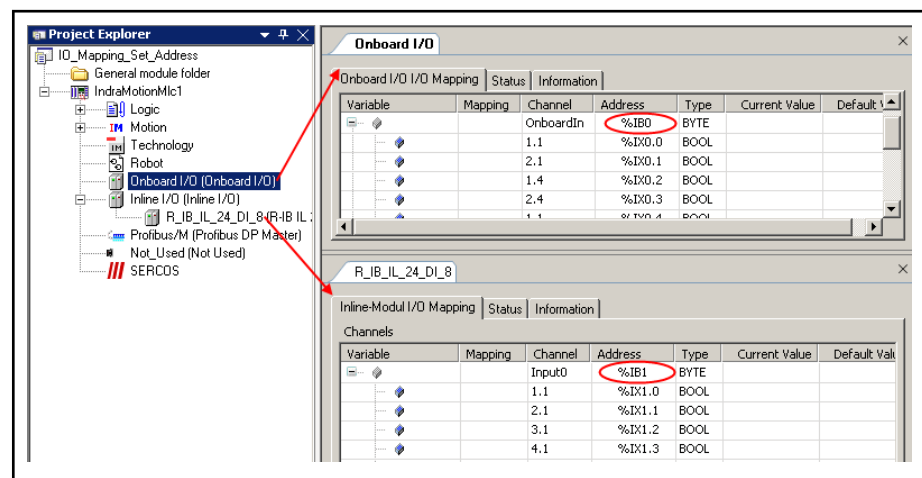


Fig. 7-31: Addresses of an I/O device

The hardware addresses of the I/Os are displayed via the % operator.

Syntax: %<RangePrefix><SizePrefix><number.number>

The following range prefixes are supported:

I	Input
Q	Output
M	Memory location

The following prefixes are supported for the sizes:

X	Single bit
None	Single bit
B	Byte (8 bits)
W	Word (16 bits) [word (16 bits)]
D	Double word (32 bits)
L	Long word (64 bits)

Examples:

%QX7.5 Output bit 7.5

or

%Q7.5

%IW215 Input word 215

%QB7 Output byte 7

%MD48 Double word at memory location 48 in flag area

Declaration via "AT" operator

The variable can be mapped on an I/O address in a variable declaration. This is performed via the "AT" operator.

Syntax: <Identifier> AT %<I|O><I/O address> : <DataType>

Example:

```
VAR_CONFIG
  PLC_PRG.locioVar1.xLocIn AT %IX1.0 : BOOL;
  PLC_PRG.locioVar1.xLocOut AT %QX0.0 : BOOL;
  PLC_PRG.locioVar2.xLocIn AT %IX1.0 : BOOL;
  PLC_PRG.locioVar2.xLocOut AT %QX0.3 : BOOL;
END_VAR
```

Declaration via I/O mapping

The second option for mapping a PLC variable on an I/O is the I/O mapping of an I/O module. In contrast to variable mapping via the AT operator, the variable is not defined in the source text of the PLC program but graphically in the "I/O Mapping" dialog of the corresponding I/O module.

The PLC variable can be specified in the "Variable" column.

The "Mapping" column specifies whether a new variable is declared or mapped on an existing variable:



It is mapped on an existing PLC variable



A new PLC variable is created

It is possible to switch between the two mapping methods by clicking on the symbol.

7.6 Memory Organization and Pointer Programming

7.6.1 Memory Alignment

Memory alignment describes the alignment of data (variables, structures) in the memory.

The alignment methods are distinguished as follows:

- Byte, word, double word alignment:

The variables are aligned independently of the data type to byte, word and double word limits. Empty memory locations are filled with so-called fill bytes. The special case "byte alignment" is also called "packed". With this method, all variables are continuously stored in the memory (without gaps) irrespective of the data type.

- Natural alignment:

The variables are saved in the memory according to the data width. In this method, word variables are always stored on even addresses and double word variables on addresses divisible by 4. If variables or structures containing elements with different data types are created, the compiler adds hidden fill bytes.

Whereas data is stored in a memory-optimized manner, i.e. without gaps (fill bytes) with the "byte aligned" (packed) method, natural alignment presents an improvement in terms of memory access times.



"Natural Alignment" is set as default with the IndraMotion MLC control.

By specifying the compiler pragma "pack_mode" in the variable definition, selected data ranges in the PLC program can optionally also be stored as packed.

PLC Programming

Alignment example The following example illustrates the differences between the two alignment methods "Byte-aligned" (packed) and "Natural-aligned".

Program:

```
STRUCT
    Out01 : BYTE_1;
    Out02 : WORD;
    Out03 : BYTE_2;
    Out04 : DWORD;
END_STRUCT
END_TYPE
```

Address	Packed (byte alignment)	Natural alignment
ADR 10000	Out01 : BYTE_1	Out01 : BYTE_1
ADR 10001	Out02 : WORD (0)	Fill byte
ADR 10002	Out02 : WORD (1)	Out02 : WORD (0)
ADR 10003	Out03 : BYTE_2	Out02 : WORD (1)
ADR 10004	Out04 : DWORD (0)	Out03 : BYTE_2
ADR 10005	Out04 : DWORD (1)	Fill byte
ADR 10006	Out04 : DWORD (2)	Fill byte
ADR 10006	Out04 : DWORD (3)	Fill byte
ADR 10006	-	Out04 : DWORD (0)
ADR 10006	-	Out04 : DWORD (1)
ADR 10006	-	Out04 : DWORD (2)
ADR 10006	-	Out04 : DWORD (3)

Fig. 7-32: Memory allocation depending on alignment

7.6.2 Byte Order

"Byte order" describes the memory organization for simple numerical values (e.g. integer values).

It is distinguished between two basic methods:

- Big endian:
High byte first. This is also called "Intel format".
- Little endian:
Low byte first. This is also called "Motorola format".

Example of byte order Integer value 439,041,101 as 32-bit integer variable from memory address 10000 and higher.

Value in hexadecimal representation: 1A 2B 3C 4D

Value in binary representation: 00011010 00101011 00111100 01001101

Address	Big endian		Little endian	
	Hex	Binary	Hex	Binary
ADR 10000	1A	00011010	4D	01001101
ADR 10001	2B	00101011	3C	00111100

	Big endian		Little endian	
ADR 10002	3C	00111100	2B	00101011
ADR 10003	4D	01001101	1A	00011010

Fig.7-33: Memory allocation depending on the byte order



Numerical values are stored in "big endian" format in the IndraMotion MLC control.

7.6.3 Pointer Addressing



Memory alignment and byte order have to be considered when addressing variables via pointers.

Pointer access to individual variable or array

A pointer to the respective data type of the variables of the respective array elements is always to be used when accessing individual variables or arrays. The byte order is to be noted when using a deviating pointer type, e.g. byte pointer to a word or double word variable.

Example of pointer access

Program:

```
Variable_01 : WORD;
Array_01 : ARRAY [0 to 10] of DWORD;
ptr_Var_01 : POINTER TO WORD;
ptr_arr_01 : POINTER TO DWORD;
ptr_Var_01 := ADR (Variable_01);
ptr_arr_01 := ADR (Array_01)
```

Pointer access to a structure

If a structure has been stored according to the "Natural Alignment" method, addressing structure elements via pointers that are calculated via the basic address of the structure and a corresponding offset is not permitted.

The offset generation to address a structure element is not safely possible, as the basic address of the structure and thus the number of fill bytes can change if there are changes made in the application.

With "Natural Alignment", the compatible use of the structures provides a direct addressing of the elements via the point operator:

Structure name.Element name := Element value;

If the address of a structure is transferred via a pointer to subordinate functions, addressing elements is also only permitted with offset calculation by the compiler. StructurePointer^.ElementName := ElementValue;

Example (Natural Alignment)

Program:

```
TYPE OutStruct :
STRUCT
Out01 : BYTE;
Out02 : WORD;
Out03 : BYTE;
Out04 : DWORD;
END_STRUCT
END_TYPE
StructInst : OutStruct; (* Declaration structure instance *)
pt : POINTER TO StructInst; (* Declaration pointer to structure instance *)
pt2 : POINTER TO BYTE; (* Declaration pointer to BYTE *)
pt^.Out02 := 16#FF00; (* Access to structure element with pointer *)
```

Valid addressing

Program:

```
StructInst.Out03 := 2#11110000; (*Access to structure element *)
pt := ADR (StructInst); (* Pointer initialization *)
pt^.Out02 := 16#FF00; (* Access to structure element with pointer *)
```

PLC Programming

Invalid addressing

Program:

```
pt2 := ADR (StructInst.Out01); (* Pointer initialization *)
pt2 := pt + 3;                 (* Offset calculation for third structure element Out03 *)
pt2^:= 2#11110000;            (* Access to structure element with pointer *)
```

Depending on the address at which the first byte of the structure is located (specified during compilation), there is a differing number of fill bytes for the following elements of the structure. Therefore, it is not reliable to calculate the address at runtime via the basic address of the structure and a corresponding offset.

Byte alignment (packed):

The "Byte-Aligned" (packed) alignment method can be forced with IndraMotion MLC for selected data ranges by a corresponding compiler instruction ("pack_mode" pragma) with structure declaration.

Example of "pack_mode" compiler instruction

Program:

```
{Attribute 'pack_mode' := '1'}
TYPE mystruct:
STRUCT
  Enable   : BOOL;
  Counter  : INT;
  Maxsize  : BOOL;
  MaxSizeReached : BOOL;
END_STRUCT
END_TYPE
```

In this case, the structure is stored as packed in the memory (that is without fill bytes).

With a packed structure, pointer addressing with address calculation at runtime is possible via the basic address of the structure and a corresponding offset.



Packing structures using the compiler instruction {pack_mode} causes the PLC program processing in the IndraMotion MLC to slow down as the access to word and double word operands is always performed byte by byte.

8 Axes

8.1 Introduction and Overview

Axis types

The IndraMotion MLC control provides several axis types. All axis types are provided with a uniform interface with a functional scope differing according to the axis type. The axes are generally accessed via A-parameters.

A differentiation is made between the following axis types:

- **Real axis**
 - **IndraDrive with interpolation in the drive**
Rexroth IndraDrive MPH0x, MPB0x, MPD0x, MPB1x and MPH1x drives. The axis motion is calculated in the drive
 - **IndraDrive with interpolation in the control**
Rexroth IndraDrive MPH0x, MPB0x, MPD0x, MPB1x and MPH1x drives. The axis motion is calculated in the control. Depending on the operation mode, only position or velocity command values are transferred to the drive
 - **HydraulicDrive with interpolation in the control**
Rexroth HydraulicDrive HNC100-3X and IAC drives. The axis motion is calculated in the control. Depending on the operation mode, only position or velocity command values are transferred to the drive
 - **HydraulicDrive with interpolation in the drive**
Rexroth HydraulicDrive HNC100-3X and IAC drives. The axis motion is calculated in the drive
 - **SercosDrive**
PackProfile-compatible drives. The axis motion is calculated in the control and only the position command values are transferred to the drive
- **Virtual axis**
The axis only exists in the control and is not assigned to any real drive. This axis type is often used for virtual axes (vertical shaft)
- **Controller axis**
An axis whose controllers are implemented into the PLC. This can for example be the case at hydraulic axes
- **Link axis**
An axis providing position values of an axis to another control via the sercos II cross link



The link axis is not required/supported in the sercos III cross link.

- **Encoder axis**
Axis only providing actual values (e.g. from an additional drive encoder)

The different axis types provide a variety of functionalities:

Axes

Functions of the axis types

	Real axis	Virtual Axis	Controller axis	Link axis	Encoder axis
Commanding	X	X	X		
Use as master axis	X	X	X	X	X
Use as slave axis	X	X	X		
Use as axis of a kinematics	X	X	X		

Fig. 8-1: Overview on the functional scope of the axis types

Real axes Real axes are divided into the following types:

- **IndraDrive**
The creation of electric drives is described in section [Creating a Real Axis, page 234](#),.
- **SercosDrive**
The creation of a real axis is described in section [Creating a Real Axis, page 234](#),.
- **Hydraulic drives**
The following hydraulic drive designs are supported:
 - HydraulicDrive, HNC
 - HydraulicDrive, IAC
 - HydraulicDrive, central ControllerAxis
 The creation of a real HydraulicDrives axis is described in the section [Creating a Real Axis, page 234](#),.

Virtual Axes The virtual axis can also be used as master axis. It patterns the behavior of a real axis in a mathematically ideal way.

The creation of a virtual axis is described in the section [Creating a Virtual Axis \(Virtual Master Axis\), page 243](#),.

Encoder axes The encoder axis can be used as real master axis.

The data of a real master axis is calculated based on the measuring results of the encoder.

The creation of an encoder axis is described in the section [Creating an Encoder Axis \(Real Master Axis\), page 262](#),.

Link axes with sercos II Cross-Comm

This axis type is only available with sercos II cross link.

Link axes exchange data between several controls. Axis data is transferred via sercos II cross link connected to a sercos II (CFL01.1-Q2) function module.

Each control can define up to two axes (real, virtual, or encoder axes) as globally usable - i.e. usable on the other controls - master axis. The synchronously exchanged data is assigned to a link axis on each of the controls.

The creation of a link axis is described in the section [Creating a Link Axis, page 257](#),.

Link axes with sercos III C2C (only IndraMotion MLC)

This axis type is not required/supported for the IndraMotion MLC with sercos III C2C.

Axes

For all axes provided (produced) in the link, the data is summarized as "consumer axis" in an array on each link device in the "ML_Base" library.

Data of the "consumer axis" can be accessed in the PLC program via the array "GV_LinkAxis[]"

This axis type is neither defined nor displayed in the user interface.

Maximum number of axes

Different maximum stages of expansion are possible depending on the control and the field bus used. Depending on the control variant, different field buses are used.

Hardware	sercos II	sercos III
IndraControl L25	16*	16
IndraControl L45	32*	32
IndraControl L65	42*	64

* Only with function module sercos II (CFL01.1-Q2)

Fig. 8-2: Maximum number of axes for each control

Field buses for real axes

Real axes can be controlled via the field buses sercos II, sercos III and Profibus. To connect via sercos II and sercos III, dialogs are available in the interface.

sercos II

sercos II is an optical fiber-based field bus with a maximum transmission rate of up to 16 MBit/s.

sercos III

sercos III is based on the Ethernet and transmits a data rate of 100 MBit/s.

Profibus DP

Profibus is supported with a twisted two-wire line. The data rate is up to 12 MBit/s.

Supported function blocks	Real axes					Virtual axes	Encoder / link axis	Controller axis
	IndraDrive		HydraulicDrive		Sercos-Drive			
	IPO Drive	IPO Control	IPO Drive	IPO Control				
Function block								
MB_ChangeCamData	X	X	-	X	X	X		X
MB_ChangeFlexEventSet	-	X	-	X	X	X		X
MB_ChangeFlexProfilSet	-	X	-	X	X	X		X
MB_ChangeProfilSet	-	X	-	X	X	X		X
MB_ChangeProfilStep	-	X	-	X	X	X		X
MB_GearInPos	X	X	-	X	X	X		X
MB_Home	X	X	X	X	X	X		X(3)
MB_MotionProfile	-	X	-	X	X	X		X
MB_Phasing	X(1)	X	X	X	X	X		X
MB_PhasingSlave	X(1)	X	X	X	X	X		X
MC_CamIn	X	-	-	-	-	-		-
MC_CamOut	X	-	-	-	-	-		-

Axes

Function block	Real axes					Virtual axes	Encoder / link axis	Controller axis
	IndraDrive		HydraulicDrive		Sercos-Drive			
	IPO Drive	IPO Control	IPO Drive	IPO Control				
MC_GearIn	X(2)	X	-	X	X	X		X
MC_GearOut	X(2)	X	-	X	X	X		X
MC_MoveAbsolute	X	X	X	X	X	X		X
MC_MoveAdditive	X	X	X	X	X	X		X
MC_MoveRelative	X	X	X	X	X	X		X
MC_MoveVelocity	X(2)	X	X	X	X	X		X
MC_Power	X(2)	X	X	X	X	-		X
MC_Reset	X(2)	X	X	X	X	X		X
MC_Stop	X(2)	X	X	X	X	X		X
MC_TorqueControl	X(1)	X	X	X	X	-		X
ML_FlexProfile	-	X	-	X	X	X		X
ML_OpenCyclicAnalogChannel	-	-	X	X	-	-		-
ML_OpenCyclicPositionChannel	X	X	X	X	X	X		X
ML_OpenCyclicTorqueChannel	X	X	X	X	-	-		X
ML_OpenCyclicVelocityChannel	X	X	X	X	-	-		X
ML_WriteCyclicAnalog	-	-	X	X	-	-		-
ML_WriteCyclicPosition	X	X	X	X	X	X		X
ML_WriteCyclicTorque	X	X	X	X	-	-		X
ML_WriteCyclicVelocity	X	X	X	X	-	-		X
ML_ControlOff	X	X	X	X	X	X		X
ML_ControlOn	X	X	X	X	X	X		X
ML_ControlWriteActualData	-	-	-	-	-	-		X
ML_OpenCyclicAnalogChannelCtrl	-	-	X	X	-	-		-
ML_OpenCyclicPositionChannelCtrl	X	X	X	X	X	X		X
ML_OpenCyclicTorqueChannelCtrl	X	X	X	X	-	-		X
ML_OpenCyclicVelocityChannelCtrl	X	X	X	X	-	X		X
ML_WriteCyclicAnalogCtrl	-	-	X	X	-	-		-
ML_WriteCyclicPositionCtrl	X	X	X	X	X	X		X
ML_WriteCyclicTorqueCtrl	X	X	X	X	-	-		X
ML_WriteCyclicVelocityCtrl	X	X	X	X	-	X		X

(1)

Not MPx03

(2)

Also with OpenLoop

(3)

Depends on the implementation of the controller

Fig. 8-3:

Supported function blocks in an IndraMotion MLC

8.2 Real Axis

8.2.1 General Information

The real axis maps physically existing drives at the machine. The group of real axes is divided again and listed according to the hardware used or the mode of operation of the hardware is to be operated.

It is distinguished between:

- **SercosDrive**

The functional scope of the sercos interface specified in the sercos PackProfile (see <http://www.sercos.com> under Technology ⇒ sercos Packaging Profile) is used as interface between drive and axis. sercos PackProfile defines an application profile for the continuous interoperability in packaging machines. This operating mode of the real axis ensures the highest compatibility possible with the very different drive controllers and allows the operation of drive controllers from third-party manufacturers also supporting the sercos PackProfile. When operating as SercosDrive, the drive is only controlled in all operation modes and motion commands via the cyclic position command value interface in the position control. The command value is centrally interpolated by the control. Device-specific dialog are not displayed.

- **IndraDrive, interpolation in the drive**

Devices of the IndraDrive family are provided with a bigger performance and functional scope as specified in the sercos PackProfile. For an optimum hardware support, the control provides an individual operating mode for IndraDrive devices. This is characterized for example by a comprehensive telegram structure, additional operation modes (velocity control, torque control), improved diagnostic options and adapted dialog support in IndraWorks. The command values are calculated in this operating mode by the drive controller itself, that is a decentral calculation. The control forwards the commanded motion commands to the drive.

Based on IndraDrive (function package: "Synchronous package") real axes can be created using the following firmware:

- FWA-INDRV*-MPH-06VRS-D5
FWA-INDRV*-MPD-06VRS-D5
FWA-INDRV*-MPB-06VRS-D5
FWA-INDRV*-MPC-06VRS-D5
- FWA-INDRV*-MPH-07VRS-D5
FWA-INDRV*-MPD-07VRS-D5
FWA-INDRV*-MPB-07VRS-D5
- FWA-INDRV*-MPH-08VRS-D5
FWA-INDRV*-MPD-08VRS-D5
FWA-INDRV*-MPB-08VRS-D5
- FWA-INDRV*-MPB-16VRS-D5
- FWA-INDRV*-MPB-17VRS-D5
FWA-INDRV*-MPM-17VRS-D5
FWA-INDRV*-MPC-17VRS-D5

It is also distinguished between the drive is to be operated with (closed loop) or without (open loop) position feedback.

Axes

Without position feedback (open loop), the PLCOpen function blocks are only provided with limited support (only speed control possible), see [fig. 8-3 "Supported function blocks in an IndraMotion MLC" on page 229](#). In this case, only a limited number of axis parameters is synchronized with the drive parameters.

- **IndraDrive, interpolation in the control**

This operating mode combines efficiency and simplicity of the central interpolations in the control with the advantages and the extended functional scope of the IndraDrive hardware. The command values are interpolated for all operation modes in the control. Depending on the motion commands, it is commanded to the drive as cyclic position, velocity and torque command values via a telegram adapted to the IndraDrive.

The motions in the drive are exclusively specified via the operating states.

- "Closed-loop position control with cyclic command value specification"
- "Velocity control with cyclic command value specification"
- "Torque/force control"

Based on IndraDrive (function package: "Synchronous package") real axes can be created using the following firmware:

- All drives listed under the point "IndraDrive, Interpolation in the drive"
- FWA-INDRV*-MPE-16VRS-D5
- FWA-INDRV*-MPE-17VRS-D5

It is also distinguished between the drive is to be operated with (closed loop) or without (open loop) position feedback.

Without position feedback (open loop), the PLCOpen function blocks are only provided with limited support (only speed control possible), see [fig. 8-3 "Supported function blocks in an IndraMotion MLC" on page 229](#). In this case, only a limited number of axis parameters is synchronized with the drive parameters.

- **HydraulicDrive, interpolation in the drive**

Since this is also an IndraDrive-based device, the same characteristics as described above apply for this hardware. The modulo operation of the axis as well as the synchronization modes are not supported.

HydraulicDrive axes HNC100-3x Compact (function package: "Servo package") with the following firmware

- FWA-HNC3X*-HDH-06VRS-D5-1SRV-NN

IAC-R axes with firmware:

- FWA-HYDRV*-HDB-16

- **HydraulicDrive, interpolation in the control**

Since this is also an IndraDrive-based device, the same characteristics as described above apply for this hardware. The modulo operation of the axis is not supported. Thus, it can only be translatorily scaled.

HydraulicDrive axes HNC100-3x Compact (function package: "Servo package") with the following firmware

- FWA-HNC3X*-HDH-06VRS-D5-1SRV-NN

IAC-R axes with firmware:

- FWA-HYDRV*-HDB-16



Whether the real axis

- is to be interpolated in the drive or in the control,
- is to be operated with (closed loop) or without position feedback

has to be specified when creating the axis

Motion functionality

	IndraDrive / HydraulicDrive	IndraDrive / HydraulicDrive	SercosDrive
	IPO Drive	IPO Control	
Position limit value monitoring in the control	Yes	Yes	Yes
Position limit value monitoring in the drive	Yes	Yes	No
Jerk limitation	Implemented as filter	Analytical in path calculation	Analytical in path calculation
Scaling	Can be freely selected	Preference scaling only	Preference scaling only

Fig. 8-4: Overview on the Motion functionality

Supported Motion function blocks:



The support of the PLCopen function blocks depends on the axis types.

For an overview, refer to the table [fig. 8-3 "Supported function blocks in an IndraMotion MLC" on page 229](#).

Drive operating states used

	IndraDrive	HydraulicDrive	IndraDrive / HydraulicDrive	SercosDrive
	IPO Drive	IPO Drive	IPO Control	
Torque/force control	Yes	Yes	Yes	No
Velocity control	Yes ¹⁾	Yes	Yes ¹⁾	No
Closed-loop position control with cyclic command value specification	Yes	Yes	Yes	Yes
Drive-controlled positioning	Yes	Yes	No	No
Velocity synchronization with virtual master axis	Yes ¹⁾	No	No	No
Phase synchronization with virtual master axis	Yes	No	No	No
Cam table with virtual master axis	Yes	No	No	No

Axes

MotionProfile with virtual master axis	Yes	No	No	No
Valve direct control	No	Yes	No (only Hydraulic-Drive)	No

¹⁾ Also possible for IndraDrive without position feedback (open loop)
Fig. 8-5: Overview on the drive operating states used

8.2.2 Error Reaction of the Axis

Set the axis parameter A-0-0651 Bit 0 = 1 and all drive errors of state class 1 (F2...F8) become the control error F00B0165. There is no error reaction from the control.

For details, refer to the parameter description A-0-0651 in the documentation "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Parameters" and [chapter 8.7.13 "Error Reaction" on page 285](#)).

8.2.3 Creating a Real Axis



Axes can only be created in offline mode!

All real axes (IndraDrive, HydraulicDrive and SercosDrive) can be created the same way. A real axis is created according to the drag&drop principle. Drag the required device "IndraDrive" from the library and

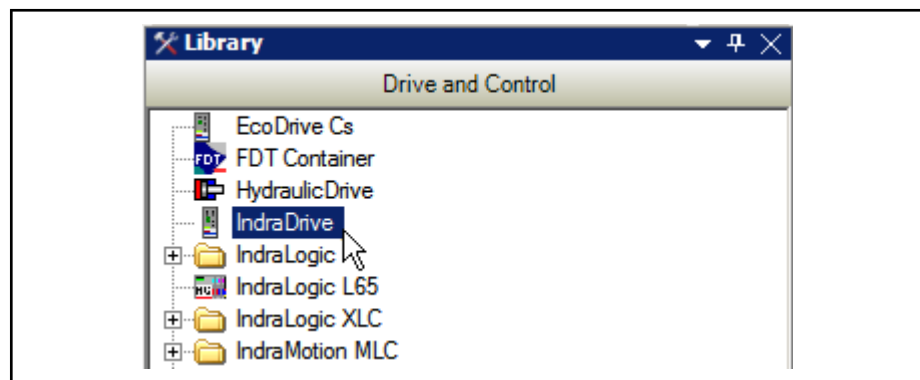


Fig. 8-6: Creating a real axis - Dragging from the library
 drop it to the desired bus, in this case sercos.

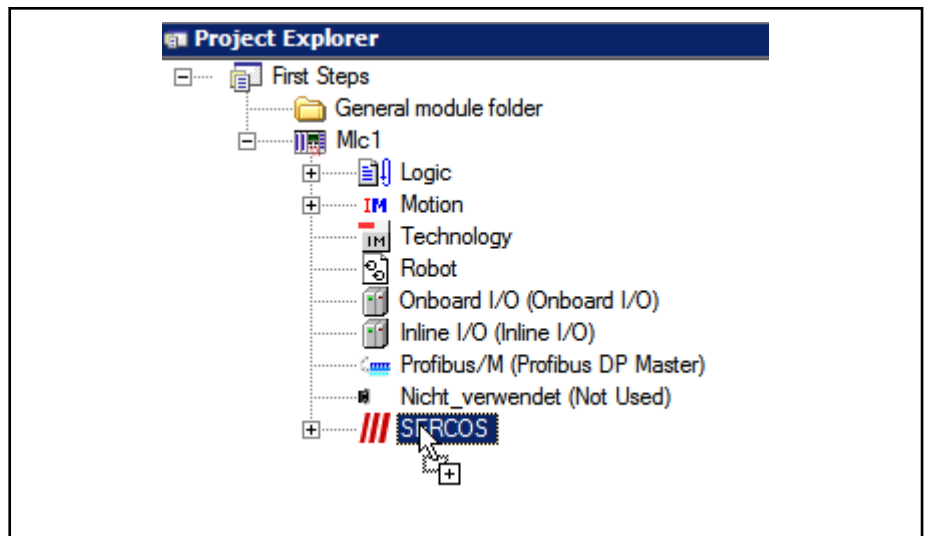


Fig.8-7: Creating a real axis - Dropping to the Project Explorer

Dialogs to create a real axis

The creation of a real axis is supported by a wizard with two dialog boxes. These windows look similar for all drive types.

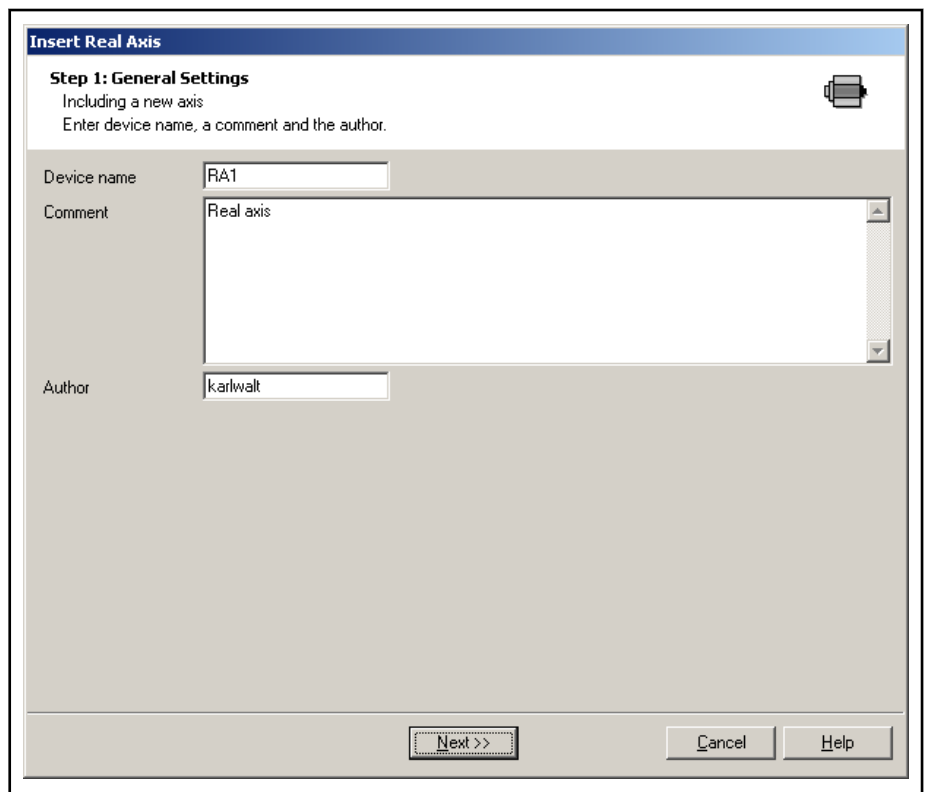


Fig.8-8: Creating a real axis (IndraDrive) - General Settings

In the "General Settings" step, the device name, a comment on this device and the author are entered.

Apply or change the preset values.

Axes

Insert Real Axis

Step 2: Configuration
Configuration of the new axis
Enter the axis-specifically settings.

Axis type

- ☒ Real axis
- ☐ Virtual axis
- ☐ Encoder axis
- ☐ Link axis
- ☐ Controller axis

Configuring real axis/axes

Firmware: FWA-INDRV*-MPH-08VRS-D5

	Nr.	Adr.	Achsname	Ipo Antrieb	Closed Loop	Erw.-paket
Achse	1	1	Drive1	☒	☒	SNC

Device details

Drive configuration

☐ Start wizard to configure the offline parameterization

<< Back Finish Cancel Help

① It provides information on the checkbox "Interpolation in drive"

Fig. 8-9: Creating a real axis (IndraDrive) - Configuration

Configuration via offline parameterization

The system provides a location (red arrow in the figure) to the "Offline Wizard" to configure a not yet existent drive offline.

Since the default values are kept, the initial entry can only be changed by reparameterization.

Configuration of the real axis

The "Firmware" setting allows to select from two different software states, here MPH-07. Permitted firmwares are listed in [chapter 8.2.1 "General Information" on page 231](#).

In the configuration step, the address of the drive in the sercos bus **must** be set (address in the drive configuration, in this case 1). Otherwise, communication with the drive is not possible.

The "Interpolation in drive" leads to the drive automatically calculating its position command values for the operating state selected in each case, for instance synchronization or cam. Otherwise, the control provides the drive with position command values.

If the checkmark is not set, the control provides position command values to the drive. This is a default setting for sercos and EcoDrives.

The "Closed Loop" option is used for an axis with position feedback. If the checkbox is deselected, an axis without position feedback ("OpenLoop") is used.

In this case (open loop), only a limited functional scope is available (position-controlled operation modes are not possible; axis parameters for position-controlled operation modes are not synchronized with the drive).



The type of drive and its sercos address(es) can be subsequently changed via the "Changed drive/firmware" function in the context menu of the real axes (see [Exchanging the Drive/Firmware](#) , page 329)!

Drive configuration

The drive has to be configured before starting the offline parameterization for a drive.

If the "Start wizard to configure the offline parameterization" option is checked, a dialog in which the required data can be entered is displayed once the real axes have been created. The configuration for offline parameterization is generated from the entries.

Fig.8-10: Creating a real axis (IndraDrive) - Configuration of the drive

If this option is not selected, no offline parameterization is now possible for the drive. It can be subsequently configured as follows:

1. Call the "Update offline parameterization" function of the context menu of the control at an established online connection.
2. Call the "Implement configuration for offline parameterization" in the "Properties" dialog of the real axis.

Software states and function packages

Axis assignment

Refer to [chapter 8.2.1 "General Information"](#) on page 231.

To address the axis, the PLC uses the axis number and axis name (here "1" or "RA1").

Axes

Overview in the Project Explorer

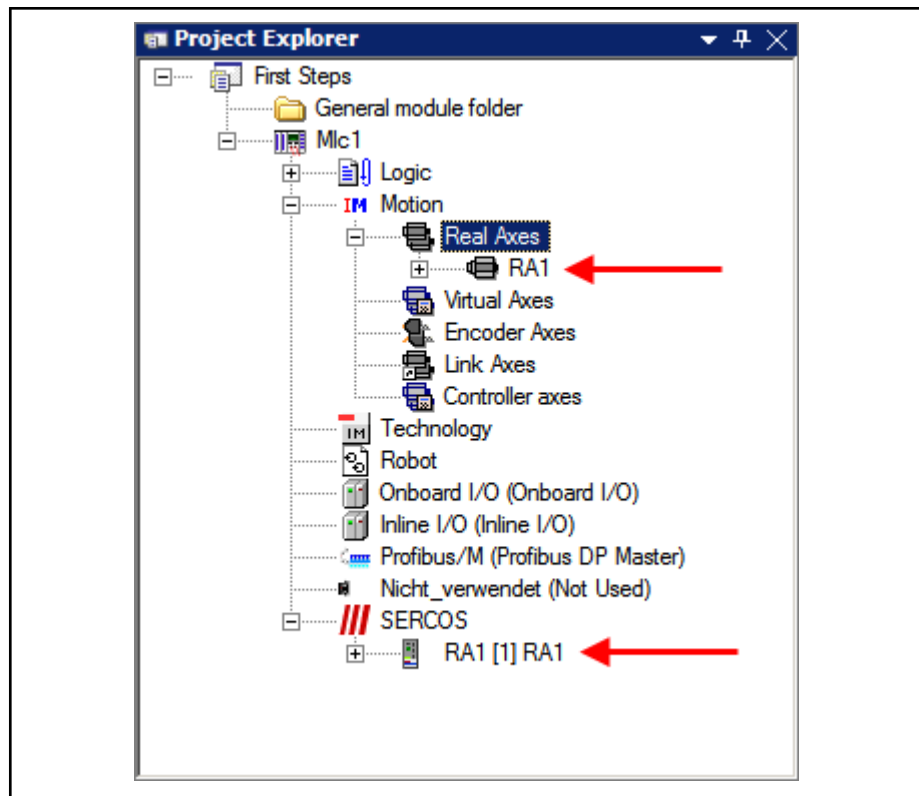


Fig.8-11: Creating a real axis - Overview in the Project Explorer

The real axis (IndraDrive) is displayed twice in the Project Explorer:

1. **Motion ► Real Axes ► RA1**, the axis dialogs are enabled in online mode or in offline parameterization and described in [Real Axis – Dialogs, page 238](#).

A-parameter values are stored in the control and overwrite the respective S- and P-parameters in IndraDrive. S- and P-parameters are only stored in the drive.

The values indicated in the A-parameters are derived from the respective S-parameters and P-parameters if required.

2. **sercos ► RA1 [1] RA1**, here **additional** drive dialogs are enabled in online mode or in offline parameterization.

S- and P-parameters are only stored in the drive and are thus directly displayed.



An offline parameterization is currently not possible for Hydraulic-Drive.

The respective axis is automatically deactivated!

8.2.4 Real Axis – Dialogs

Overview on Dialogs

When opening the folder of a real axis in online mode, the individual dialogs - display and setting options for parameters of the axis/IndraDrive - are provided.

First of all, it is assumed that the presettings cover a wide range of applications and that a modification is only required in exceptional cases.

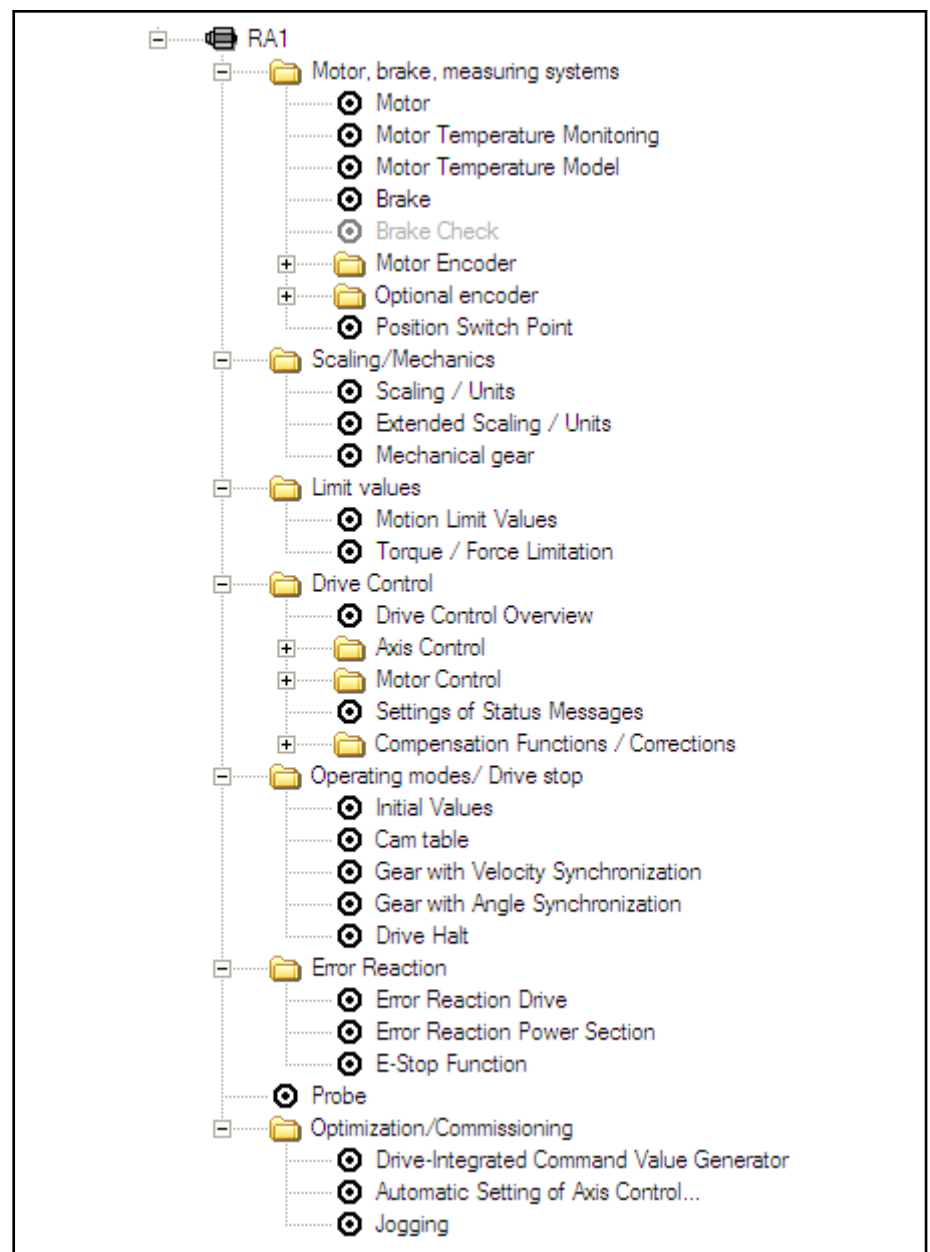


Fig. 8-12: Dialogs of a real axis (IndraDrive)



For the SercosDrive device, the drives are controlled via the PackProfile.

As the IndraDrive and EcoDrive CS devices are drive devices specified in detail - in contrast to the SercosDrive - device-specific dialogs are provided for commissioning.

Axes

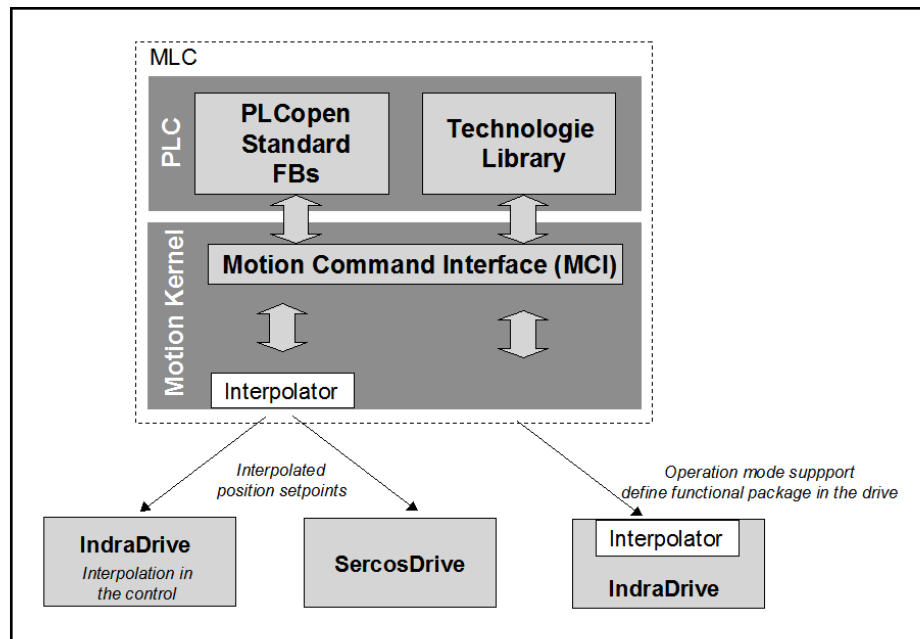


Fig. 8-13: Control principle for different drives



For a real axis without position feedback (open loop), only the operation modes without position feedback are shown.

- [Motor](#), page 272
- [Motor Temperature Monitoring](#), page 273
- [Brake](#), page 273
- [Measuring Systems](#), page 274
- [Settings: Scaling/Units](#), page 277
- [Scaling/Units Extended](#), page 278
- [Mechanical Gear](#), page 280
- [Closed-Loop Control](#), page 281 (reference to the drive documentation)
- [Settings of Status Messages](#), page 281
- [Drive Halt](#), page 283
- [Establishing Dimensional Reference](#), page 283
- [Error Reaction](#), page 285
- [E-Stop Function](#), page 288
- [Motion Limit Values](#), page 289
- [Initial Values](#), page 291
- [Compensation Functions/Corrections](#), page 293 (reference to the drive documentation)
- [Drive-integrated Command Value Generator](#), page 294 (reference to the drive documentation)
- [Cam Table](#), page 295
- [MotionProfile](#), page 300
- [FlexProfile](#), page 307
- [Gear with Velocity Synchronization](#), page 314
- [Gear with Phase Synchronization](#), page 317

- [Drive-integrated Safety Engineering](#), page 323
- [Position Switching Point](#), page 323
- [Touch Probe](#), page 324
- [Jogging](#), page 327

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is specified.



Dialogs exclusively referring to internal drive data are only described by a reference to the normally product-dependent original manuals.

8.2.5 Real Axis – Context Menu

Overview on Menu Items

The context menu of a real axis contains the following menu items:

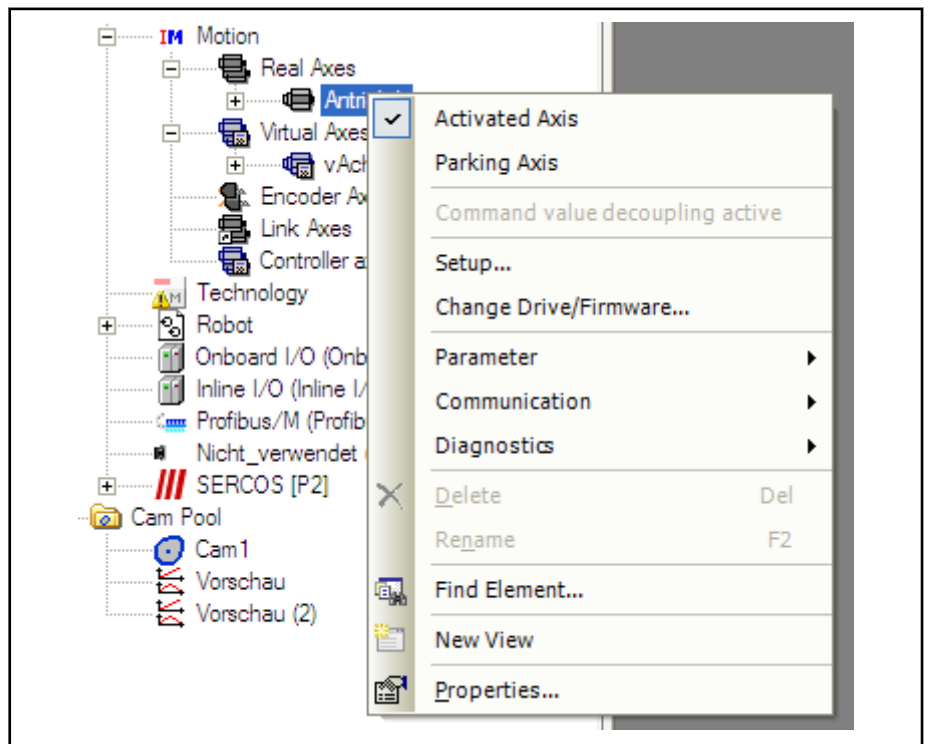


Fig. 8-14: Context menu of a real axis (IndraDrive) in online mode

The context menu of a real SercosDrive axis corresponds to the context menu of a real IndraDrive axis, except that it does not contain the menu item **Parking Axis**.

- Possibility to switch between the drive's operating states **Activated Axis/Deactivated Axis** and **Parking Axis** using the possibilities of the IndraDrive drive (see also [Modes to Operate a Real Axis](#), page 359).
- **Command value decoupling enabled** - see [Modes to Operate a Real Axis](#), page 359
- [Initial Commissioning](#), page 328 - Commissioning cycle of the real axis (IndraDrive)
- [Exchanging the Drive/Firmware](#), page 329
- [Parameters](#), page 334

Axes

- [Communication](#), page 336
- [Parameter Editor](#), page 133
- [Parameter Group](#), page 135
- [Diagnostics](#), page 341
- **Delete** - The complete axis can be deleted from the project in **Offline mode** via this menu item
- **Rename** - The axis name can be changed in **offline mode** via this menu item
- **Find element...** - Allows searching for elements in the project
- **New View** - Displays the axis in an individual window
- **Properties...** - The dialog shows the properties of the control entered during its configuration (see [Creating a Real Axis](#), page 234) There is also in offline state the option in this dialog to configure the drive or change a configuration already executed.

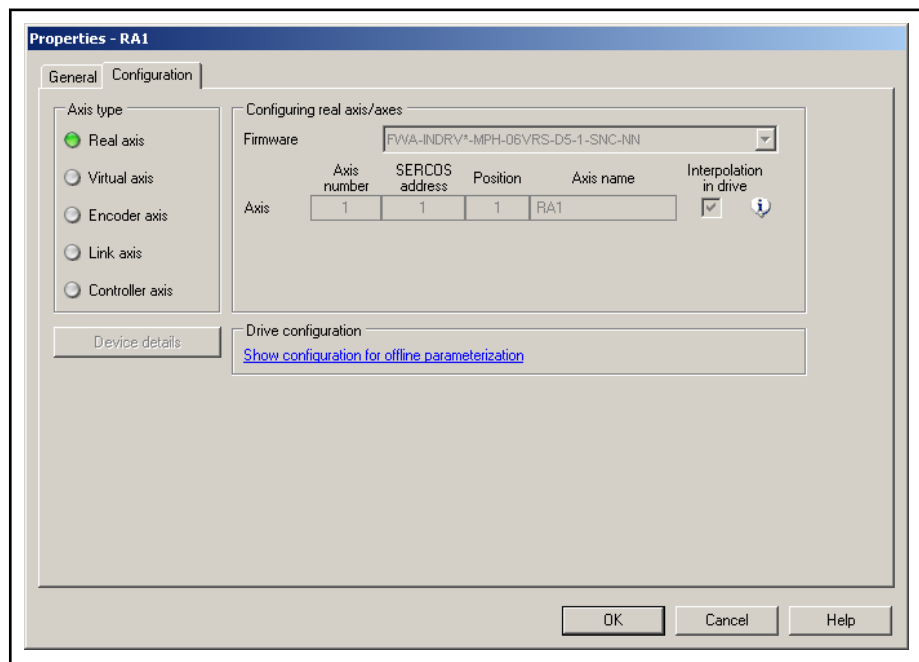


Fig.8-15: Opening the dialog to change the location for interpolation (control/drive) in offline mode



The axis data transferred to the PLC also changes with the axis name. The PLC program is also this way which may lead to compilation errors.

Deactivating an axis - Real axis (IndraDrive)



Deactivating a real axis, its drive, its electronics, results in the deactivation of the encoder axis whose encoder is connected to this electronics.

8.3 Virtual axis

8.3.1 General Information

The virtual axis is among the core functions of the IndraMotion MLC. It provides the master value for the synchronization of drives or other virtual axes.

The master value is effective for all slave axes at the same time. The master-slave relationship can be established using the following function blocks:

- Gear functions
MC_GearIn, MB_GearInPos
- ML_FlexProfile
- MB_MotionProfile

Virtual axes can be commanded with function blocks. They can also follow other axes. In this case, the calculation sequence for the individual axes is always arranged such that master axes are always calculated before slave axes. This prevents dead times. Recursive master-slave relationship (e.g. axis 1 follows axis 2 and axis 2 follows axis 1) are identified and rejected.

8.3.2 Creating a Virtual Axis (Virtual Master Axis)

Creating the virtual axis

With a virtual master axis, the master axis position is generated according to the preset velocity command value. Thus, the following restrictions are taken into account:

- A-0-0032, Velocity limit value, positive
- A-0-0033, Velocity limit value, negative
- A-0-0034, Acceleration limit value, bipolar
- A-0-0036, Jerk limit value, bipolar
- A-0-0224, Rapid stop deceleration



Axes can only be created in offline mode!

A virtual axis is created in the Project Explorer, in the **Motion -> Virtual Axes** folder of the control. For this purpose, the context menu has to be enabled (right mouse button or <Shift>+<F10>).

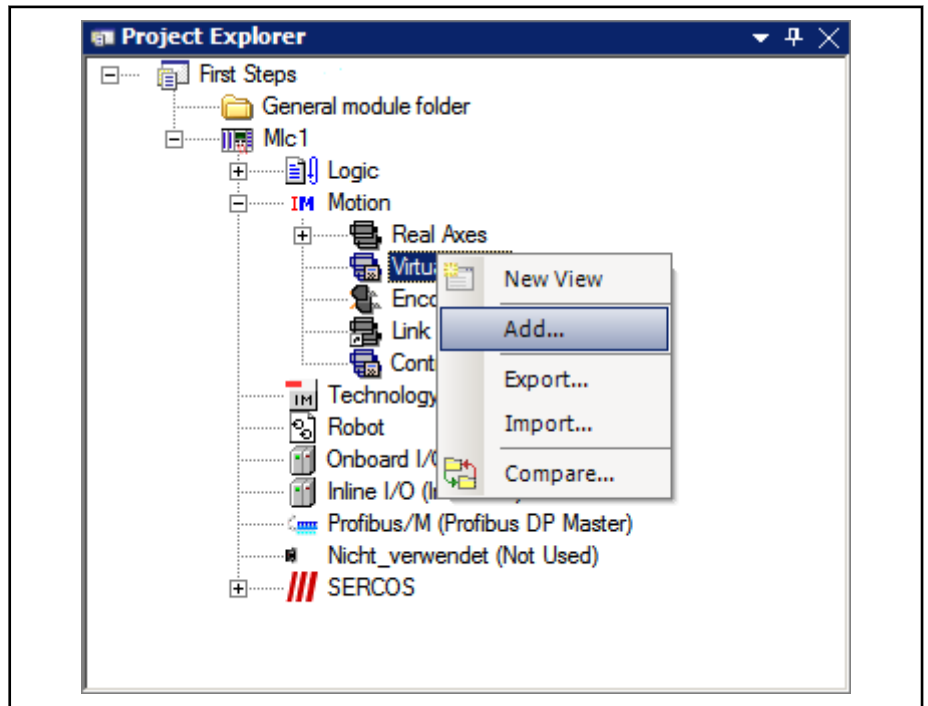
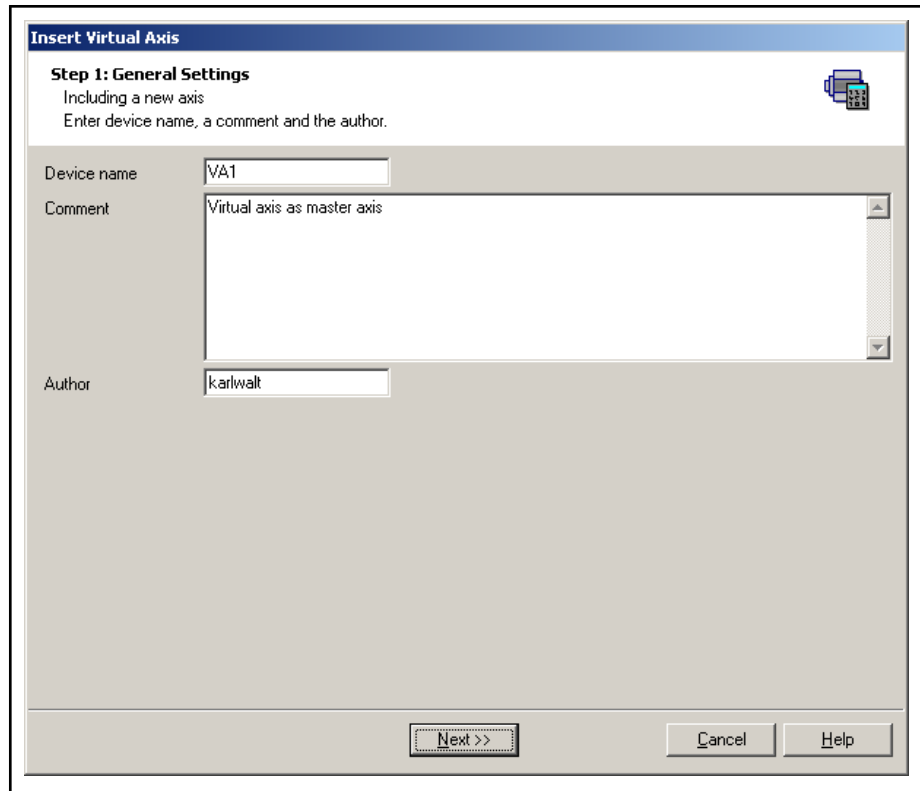


Fig. 8-16: Creating a virtual axis – Context menu

Dialogs to create a virtual axis

The creation of a virtual axis is supported by a wizard with two dialog boxes.

Axes



Insert Virtual Axis

Step 1: General Settings
Including a new axis
Enter device name, a comment and the author.

Device name: VA1

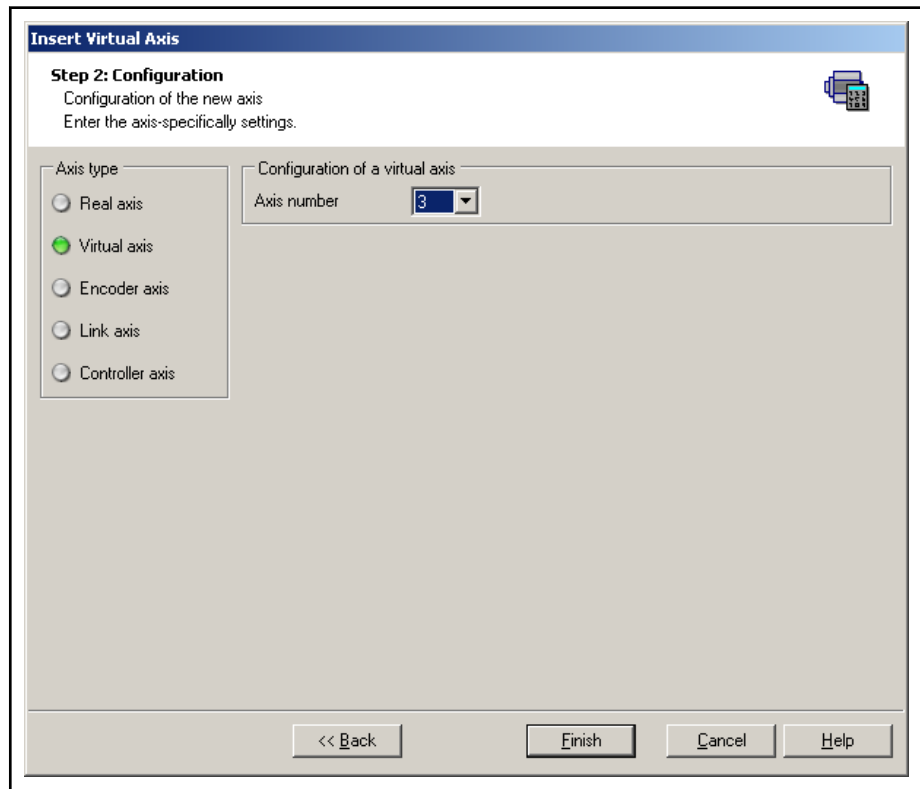
Comment: Virtual axis as master axis

Author: karl.walt

Next >> Cancel Help

Fig. 8-17: Creating a virtual axis – General Settings

In the "General Settings" step, the device name, a comment on this device and the author are entered. Apply or change the preset values.



Insert Virtual Axis

Step 2: Configuration
Configuration of the new axis
Enter the axis-specific settings.

Axis type:

- ☐ Real axis
- ☒ Virtual axis
- ☐ Encoder axis
- ☐ Link axis
- ☐ Controller axis

Configuration of a virtual axis:

Axis number: 3

<< Back Finish Cancel Help

Fig. 8-18: Creating a virtual axis – Configuration

The axis number can be set in the "Configuration" step.

Axis assignment To address the axis, the PLC uses the axis number and axis name (here "3" or "VA1").

The virtual axis is shown only once in the Project Explorer:

Context menu If the context menu is opened on the folder **Motion -> Virtual Axes -> VA1** by right-clicking on it, the following menu items appear:

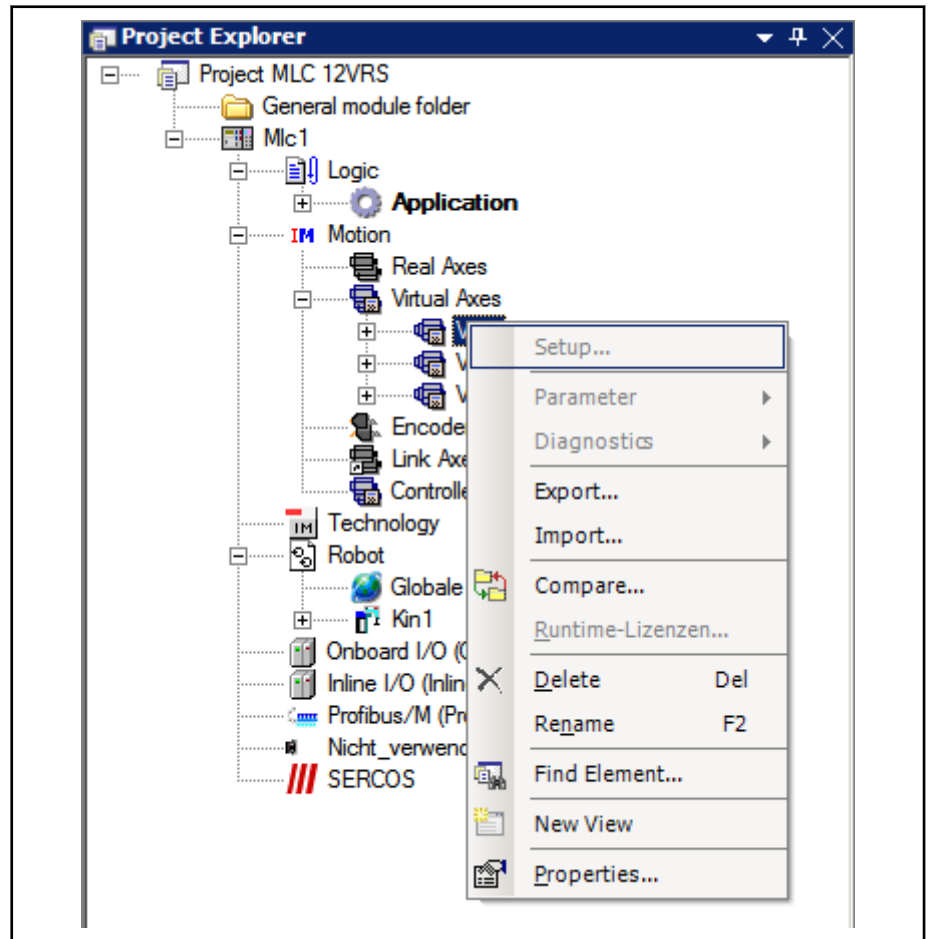


Fig. 8-19: Virtual axis - Context menu

The following can be selected:

- **Export...**
- **Import...**
- **Compare...**
- **Delete**
- **Rename**
- **New View**
- **Properties...**

Since, at this point in time, no connection with the control exists yet (offline mode), a number of commands is disabled, i.e. grayed out (for the complete description, see [Virtual axis - Context menu, page 246](#), or [Virtual Axis - Dialogs, page 246](#)).

Axes

8.3.3 Virtual Axis - Dialogs

Overview on Dialogs

When opening the folder of the virtual axis, the individual dialogs of the axis - display of data from the axis and setting options for parameters - are provided.

First of all, it is assumed that the presets cover a wide range of applications and that a modification is only required in exceptional cases.

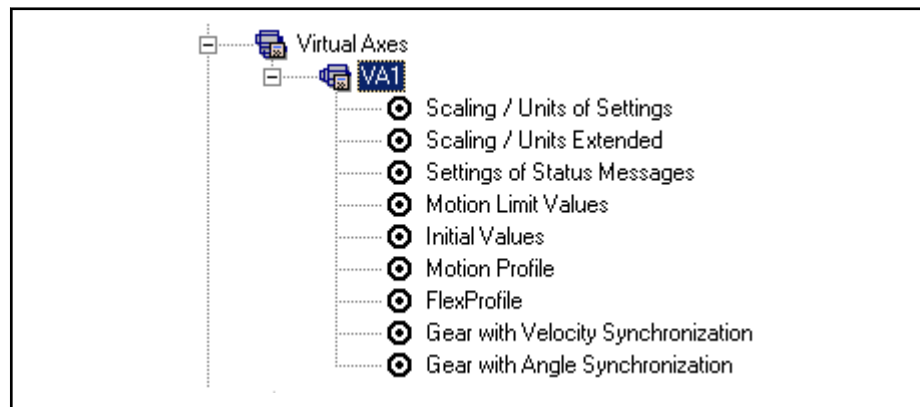


Fig. 8-20: Dialogs of a virtual axis

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is specified.

The following dialogs are available for virtual axes:

- [Settings: Scaling/Units, page 277](#)
- [Scaling/Units Extended, page 278](#)
- [Settings of Status Messages, page 281](#)
- [Motion Limit Values, page 289](#)
- [Initial Values, page 291](#)
- [MotionProfile, page 300](#)
- [FlexProfile, page 307](#)
- [Gear with Velocity Synchronization, page 314](#)
- [Gear with Phase Synchronization, page 317](#)
- [Jogging, page 327](#)

8.3.4 Virtual axis - Context menu

Overview on Menu Items

The context menu of a virtual axis in online mode contains the following menu items:

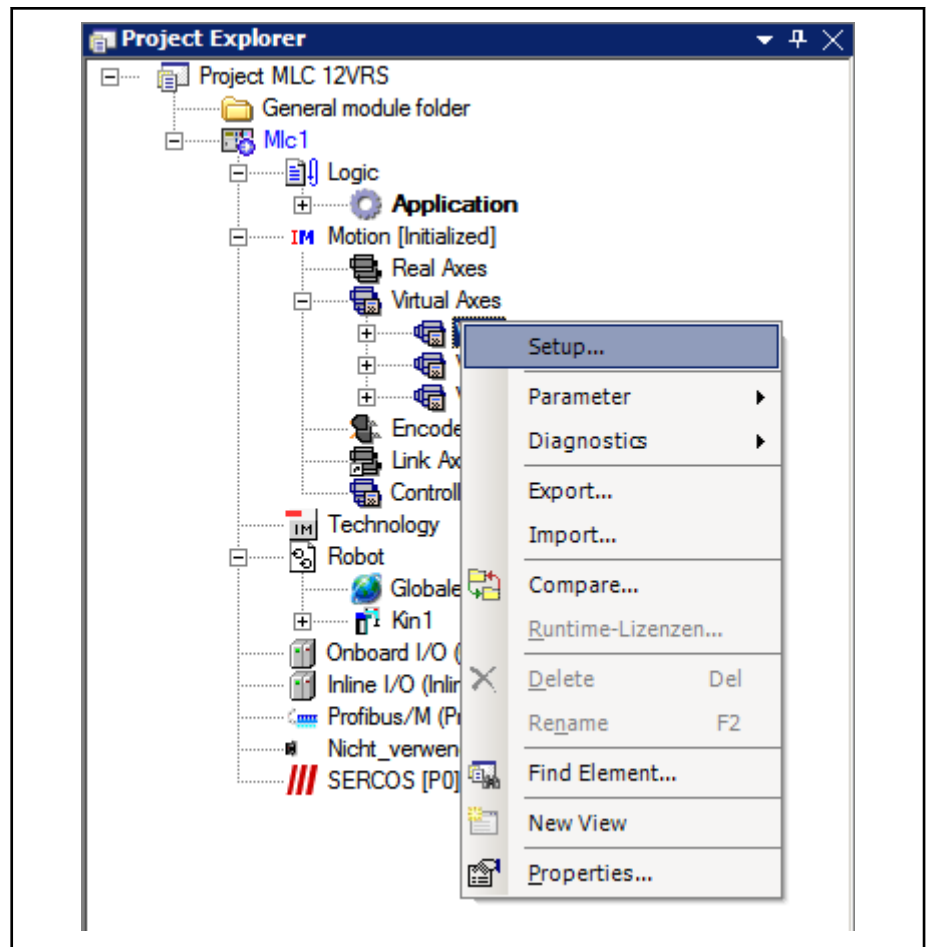


Fig. 8-21: Context menu of a virtual axis in online mode

- [Initial Commissioning](#), page 328
- [Parameters](#), page 334
- [Parameter Editor](#), page 133
- [Parameter Group](#), page 135
- [Status](#), page 341
- [Clearing Errors](#), page 342
- **Export...** - Saves axis elements in a file. This menu item is described in [Exporting](#), page 172
- **Import...** - Imports axis elements from a file This menu item is described in [Importing](#), page 174
- **Compare...** - Compares and merges objects, see [Comparing/Merging](#), page 176
- **Runtime licenses...** - See [Runtime Licenses](#), page 178
- **Delete** - The complete axis can be deleted from the project in **Offline mode** via this menu item
- **Rename** - The axis name can be changed in **offline mode** via this menu item



The axis data transferred to the PLC also changes with the axis name. The PLC program is also this way which may lead to compilation errors.

Axes

- **Find element...** - Allows searching for elements in the project
- **New View** - Displays the axis in an individual window
- **Properties...** - the dialog the properties of the control which were entered at the point in time of its configuration (see [Creating a Virtual Axis \(Virtual Master Axis\)](#), page 243).

8.4 Controller axis

8.4.1 General Information

Controller axes are used to implement the controller of an axis in the PLC and to command this axis as all other axes. All Motion function blocks are available for this axis type. A controller axis can be:

- moved directly (MC_MoveAbsolute)
- used as a master and slave axis for a synchronized motion
- used in a kinematics

User-defined controllers can be implemented in the PLC. Moreover, the control of a drive is not defined. For example, any number of inputs can be used as encoder inputs and one motor can be controlled with any number of outputs. The interface to the controller axis is the position command value or the actual command value.

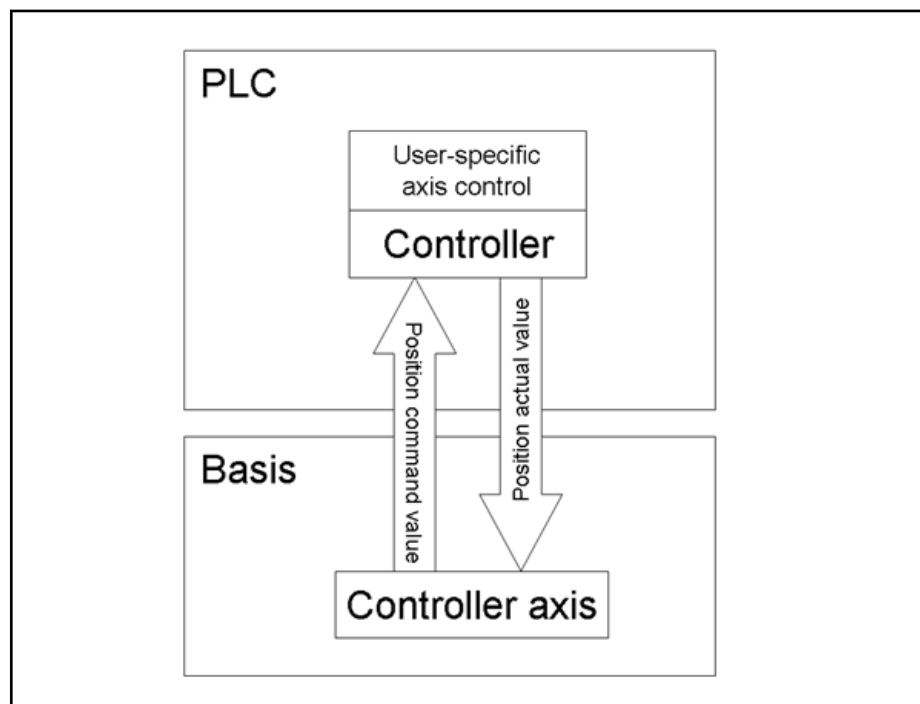


Fig.8-22: Communication between the controller axis and the controller in the PLC

8.4.2 Creating a Controller Axis

Creating an Axis in the Project Explorer

Creating controller axis A controller axis is created in the Project Explorer, in the **Motion -> Control Axes** folder of the control. For this purpose, the context menu has to be enabled (right mouse button or <Shift>+<F10>).

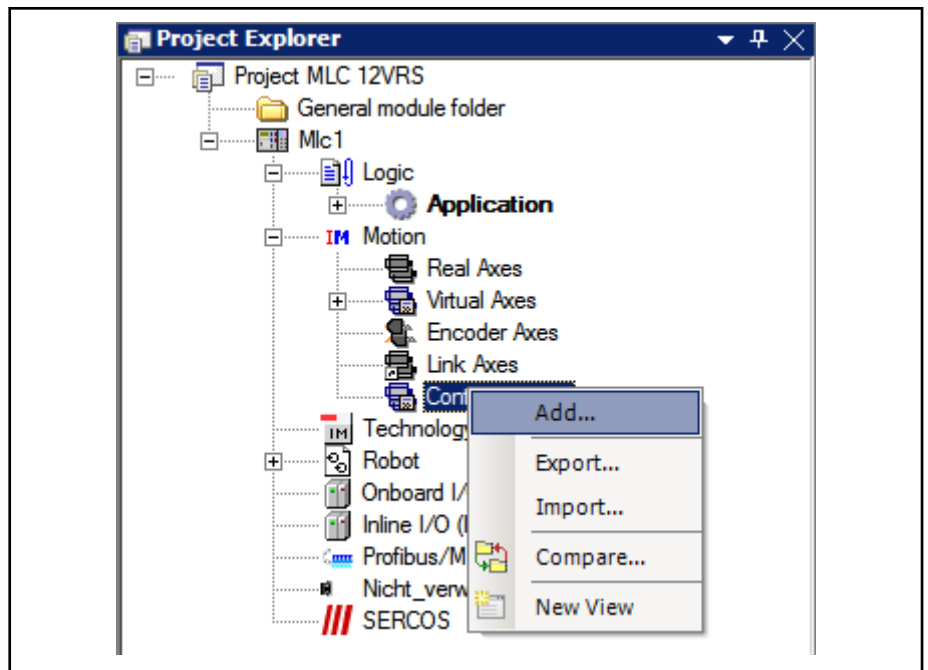


Fig.8-23: Creating a controller axis - Context menu



Axes can only be created in offline mode!

Step 1: General Settings

Dialogs to create a controller axis

The creation of a controller axis is supported by a wizard with multiple dialog boxes.

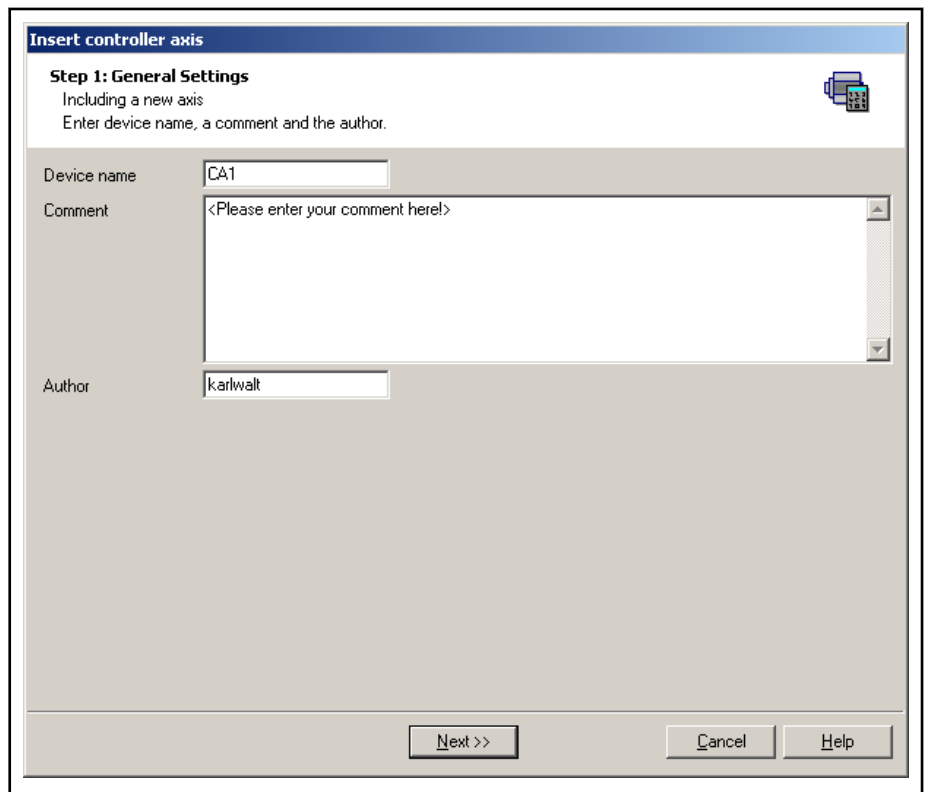


Fig.8-24: Creating a controller axis – General settings

Axes

In the "General Settings" step, the device name, a comment on this device and the author are entered. Apply or change the preset values.

Step 2: Configuration

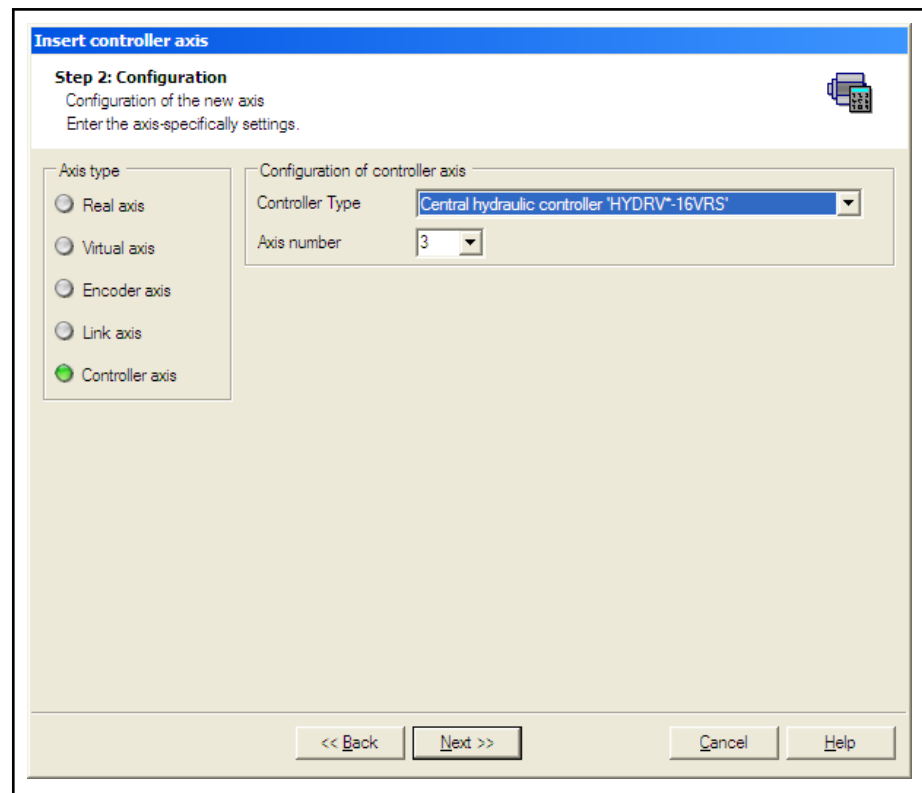


Fig.8-25: Creating a controller axis – Configuration

The axis number can be set in the "Configuration" step.

Axis number The next free axis number is assigned to the axis (here "3", since a real axis and a virtual axis already exist in the project).

To address the axis, the PLC uses the axis number .

Controller type The following controller types can be selected:

- Freely-programmable controller
- Hydraulic central controller
- Hydraulic synchronization controller

Additionally, a fitting version has to be selected for hydraulic controllers.

The following chapter describes the creation of different controller types.

Central Controller Step 3: Hardware Link

This chapter describes the creation of a central controller. It is assumed that the steps from [chapter "Creating an Axis in the Project Explorer" on page 248](#) were already performed and select **Central hydraulic controller** as controller type.

The connected hardware is selected in the "Hardware link" step.

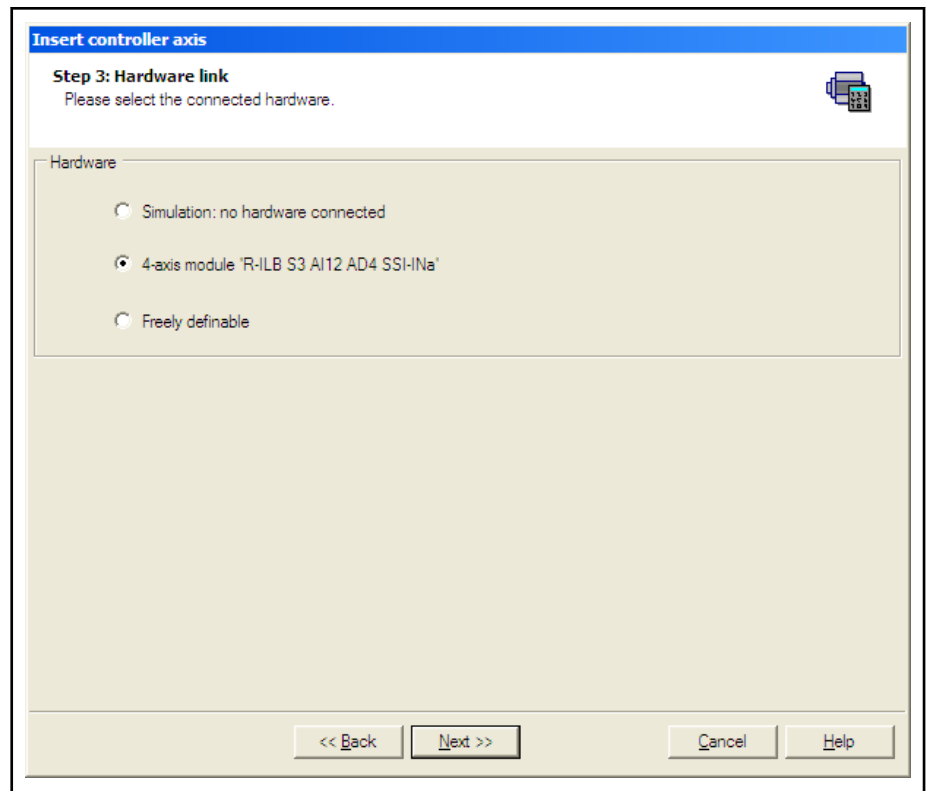


Fig.8-26: Creating a central hydraulic controller axis – Signal assignment (IP20 4-axis module)

Central Controller Step 4: Signal Assignment

The signals for the IP20 4-axis module can be assigned here. This is performed via the **Select Axis...** button (quick selected the required axis signals) or via the ... for an individual signal. If an individual signal is not required anymore, it can be deleted via the **X** button.

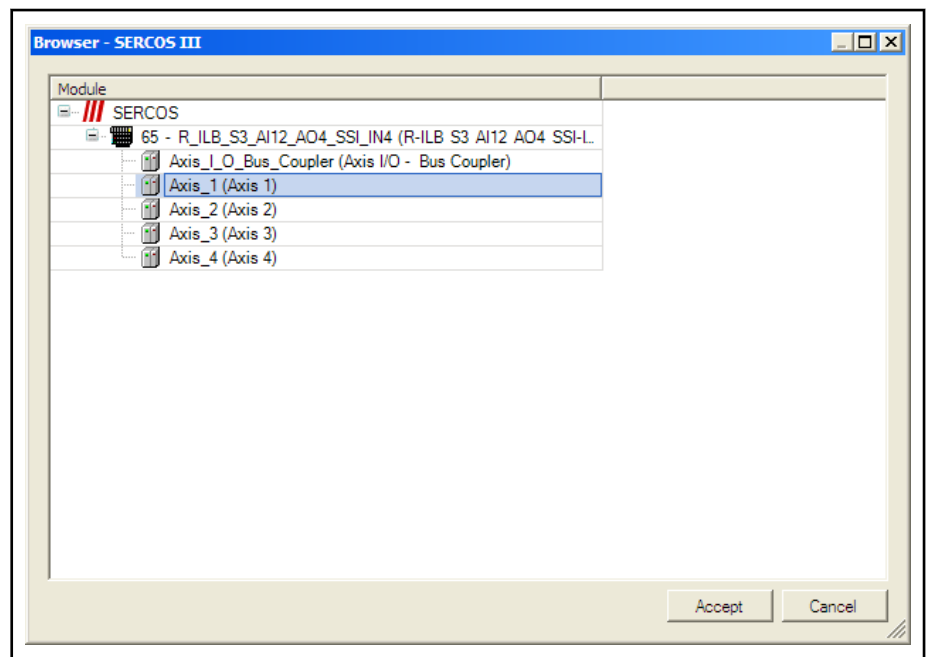


Fig.8-27: Creating a central hydraulic controller axis – Browser - sercos III - Axis selection

Axes

Insert controller axis

Step 4: Signal assignment
Please select an input/output assignment.

Default assignment
Module type: R-ILB S3 AI12 AD4 SSI-IN4 Select Axis...

Input/output assignment

Encoder 1:		!	X	...
Cmd. valve value:			X	...
Act. valve value:			X	...
Pressure A:			X	...
Pressure B:			X	...
System pressure:			X	...
Tank pressure:			X	...
Act. force value:			X	...
Encoder 2:			X	...

The selected signals have to comply with the R-ILB standard (e.g., they can be derived from the original hardware name).

<< Back Next >> Cancel Help

Fig. 8-28: Creating a central hydraulic controller axis – Browser - sercos III - Selection of the individual signals

The signals are assigned using sercos III browsers. The respective module has to be selected via mouse-click. Depending on the selection, this mask displays either all connected sercos III modules or only connected sercos III IP20 4-axis modules.

Last Step: Summary

The summary shows all parameters set. The axis in the Motion node is only created after the summary has been confirmed. If it is the first inserted controller axis, one task and one program are additionally inserted to call the controller axis.

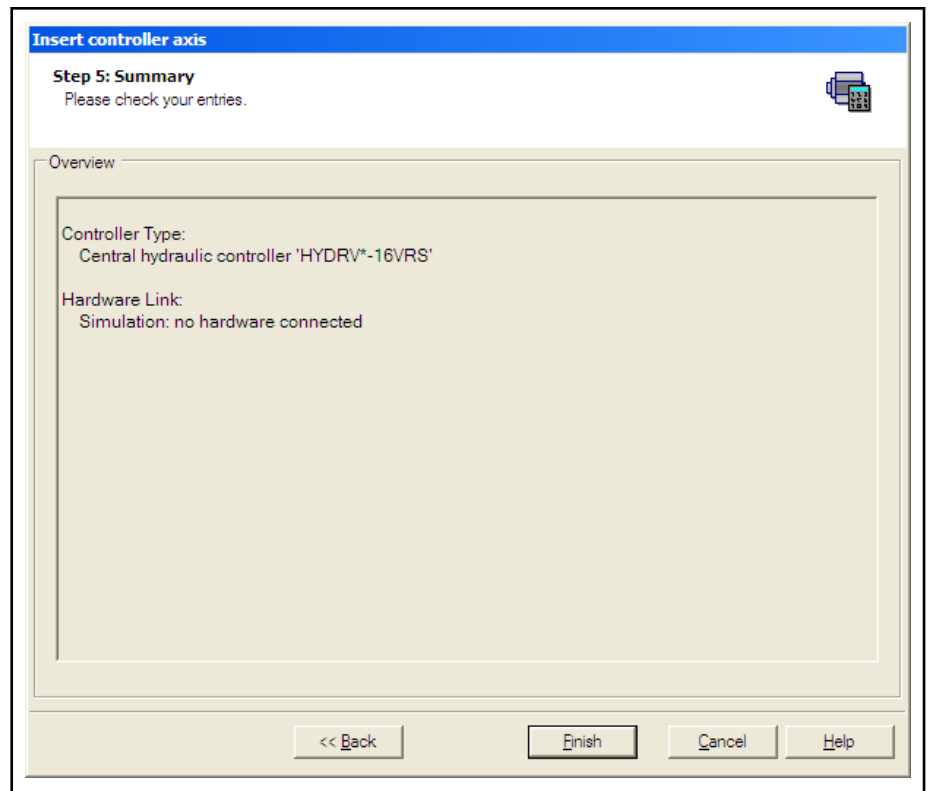


Fig. 8-29: Creating a controller axis – Summary

8.4.3 Controller Axis - Dialogs

Controller Axis - Overview on Dialogs

When opening the folder of a controller axis in online mode, the individual dialogs - display and setting options for parameters of the axis - are provided.

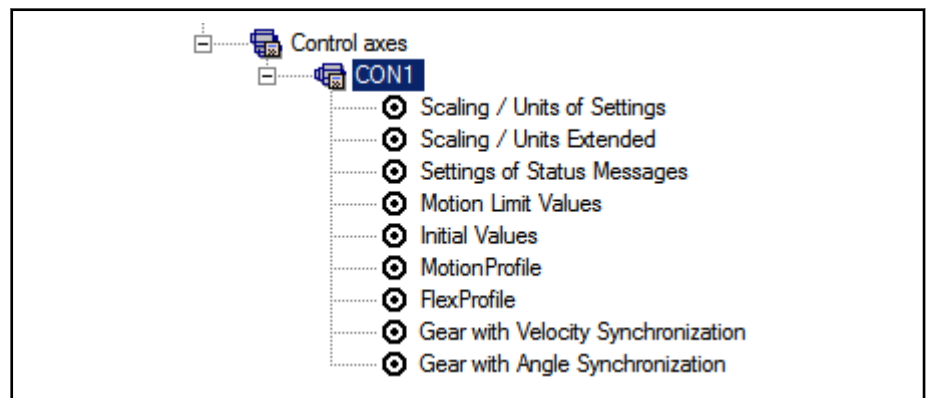


Fig. 8-30: Dialogs of a controller axis

Some of the controller axis dialogs are only slightly different from the dialogs of a real axis. The descriptions of the corresponding dialogs for a real axis are located in

- [Settings: Scaling/Units, page 277](#)
- [Scaling/Units Extended, page 278](#)
- [Settings of Status Messages, page 281](#)
- [Motion Limit Values, page 289](#)
- [Initial Values, page 291](#)

Axes

- [MotionProfile](#), page 300
- [FlexProfile](#), page 307
- [Gear with Velocity Synchronization](#), page 314
- [Gear with Phase Synchronization](#), page 317

8.4.4 Controller Axis - Context Menu

Overview on Menu Items

The context menu of a controller axis comprises the following menu items:

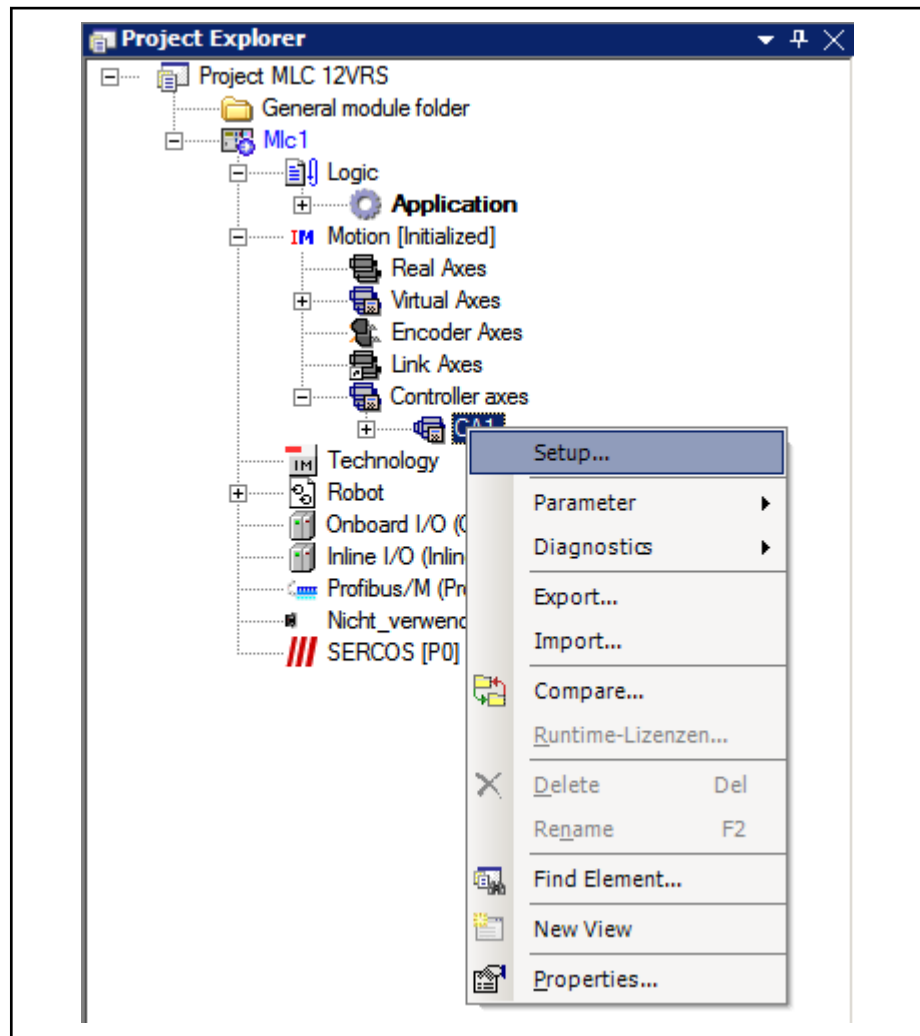


Fig.8-31: Context menu of a controller axis in online mode

- **Setup...** - See [chapter "Initial Commissioning"](#) on page 255
- **Parameter** - see
 - [Parameter Editor](#), page 133
 - [Parameter Group](#), page 135
 - [Parameters](#), page 334
- **Diagnostics** - See [chapter "Diagnostics"](#) on page 255
- **Export...** - Saves axis elements in a file. This menu item is described in [Exporting](#), page 172

Axes

- **Import...** - Imports axis elements from a file This menu item is described in [Importing, page 174](#)
- **Compare...** - Compares and merges objects, see [Comparing/Merging, page 176](#)
- **Runtime licenses...** - See [Runtime Licenses, page 178](#)
- **Delete** - The complete axis can be deleted from the project in **Offline mode** via this menu item
- **Rename** - The axis name can be changed in **offline mode** via this menu item



The axis data transferred to the PLC also changes with the axis name. The PLC program is also this way which may lead to compilation errors.

- **New View** - Displays the axis in an individual window
- **Properties...** - the dialog shows the properties of the control entered during its configuration (see [Creating a Controller Axis, page 248](#)).

Initial Commissioning

Since only some of the controller axis dialogs differ only slightly from the dialogs of a real axis, the following refers to the dialogs of the real axes.

For the initial commissioning of a controller axis, the following dialogs are processed by a wizard:

- Load basic parameters. The parameters are set to factory default values. This procedure is described in [Loading Basic Parameters, page 334](#)
- [Settings: Scaling/Units, page 277](#)
- [Motion Limit Values, page 289](#)
- [Initial Values, page 291](#)
- [Settings of Status Messages, page 281](#)

Diagnostics

Status

With the dialog, the status information on the respective axis is graphically prepared and displayed. The window is split in two, the upper half containing general information, whereas the lower half contains detailed supplements.

To identify the axis:

- A-0-0001, Logical axis number
- A-0-0002, Axis name
- A-0-0005, Drive address
- A-0-0004, Axis type
 - Bit 3 - Real axis, IndraDrive or HydraulicDrive device
 - Bit 4 - Virtual axis
 - Bit 5 - Encoder axis
 - Bit 6 - Link axis (only IndraMotion MLC)
 - Bit 7 - Controller axis
- A-0-0020, Axis diagnostic message

Axes

- A-0-1030, Command: Clear all axis errors, deletes the axis errors in the drive and in the control

Current values of the axis:

- A-0-0100, Actual position value
- A-0-0102, Actual velocity value
- A-0-0103, Actual acceleration value
- A-0-0110, Actual torque/force value

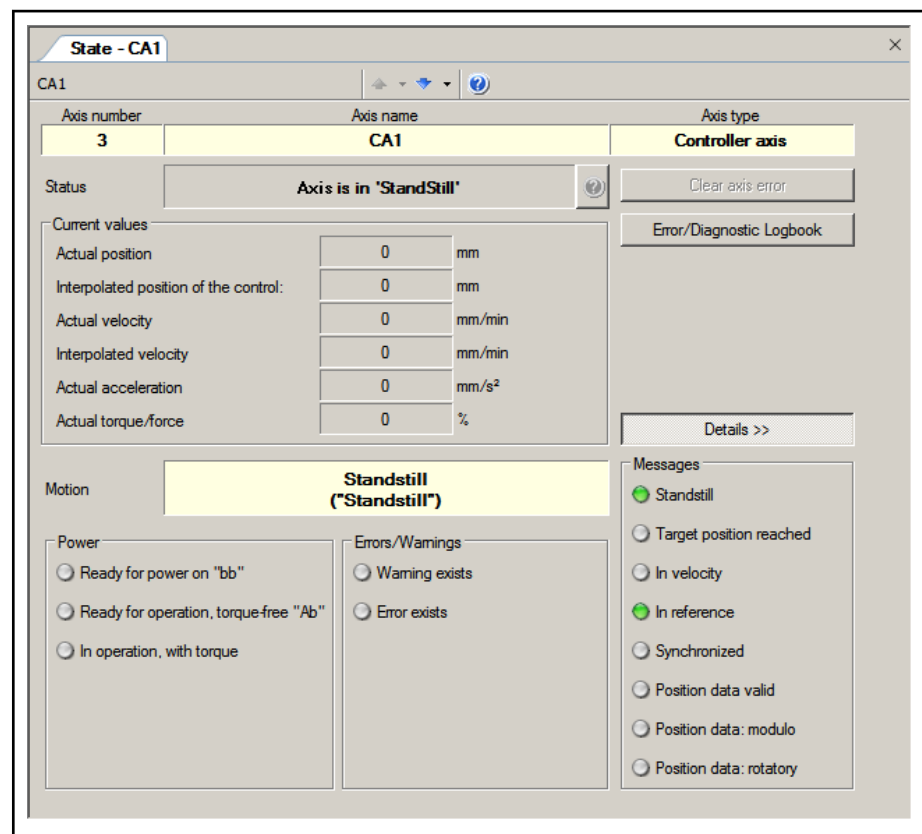


Fig. 8-32: Axis status of a controller axis

The detailed information is encoded as bit arrays in the parameters

- A-0-0021, Axis status and
- A-0-0022, Extended axis status



Also double click to the "Axis status" on the desired axis in the "Device status" of the control.

Clearing Errors

Selecting this context menu item triggers "A-0-1030, Command: Clear all axis errors" and deletes the axis errors in the drive and on the control.

8.5 Link axis

8.5.1 General Information

The link axis represents a sercos II axis provided on another control in the cross communication. Link axes can only be used as masters for other axes and cannot be directly commanded.

Example:

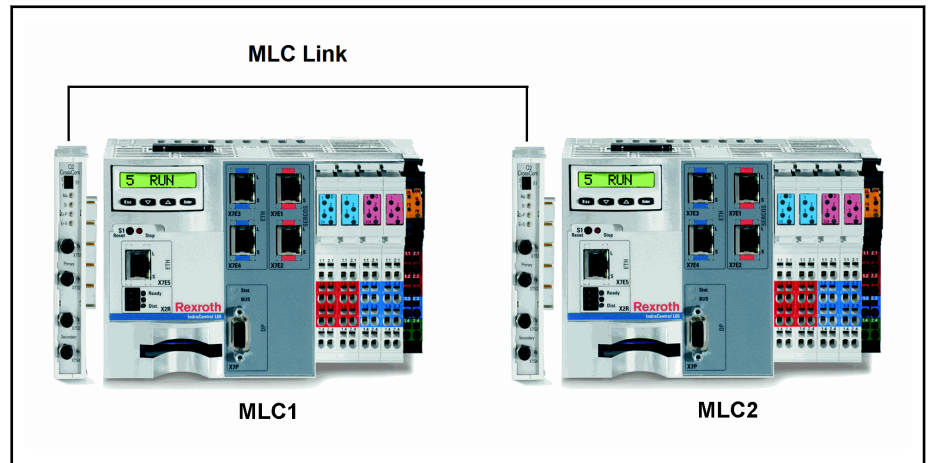


Fig. 8-33: Example: sercos II cross communication with two controls

MLC 1 has a virtual axis that is provided via the cross link. To use this axis in the MLC 2, a link axis has to be created on the MLC 2. This link axis can be used as a master axis to synchronize other MLC 2 axes.

Supported Motion function blocks:



The support of the PLCopen function blocks depends on the axis types.

For an overview, refer to [fig. 8-3 "Supported function blocks in an IndraMotion MLC" on page 229](#).

8.5.2 Creating a Link Axis

Settings on the Consumer Side

Local link axes To "consume" an axis provided as "link axis A" or "link axis B" as master for own local axes on the side of the producer, a representative has to be locally defined showing its own control.

Thus, a link axis is created locally in the IndraWorks node Motion ► Link Axis.

This axis receives axis number and axis name to be able to be addressed as well as the assignment to a master axis in the control link.

Axes

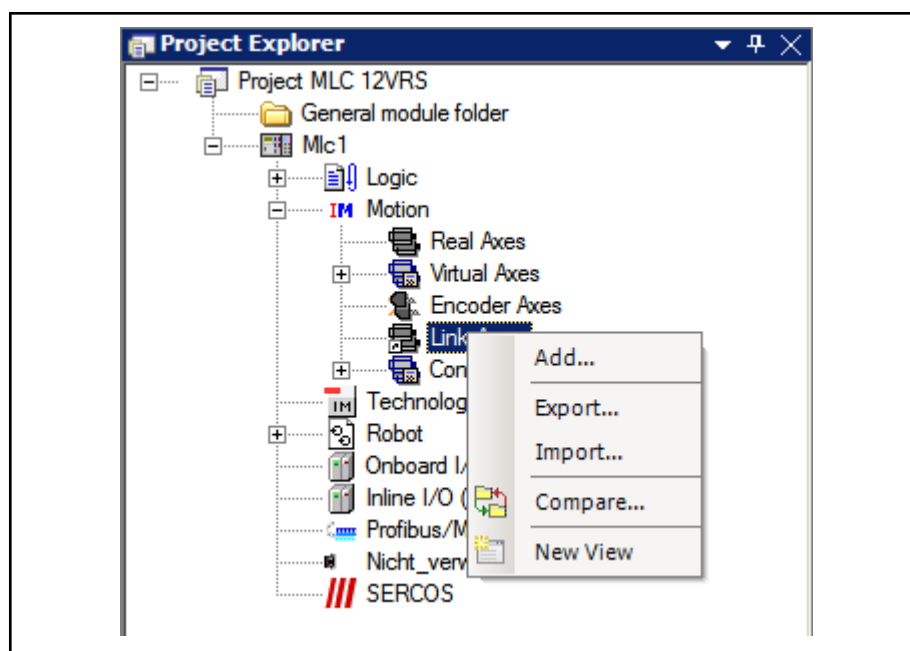


Fig.8-34: Adding a link axis to the project

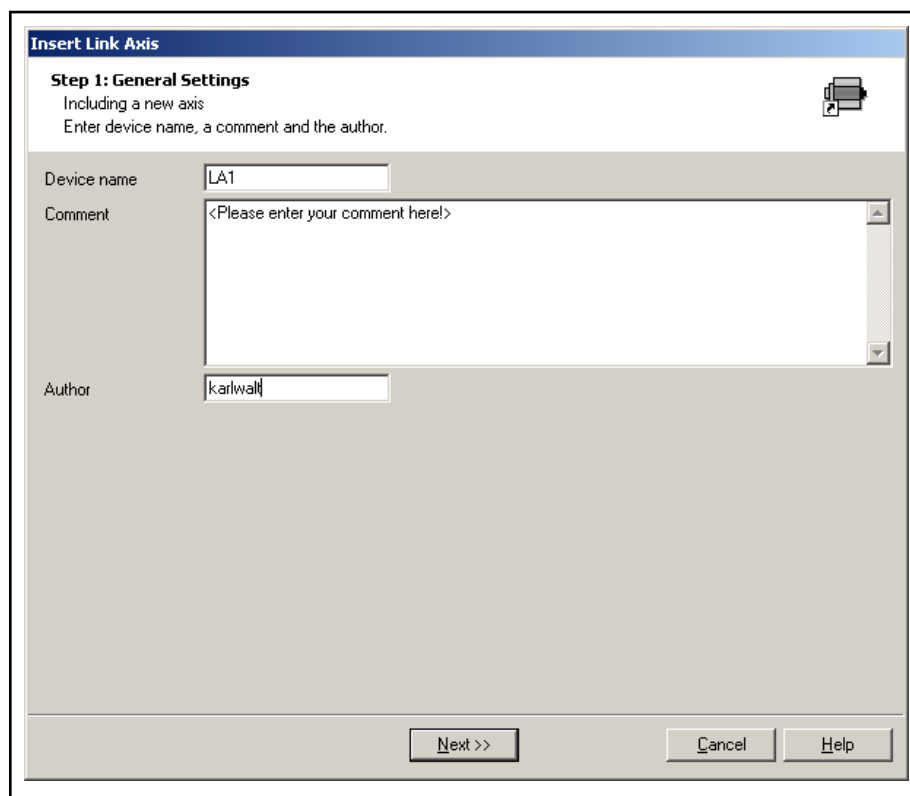


Fig.8-35: Selecting the axis of an external control as link axis ①

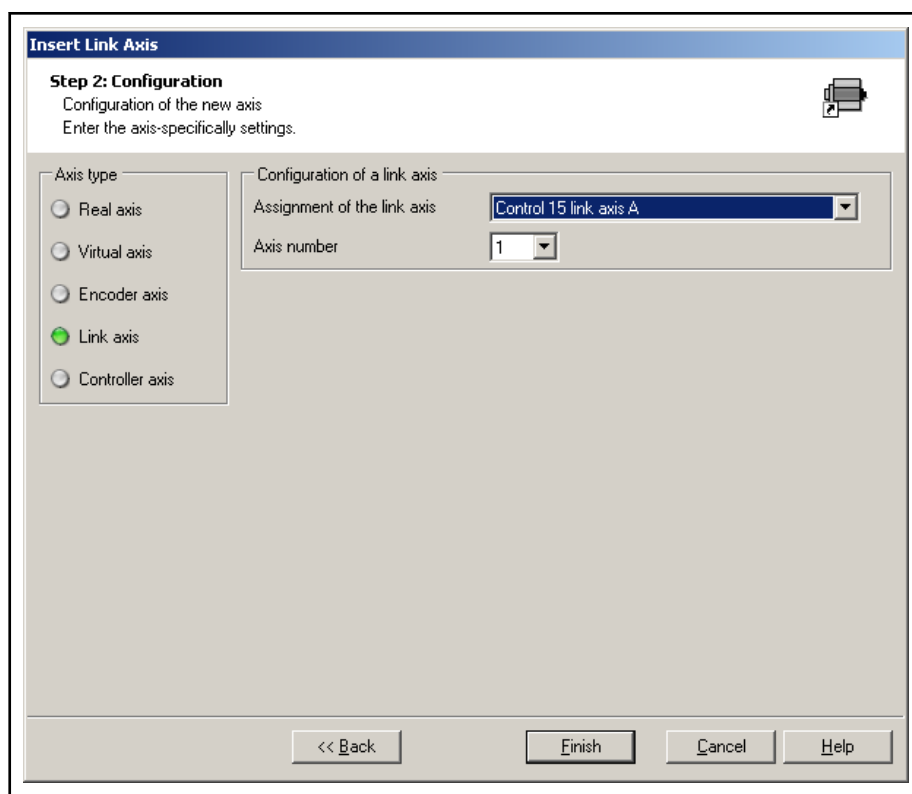


Fig. 8-36: Selecting the axis of an external control as link axis ②



All available link axes produced are always listed, since at configuration (offline), the link devices available or the link axes available cannot be determined.



Ensure that the selected axis is also actually provided by the respective control in the link.

The assignment of a local link axis to an (external) master axis in the link is executed via the list parameter "A-0-0008, Link axis – master axis selection". The control address and the number of the desired master axis of this control are entered in this parameter.

The following applies:

Entry in A-0-0008	Effective master axis position
List element 1: xxxx (control selection)	Master axis position A/B of the control with control address xxxx
List element 2: yyyy (link axis selection)	yyyy = 1: master axis position A yyyy = 2: master axis position B
Example: A-0-0008 = 15 1	MLC 15 master axis position A (MLC 15 link axis A)

Fig. 8-37: Master axis assignment

Axes



A change in the assignment on the consumer side, e.g. by the PLC can be executed via the parameter "A-0-0008" (see configuration example "Rotary printing with 2 outputs").

Like other local axes, local link axes have their own parameters and related dialogs.

The number of supported parameters is limited but corresponds approximately to that of an encoder axis (see parameter "A-0-0010, List of all A-parameters").

The creation of a link axis with the help of a CrossComm module is described in /24/, [Required and Supplementing Documentations, page 14](#).

Settings on the Producer Side

The settings to be made by the producer for the sercos link are described in /24/, [Required and Supplementing Documentations, page 14](#).

8.5.3 Link Axis - Dialogs

Overview on Dialogs

When opening the folder of the link axis, the individual dialogs of the axis – display of axis data and setting options for axis parameters – are provided.

First of all, it is assumed that the presets cover a wide range of applications and that a modification is only required in exceptional cases.

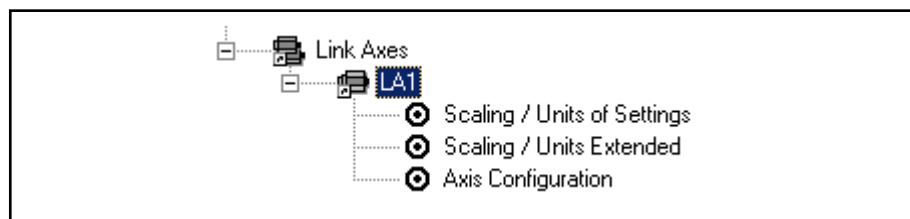


Fig.8-38: Dialogs of a link axis

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is specified.

The following dialogs are available for link axes:

- [Settings: Scaling/Units, page 277](#)
- [Scaling/Units Extended, page 278](#)
- [Axis Configuration, page 321](#)

8.5.4 Like Axis – Context Menu

Overview on Menu Items

The context menu of a link axis in online mode contains the following menu items:

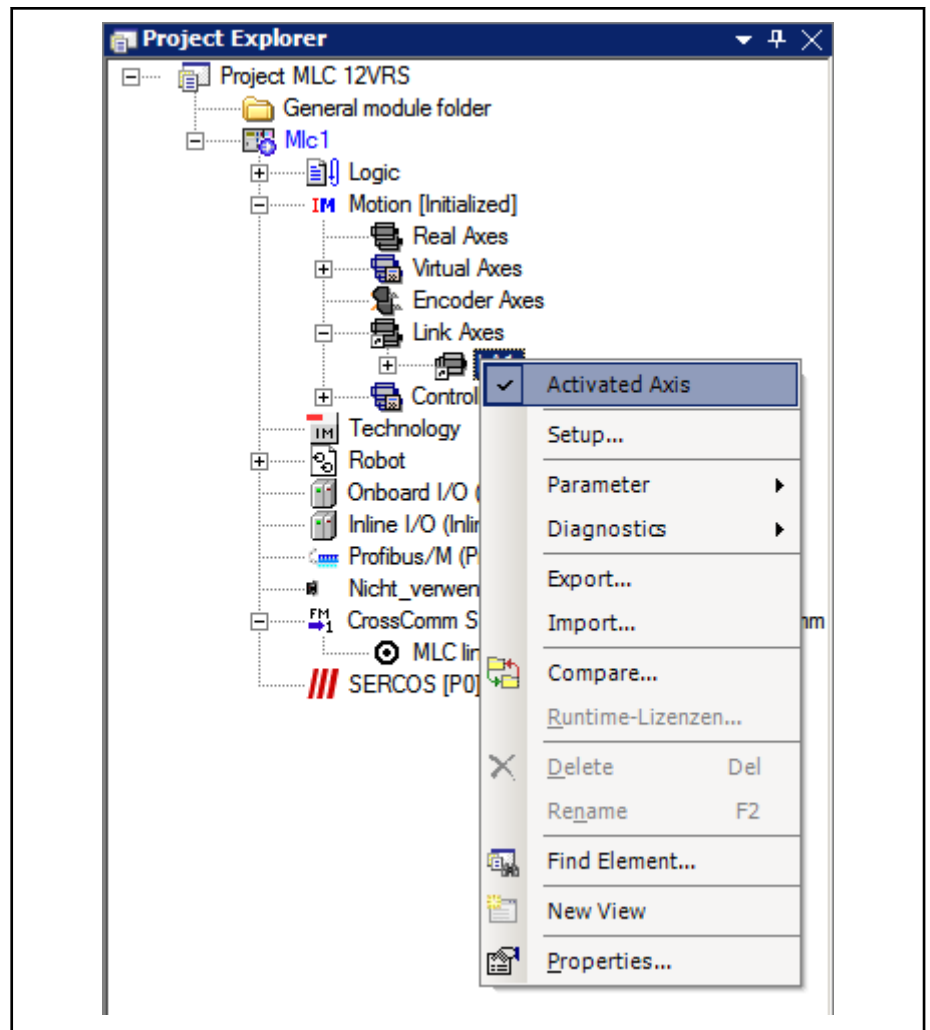


Fig. 8-39: Context menu of a link axis in online mode

- [Initial Commissioning, page 328](#) - Commissioning cycle of the link axis
- [Parameters, page 334](#)
- [Parameter Editor, page 133](#)
- [Parameter Group, page 135](#)
- [Status, page 341](#)
- [Clearing Errors, page 342](#)
- **Export...** - Saves axis elements in a file. This menu item is described in [Exporting, page 172](#)
- **Import...** - Imports axis elements from a file. This menu item is described in [Importing, page 174](#)
- **Compare...** - Compares and merges objects, see [Comparing/Merging, page 176](#)
- **Runtime licenses...** - See [Runtime Licenses, page 178](#)
- **Delete** - The complete axis can be deleted from the project in **Offline mode** via this menu item
- **Rename** - The axis name can be changed in **offline mode** via this menu item

Axes



The axis data transferred to the PLC also changes with the axis name. The PLC program is also this way which may lead to compilation errors.

- **New View** - Displays the axis in an individual window
- **Properties...** - the dialog shows the properties of the function module entered during configuration (e.g. axis number, assignment of the link axis)

8.6 Encoder axis

8.6.1 General Information

Encoder axes are used to evaluate an additional encoder of a real, existing drive and to provide the position information as axis in the system. An encoder axis can be used as master axis for synchronized motions and cannot be commanded.

8.6.2 Creating an Encoder Axis (Real Master Axis)

Creating an Axis in the Project Explorer

Creating the real axis to connect the encoder of the encoder axis

For a real master axis, the master axis position is derived from the signals of a master axis encoder. The encoder uses the drive control electronics of a predefined real axis, e.g. RA2, for the provision of the data required, i.e. before defining an encoder axis, the real axis RA2 has to be created first (see also [Creating a Real Axis, page 234](#)) and then the procedure described below is to be followed.

In the example, the axis RA2 has the drive address 2 and the axis number 2.



Without previous measuring encoder configuration (see [chapter "Configuration of the Measuring Encoder" on page 265](#)) with regard to the respective real axis, operating the encoder axis is not possible.



Axes can only be created in offline mode!

An encoder axis is created in the Project Explorer, in the **Motion -> Encoder Axes** folder of the control. For this purpose, the context menu has to be enabled (right mouse button or <Shift>+<F10>).

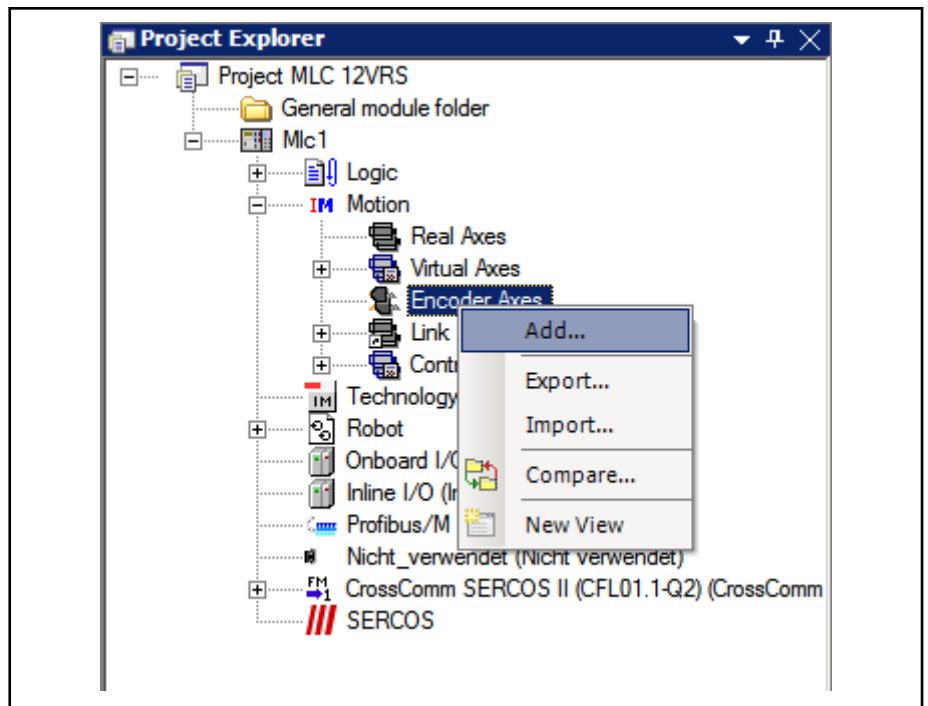


Fig. 8-40: Creating an encoder axis - Context menu

Dialogs to create an encoder axis

The creation of an encoder axis is supported by a wizard with two dialog boxes.

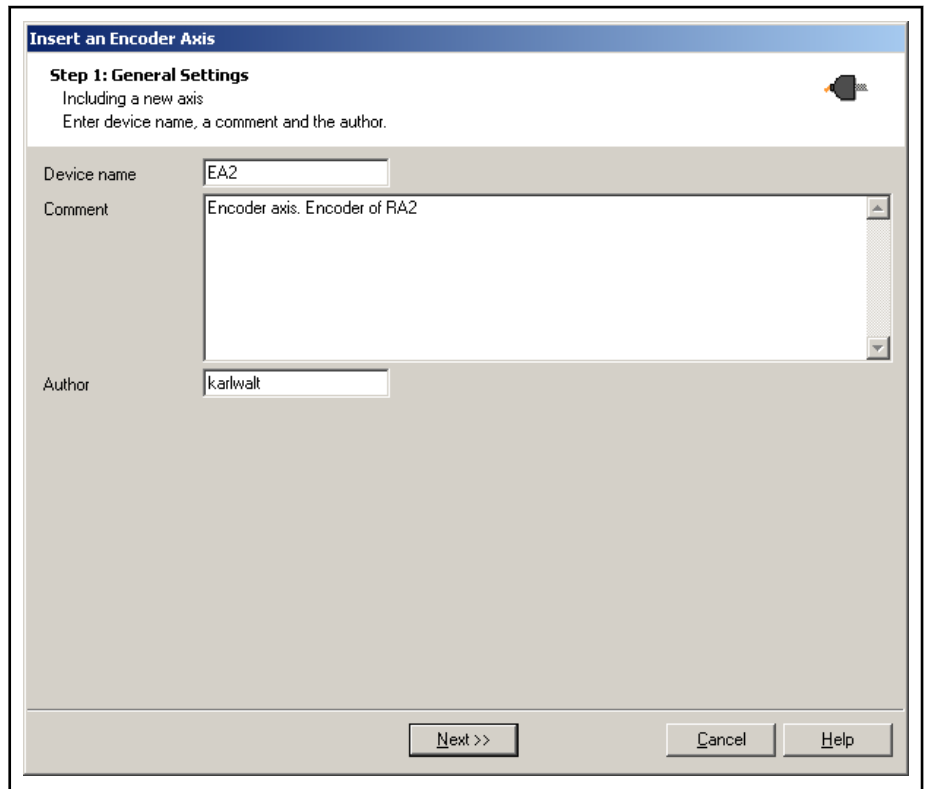


Fig. 8-41: Creating an encoder axis – General Settings

In the "General Settings" step, the device name, a comment on this device and the author are entered. Apply or change the preset values.

Axes

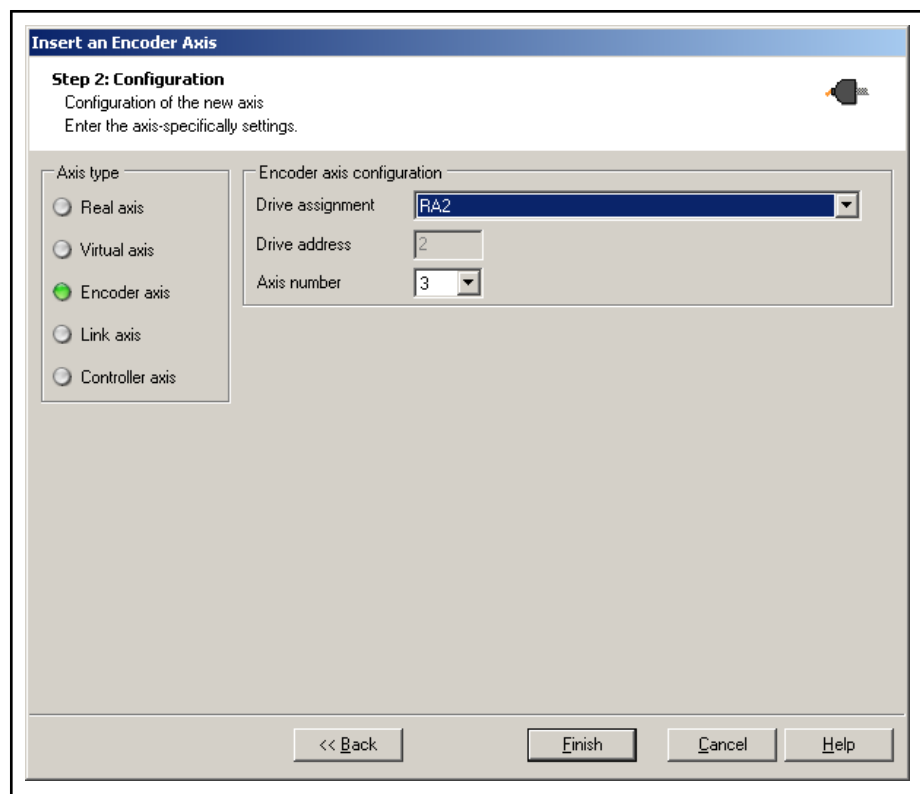


Fig.8-42: Creating an encoder axis – Configuration

In the "configuration" step, the drive has to be selected (scroll window drive assignment: in the example, RA1 or RA2).

RA2 and thus its drive address 2 are selected.



Deactivating the real axis RA2, its drive, its electronics, results in the deactivation of the real encoder axis EA1 whose encoder is connected to this electronics (see also [Modes to Operate a Real Axis](#), page 359).



An axis without position feedback (open loop) cannot be assigned to an encoder axis.

Axis assignment To address the axis, the PLC uses the axis number and axis name (here "3" or "EA1").

The encoder axis is shown only once in the Project Explorer:

Context menu If the context menu is opened on the folder **Motion -> Encoder Axes -> EA1** via right click, the following menu items are displayed:

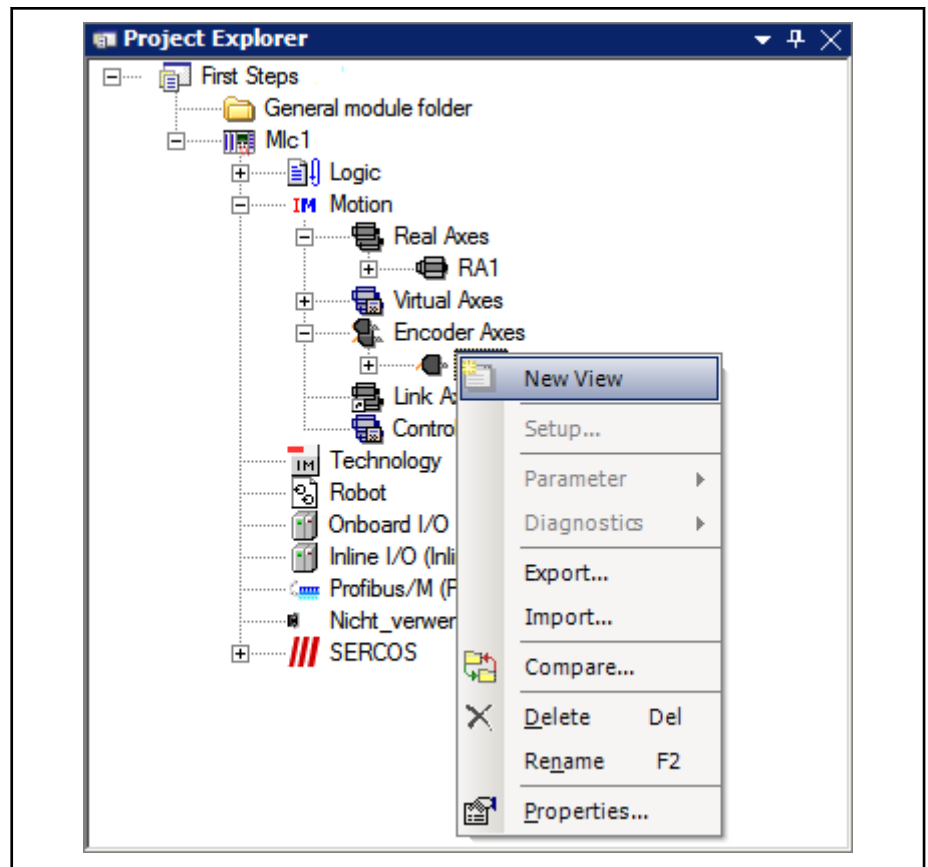


Fig. 8-43: Encoder axis - Context menu

The following can be selected:

- Export...
- Import...
- Compare...
- Delete
- Rename
- New View
- Properties...

Since, at this point in time, no connection with the control exists yet (offline mode), a number of commands are disabled, i.e. grayed out (for the complete description, see [Encoder axis - Context menu](#), page 267, or [Encoder Axis - Dialogs](#), page 266).

Configuration of the Measuring Encoder

The measuring encoder has to be configured below the sercos node in the real axis whose control electronics is used by the encoder. Selecting the encoder and pressing the right mouse button opens the configuration dialog.

Axes

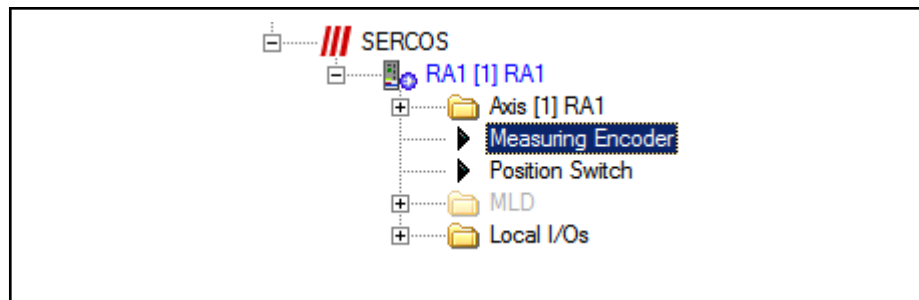


Fig.8-44: Measuring encoder connected to the real axis RA1

The configuration of an encoder axis requires a referenced measuring encoder.

The dialog distinguishes with regard to the measuring encoder type.

The left side of the dialog presets the encoder data.

On the right, the data for referencing is preset.

The **Set absolute position for measuring encoder** button triggers the processing of the complete referencing.



"Measuring encoder referenced" is the prerequisite for any further operation with the encoder axis.

The measurement encoder dialog is explained in [chapter 8.7.5 "Measuring Systems" on page 274](#).



For details, see /10/, [Required and Supplementing Documentations, page 14](#), Measuring encoder → Extension package "synchronization".

8.6.3 Encoder Axis - Dialogs

Overview on Dialogs

When opening the folder of the encoder axis, the individual dialogs of the axis – display of axis data and setting options for parameters – are provided.

First of all, it is assumed that the presettings cover a wide range of applications and that a modification is only required in exceptional cases.

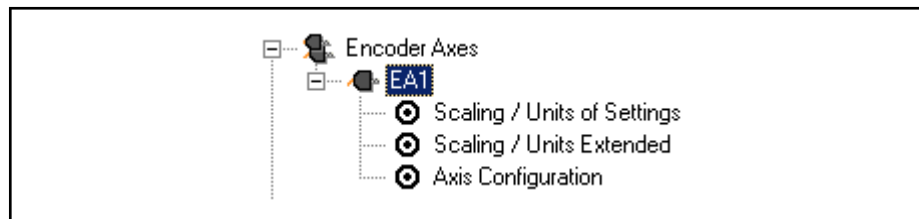


Fig.8-45: Dialogs of an encoder axis

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is specified.

The following dialogs are provided for the encoder axis:

- [Settings: Scaling/Units, page 277](#)
- [Scaling/Units Extended, page 278](#)
- [Axis Configuration, page 321](#)

8.6.4 Encoder axis - Context menu

Overview on Menu Items

The context menu of an encoder axis in online mode – in this example: EA1 – contains the following menu items:

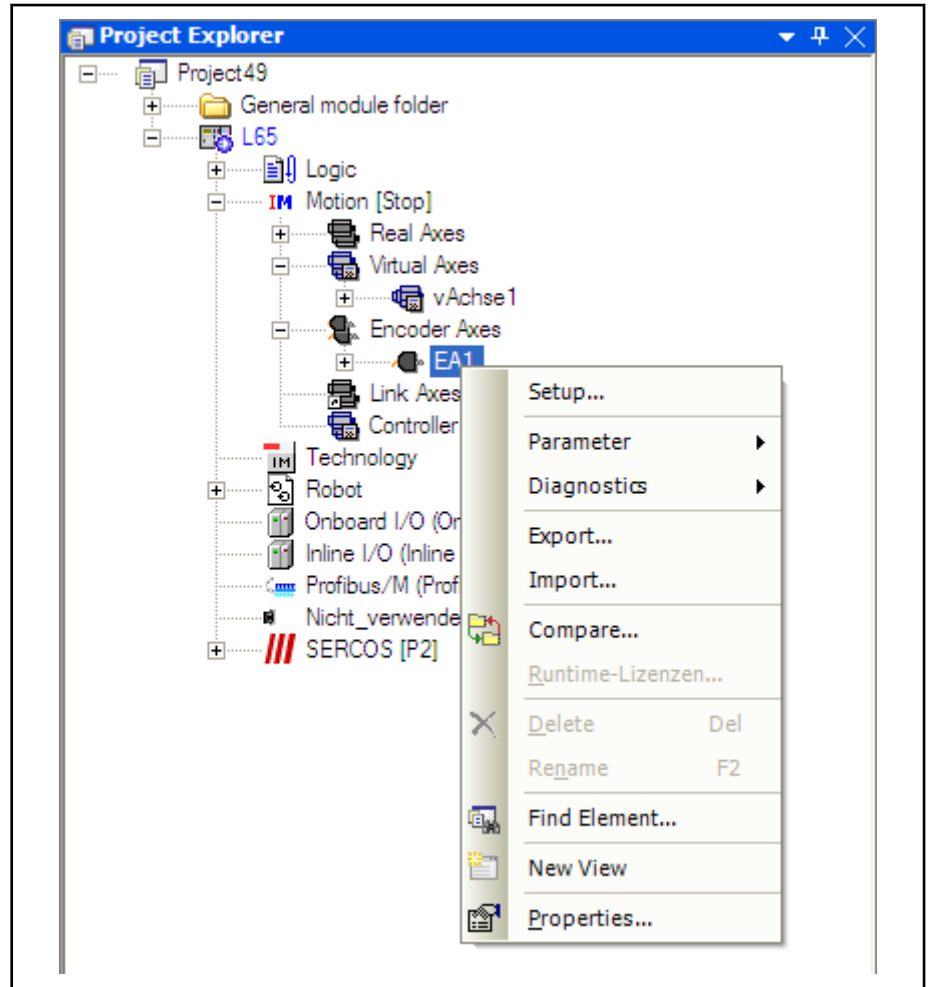


Fig. 8-46: Context menu of an encoder axis in online mode

- [Initial Commissioning](#), page 328
- [Parameters](#), page 334
- [Parameter Editor](#), page 133
- [Parameter Group](#), page 135
- [Status](#), page 341
- [Clearing Errors](#), page 342
- **Export...** - Saves axis elements in a file. This menu item is described in [Exporting](#), page 172
- **Import...** - Imports axis elements from a file This menu item is described in [Importing](#), page 174
- **Compare...** - Compares and merges objects, see [Comparing/Merging](#), page 176
- **Runtime licenses...** - See [Runtime Licenses](#), page 178

Axes

- **Delete** - The complete axis can be deleted from the project in **Offline mode** via this menu item
- **Rename** - The axis name can be changed in **offline mode** via this menu item



The axis data transferred to the PLC also changes with the axis name. The PLC program is also this way which may lead to compilation errors.

- **Find element...** - Allows searching for elements in the project
- **New View** - Displays the axis in an individual window
- **Properties...** - the dialog box shows the properties of the control entered during its configuration (see [Creating an Encoder Axis \(Real Master Axis\)](#), page 262)

Disabling an axis - Encoder axis



Disabling the real axis results in disabling a real encoder axis connecting its encoder to this electronics.



The parameter assignments of the axes to be deactivated are provided for export.

8.7 Dialogs

8.7.1 Overview on Dialogs

When opening the folder of an axis, the dialogs available for this axis are displayed below the axis nodes. These dialogs allow accessing status displays and axis settings. Most dialogs are only available if IndraWorks is online and on the control.

Axes

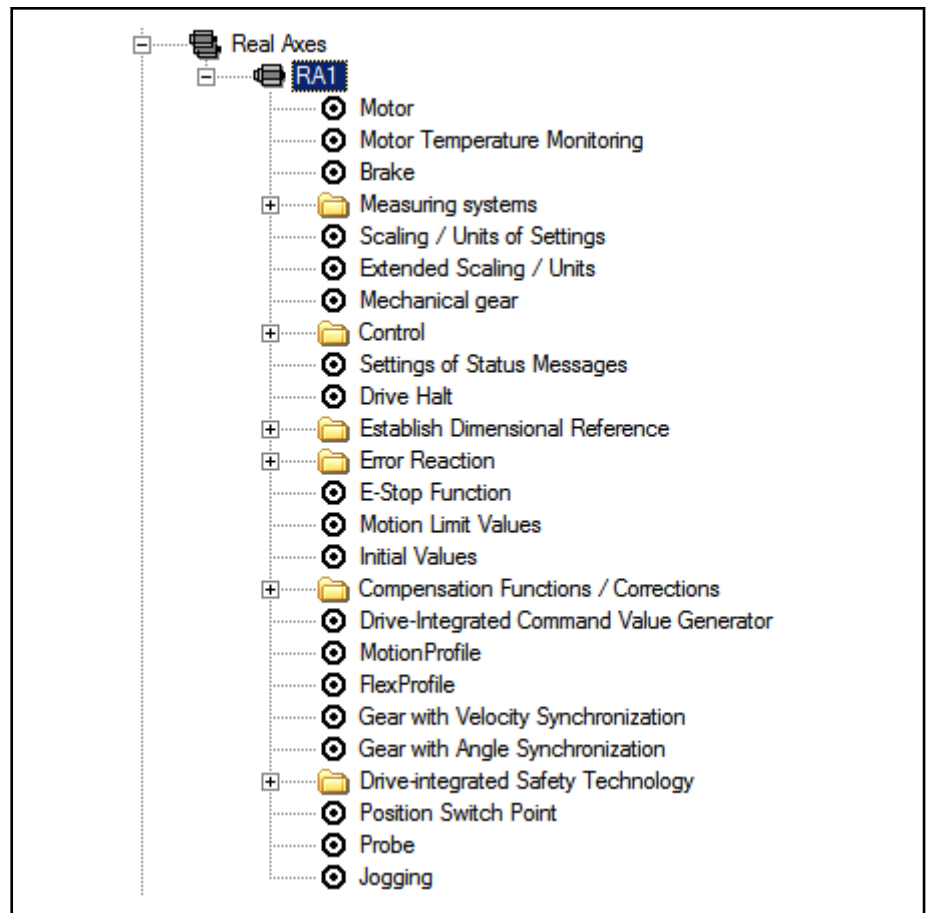


Fig. 8-47: Dialogs of a real axis (IndraDrive)

The presets already cover many use cases. A modification is only required in exceptional cases.

Depending on the axis type, a different scope of dialogs is available.

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Motor	x	x					
Motor temperature monitoring	x	x					
Brake	x	x					
Measuring Systems	x	x					
Setting: Scaling/units	x	x	x	x	x	x	x
Extended scaling/units	x	x	x	x	x	x	x
Mechanical gear	x	x					
Closed-loop control	x	x					
Status message settings	x	x	x	x	x		
Drive Halt (AH)	x	x					
Establish dimensional reference	x	x					
Error reaction	x	x					

Axes

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
E-Stop function	x	x					
Motion limit values	x	x	x	x	x		
Initial values	x	x	x	x	x		
Compensation functions/corrections	x	x					
Drive-integrated command value generator	x	x					
Cam table	x						
MotionProfile		x	x	x	x		
FlexProfile		x	x	x	x		
Gear with Velocity Synchronization	x	x	x	x	x		
Gear with Phase Synchronization	x	x	x	x	x		
Axis configuration						x	x
Drive-integrated safety engineering	x	x					
Position switching point	x	x					
Touch probe	x	x					
Jogging	x	x		x			

VA

Virtual axis

RA

Controller axis

EncA

Encoder axis

LnkA

Link axis

IPO

Interpolation

*Fig. 8-48:**Dialogs of different axis types*

If IndraDrive drive devices without a checkmark at "Interpolation in the Drive", EcoDrive CS, and SercosDrive are selected, the drive is controlled via the PackProfile.

As the IndraDrive and EcoDrive CS devices are drive devices specified in detail - in contrast to the SercosDrive - device-specific dialogs are provided for commissioning.

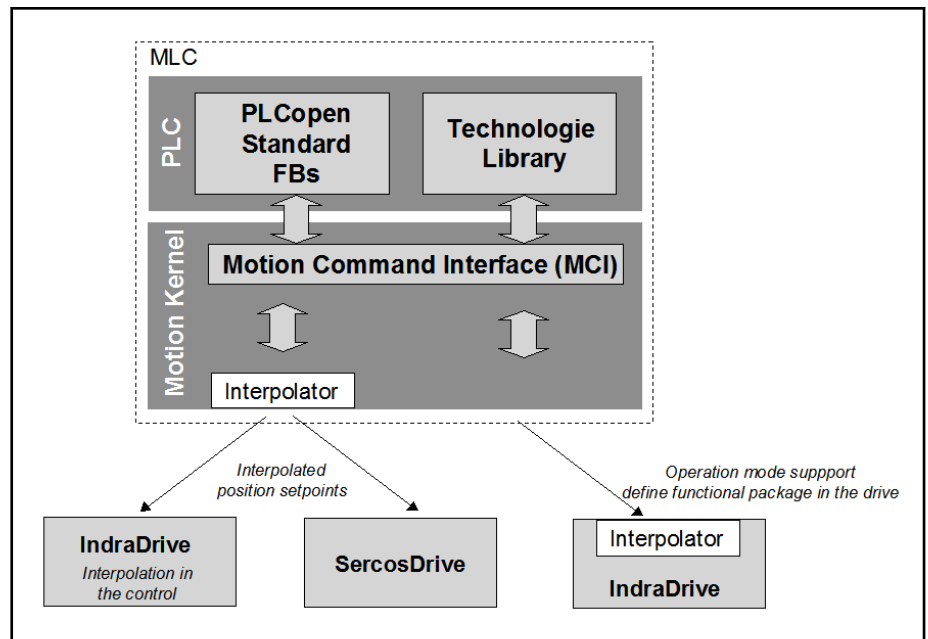


Fig. 8-49: Control principle for different drives

- [Motor, page 272](#)
- [Motor Temperature Monitoring, page 273](#)
- [Brake, page 273](#)
- [Measuring Systems, page 274](#)
- [Settings: Scaling/Units, page 277](#)
- [Scaling/Units Extended, page 278](#)
- [Mechanical Gear, page 280](#)
- [Closed-Loop Control, page 281](#) (reference to the drive documentation)
- [Settings of Status Messages, page 281](#)
- [Drive Halt, page 283](#)
- [Establishing Dimensional Reference, page 283](#)
- [Error Reaction, page 285](#)
- [E-Stop Function, page 288](#)
- [Motion Limit Values, page 289](#)
- [Initial Values, page 291](#)
- [Compensation Functions/Corrections, page 293](#) (reference to the drive documentation)
- [Drive-integrated Command Value Generator, page 294](#) (reference to the drive documentation)
- [Cam Table, page 295](#)
- [MotionProfile, page 300](#)
- [FlexProfile, page 307](#)
- [Gear with Velocity Synchronization, page 314](#)
- [Gear with Phase Synchronization, page 317](#)
- [Drive-integrated Safety Engineering, page 323](#)
- [Position Switching Point, page 323](#)
- [Touch Probe, page 324](#)

Axes

- [Jogging, page 327](#)

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is specified.



Dialogs exclusively referring to internal drive data are only described by a reference to the normally product-dependent original manuals.

The following dialogs can be opened via the context menu of the drive:

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Initial Commissioning, page 328	x	x	x	x	x	x	x
Exchanging the Drive/Firmware , page 329	x	x	x				
Parameters, page 334	x	x	x	x	x	x	x
Communication, page 336	x	x	x				
Diagnostics, page 341	x	x	x	x	x	x	x

Fig.8-50: Dialogs of different axis types reachable via context menu

8.7.2 Motor



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Fig.8-51: "Motor" dialog

The dialog shows the essential data of the motor and allows the setting of the cooling method used.

The data of this window is stored in the following S- and P-parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Motor type		S-0-0141
Motor type		P-0-4014
Cooling type		P-0-0640

For details, see /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Motor, Axis Mechanics, Measuring Systems".

8.7.3 Motor Temperature Monitoring



This dialog is available for IndraDrive for interpolation in the drive and in the control.

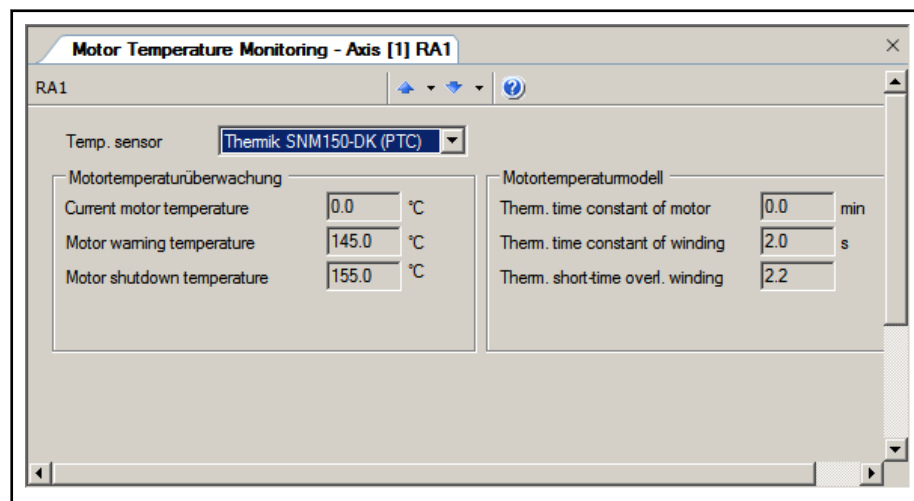


Fig. 8-52: "Motor Temperature Monitoring" dialog

The dialog sets the sensor type for the motor temperature monitoring. Furthermore, it informs about the thermal data of the motor.

The data is stored in the following S- and P-parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Motor temperature	S-0-0383	S-0-0383
Motor warning temperature	S-0-0201	S-0-0201
Motor shutdown temperature	S-0-0204	S-0-0204
Thermal time constant of motor	P-0-4035	P-0-4035
Thermal time constant of winding	P-0-4034	P-0-4034
Thermal short-time overload of winding	P-0-4037	P-0-4037

8.7.4 Brake



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Axes

Fig. 8-53: "Brake" dialog

Type and design of the brake set in this dialog is transferred as information to the controller.

The data is stored in the following S- and P-parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Holding brake, control word	P-0-0525	P-0-0525
Nominal load of holding system	P-0-0547	P-0-0547
Waiting time of the drive ON	S-0-0206	S-0-0206
Waiting time of the drive OFF	S-0-0207	S-0-0207
Maximum drive off delay time	S-0-0273	S-0-0273

For details, see /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "General Information on the Operation of Motors with IndraDrive".

8.7.5 Measuring Systems

Settings, Motor Encoder



This dialog is available for IndraDrive for interpolation in the drive and in the control.

IndraDrive supports three types of encoder:

- Motor encoder, the encoder is located on the motor shaft itself
- [Settings of Optional Encoder](#), page 275; the encoder is located on the load side to compensate, e.g. for disturbance variables of the gear
- [Configuration of the Measuring Encoder](#), page 265, the encoder has the highest degree of tolerance with regard to its application; it only uses the

drive electronics for the data communication and can e.g. realize a measuring wheel or be used as real master axis.

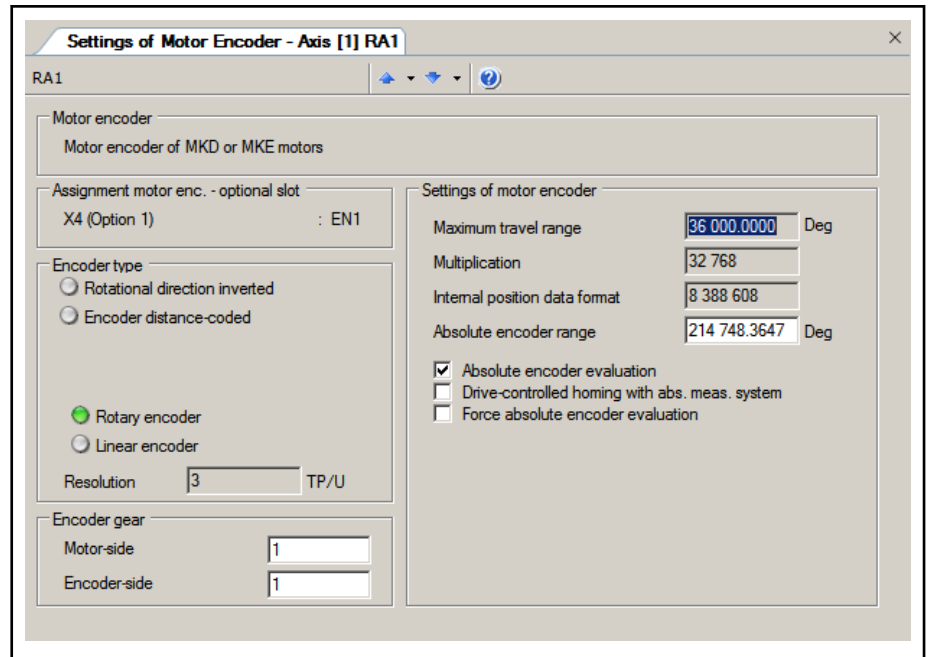


Fig. 8-54: "Settings of Motor Encoder" dialog

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Position encoder type 1	S-0-0277	S-0-0277
Encoder 1 resolution	S-0-0116	S-0-0116
Maximum traversing range	S-0-0278	S-0-0278
Multiplication 1 (motor encoder)	S-0-0256	S-0-0256
Position data format, internal	P-0-0129	P-0-0129
Gear 1 motor side (motor encoder)	P-0-0121	P-0-0121
Gear 1 encoder side (motor encoder)	P-0-0122	P-0-0122
Transmission ratio	(P-0-0122)/(P-0-0121)	(P-0-0122)/(P-0-0121)
Position encoder type 1	S-0-0277	S-0-0277

For details, see /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Motor, Axis Mechanics, Measuring Systems".

Settings of Optional Encoder



This dialog is available for IndraDrive for interpolation in the drive and in the control.

IndraDrive supports three types of encoder:

- [Settings, Motor Encoder](#), page 274, the encoder is located on the motor shaft

Axes

- Optional encoder. This encoder is located on the load side to compensate, e.g. for disturbance variables of the gear
- [Configuration of the Measuring Encoder, page 265](#), the encoder has the highest degree of tolerance with regard to its application; it only uses the drive electronics for the data communication and can e.g. realize a measuring wheel or be used as real master axis.

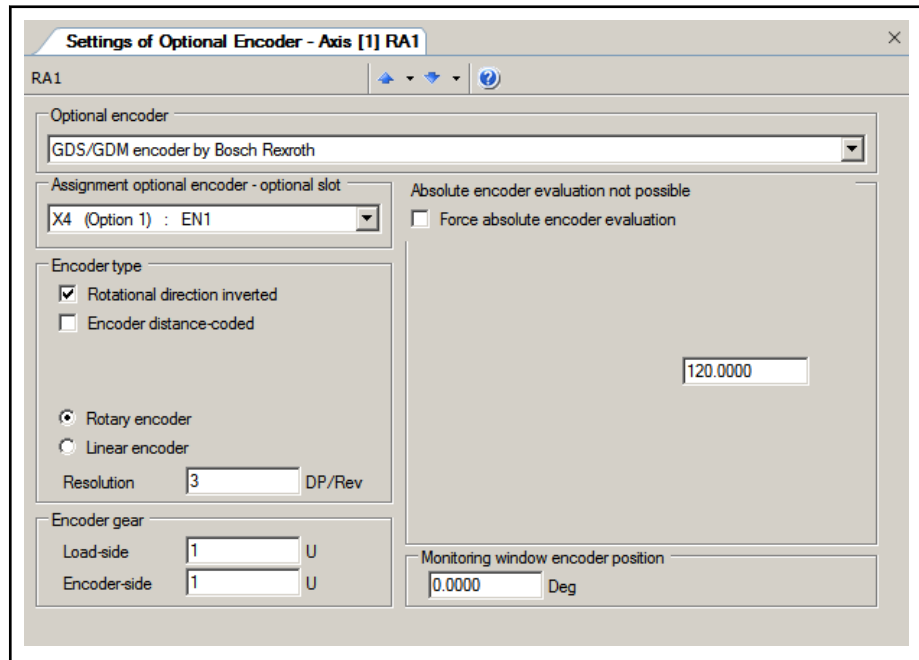


Fig. 8-55: "Settings of Optional Encoder" dialog

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Assignment optional encoder → optional slot	P-0-0078	P-0-0078
Encoder 2 resolution	S-0-0117	S-0-0117
Maximum traversing range	S-0-0278	S-0-0278
Multiplication 2 (optional encoder)	S-0-0057	S-0-0057
Position data format, internal	P-0-0129	P-0-0129
Gear 2 load side (optional encoder)	P-0-0124	P-0-0124
Gear 2 encoder side (optional encoder)	P-0-0125	P-0-0125
Transmission ratio	(P-0-0125)/(P-0-0124)	(P-0-0125)/(P-0-0124)
Position encoder type 2	S-0-0115	S-0-0115

For details, see /10/, /11/, [Required and Supplementing Documentations, page 14](#), section "Motor, Axis Mechanics, Measuring Systems".

8.7.6 Settings: Scaling/Units



This dialog is available for all axis types.

Fig. 8-56: Settings of "Scaling/Units of Settings" dialog (example of real axis)

The "Scaling/Units" dialog of settings derives the presettings for position, velocity and torque/force data from the same scheme.

Scaling type, scaling factor, scaling exponents

The scaling type of the data determines the format and the relation in which the data is exchanged between the drive and the control or the interface.

With the scaling factor and the scaling exponent, the scaling factor and the decimal places are specified in case of parameterized scaling. In case of a preference scaling, these two units are automatically set.

	Scaling type	Scaling factor	Scaling exponent
Position	A-0-0059	A-0-0060	A-0-0061
Velocity	A-0-0056	A-0-0057	A-0-0058
Torque/force	A-0-0050	A-0-0051	A-0-0052

Fig. 8-57: Overview on scaling parameters

Traversing velocity

If the axis is modulo scaled, the maximum traversing velocity during the interpolation in the control depends on how the Motion cycle time and the modulo value are set. This is limited by the "Nyquist-Shannon sampling theorem".

Max. traversing velocity = (0.5 x modulo [degree or mm]) / cycle time [s].

Theoretical maximum traversing velocity assuming a modulo of 360°

Motion cycle time in ms	Velocity in rpm
1 (IndraControl L45/L65 hardware only)	30000
2	15000
4	7500

Axes

8	3750
16	1875

Fig.8-58: Maximum traversing velocity with a modulo of 360°

Scaling type, format of the position data

Either the scaling type rotary or translatory scaling (A-0-0053, A-0-0056, A-0-0059 bit 1...0 each) can be selected. The position data format allow the settings "absolute" and "modulo" (A-0-0059, Bit 7) .

If the format is set to "modulo", the modulo value has to be entered in degrees or feed distance in mm into the "Modulo value" field.

The following parameters are used by the dialog:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Scaling type for position data	A-0-0059						
Scaling type for velocity	A-0-0056						
Scaling type for acceleration	A-0-0053						
Modulo value	A-0-0045						
Feed distance of the slave drive	A-0-0046						

Detailed data can be entered into [Scaling/Units Extended, page 278](#),.

8.7.7 Scaling/Units Extended

 This dialog is available for all axis types.

The dialog box provides supplementary detailed data for [Settings: Scaling/Units, page 277](#),.

If the axis is an IndraDrive, the following dialog is displayed:

Weighting type of position data

☒ Rotatory	☒ Load reference	☒ Preferred scaling	Position resolution, rotatory Unit
☐ Translatory	☐ Motor reference	☐ User-defined	
☐ Absolute	☐ Negation		
☒ Modulo	360 Deg	Resolution 0.0001	
			3 600 000 v

Velocity Data Scaling Type

☐ Rotatory	☒ Load reference	☒ Preferred scaling	Factor 1
☒ Translatory	☐ Motor reference	☐ User-defined	Exponent -6
	☐ Negation	Resolution 0.001	Unit mm/min v

Acceleration Data Scaling Type

☒ Rotatory	☒ Load reference	☒ Preferred scaling	Factor 1
☐ Translatory	☐ Motor reference	☐ User-defined	Exponent -3
☐ Ramp time		Resolution 0.001	Unit rad/s² v

Scaling type for torque/force data

☒ Rotatory	☒ Load reference	☒ Preferred scaling	Factor 1
☐ Translatory	☐ Motor reference		Exponent -2
☐ In percent	☐ Negation	Resolution 0.01	Unit Nm v

Torque/force reference value

Torque/Force reference value	1000

Scaling type for temperature data

☒ Degree Celsius (°C)	☐ Degree Fahrenheit (°F)

Fig.8-59: "Scaling/Units Extended" dialog for IndraDrive axes

The dialog of the other axis types provides respectively less setting options.

The following parameters are analyzed:

[illegible]

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Scaling type for acceleration	A-0-0053						
Scaling factor for acceleration data	A-0-0054						
Scaling exponent for acceleration data	A-0-0055						
Force/torque	x						
Scaling type for force/torque data	A-0-0050						
Scaling factor for torque/force data	A-0-0051						
Scaling exponent for torque/force data	S-0-0052						
Temperature	x						
Scaling type for temperature data	A-0-0063						

8.7.8 Mechanical Gear

 This dialog is available for IndraDrive for interpolation in the drive and in the control.

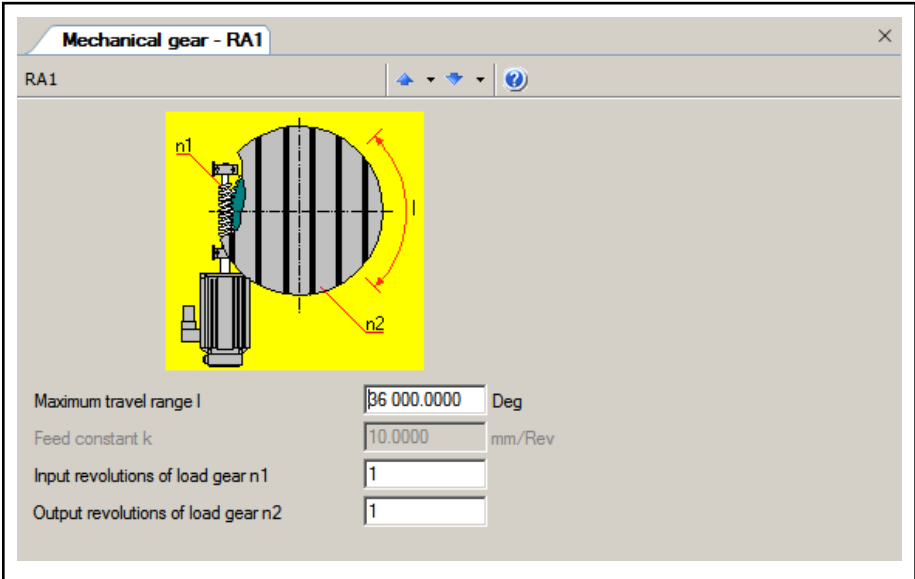


Fig.8-60: "Mechanical gear" dialog

The data of this window is stored in the following S- and P-parameters:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Maximum traversing range	S-0-0278	
Load gear input revolutions	S-0-0121	
Load gear output revolutions	S-0-0122	
Transmission ratio	(S-0-0122)/(S-0-0121)	
Feed constant	S-0-0123	

The parameter S-0-0123 is only relevant for linear axes driven by a rotary motor. The feed constant is the distance covered by the axis, when the gear output shaft or motor shaft performs one rotation. Determination of the feed constant with different mechanical transmission elements. For ball screw spindle, the default value is:

Feed constant = spindle lead type. value 10.00 mm/rev.

For details, see /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Motor, Axis Mechanics, Measuring Systems".

8.7.9 Closed-Loop Control



This dialog box is available for the interpolation in the drive and in the control.

The basic principles of the closed-loop control dialogs are described product-specifically in /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Overview of Drive Control".

Support is provided by the implemented online help.

8.7.10 Settings of Status Messages



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive, virtual axis, controller axis.

The dialog box evaluates the status of the axis (A-0-0021).

The messages

- "A-0-0021, Standstill (bit 5)"
- "A-0-0021, Actual velocity = velocity command value (bit 4)"
- "A-0-0021, In position (bit 6)"
- "A-0-0021, Synchronous position (bit 7)" and
- "A-0-0021, Synchronous velocity" likewise (bit 7)

depend on whether the respective value is within its window.

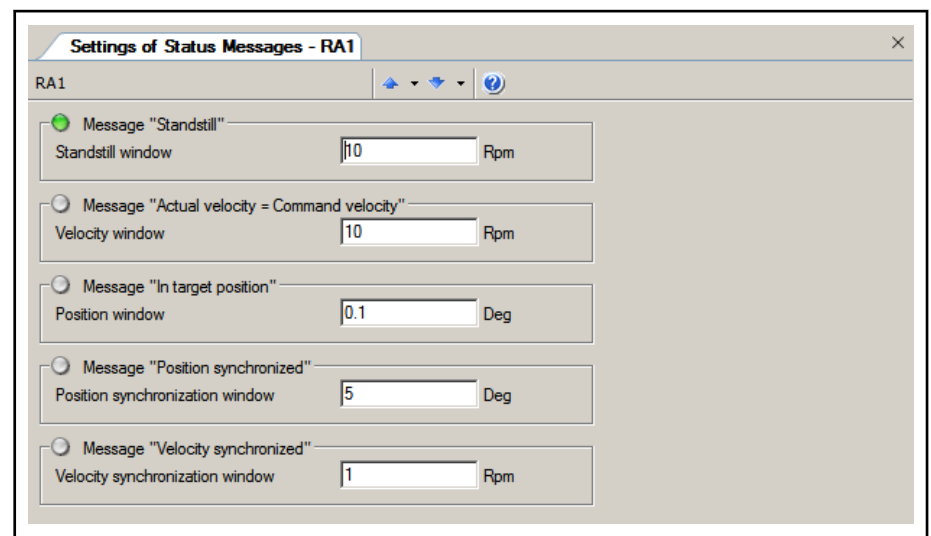


Fig. 8-61: "Settings of Status Messages" dialog

Axes

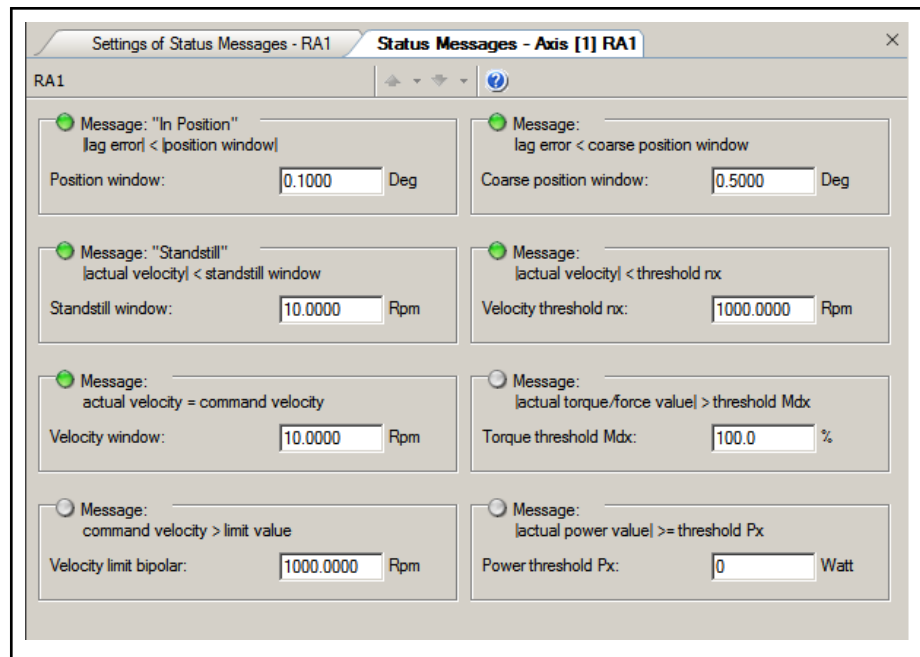


Fig. 8-62: Status Messages – Corresponding drive dialog for IndraDrive

The values are filed in the following parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Axis status	A-0-0021						
Velocity	x						
Actual velocity value	A-0-0102						
Standstill window	A-0-0222						
Velocity window	A-0-0223						
Synchronization run window, velocity	A-0-0623						
Position							
Actual position value	A-0-0100						
Positioning window	A-0-2795						
Synchronization window	A-0-0622						

A-0-0222, Standstill window The "Standstill" status bit is set if "A-0-0102, Actual velocity value" is in "A-0-0222, Standstill window" (-> A-0-0021, Bit 5).

A-0-0223, Velocity window For any velocity command value, the "A-0-0102, Actual velocity value" is in the window defined by "A-0-0223, Velocity window" (-> A-0-0021, Bit 4).

A-0-2795, Position window For any command position value, the "A-0-0100, Actual position value" is in the window defined by "A-0-2795, Position window" (-> A-0-0021, Bit 6).

A-0-0622, Synchronization window, position With the "Synchronized Motion" function blocks with subordinate position control (MC_GearInPos, MC_CamIn, MB_MotionProfile, ML_FlexProfile), if the difference between the position command value and the actual position value is smaller than the "A-0-0622, Synchronization window, position", bit 7 in parameter "A-0-0021, Axis status" is set.

A-0-0623, Synchronization window, velocity

With the "Synchronized Motion" function blocks (MC_GearIn), if the difference between the velocity command value and velocity actual value is smaller than the "A-0-0623, Synchronization window, velocity", bit 7 in parameter "A-0-0021, Axis status" is set.

8.7.11 Drive Halt



This dialog box is available for the interpolation in the drive and in the control.

The dialog summarizes the most important preset values of the drive (rapid) stop.

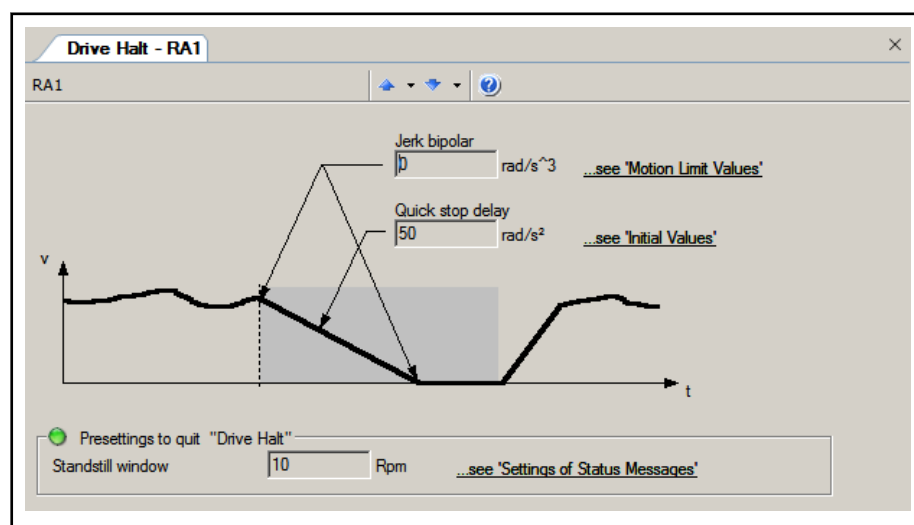


Fig. 8-63: "Drive Halt" dialog

The individual parameters are pre-adjusted in different dialogs:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Jerk limit value, bipolar Motion Limit Values, page 289		A-0-0036
Rapid stop deceleration Initial Values, page 291		A-0-0224
Standstill window Settings of Status Messages, page 281		A-0-0222
Axis status Axis State Dialogs for Real Axes, page 363		A-0-0021 (Bit 15)

8.7.12 Establishing Dimensional Reference

Motor Encoder



This dialog box is available for the interpolation in the drive and in the control.

Axes

The dialog is intended for setting the dimensional reference for the motor encoder.

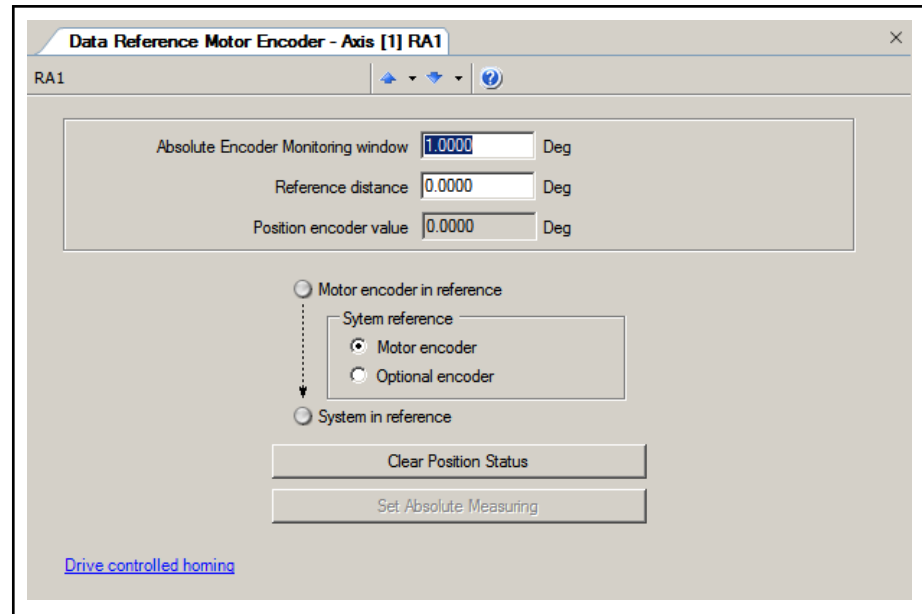


Fig. 8-64: Dialog of the dimensional reference motor encoder

P-0-0095, Absolute encoder monitoring window for motor encoder

When switching-on a drive with absolute motor encoder, it is checked whether the actual position value deviates from the actual position value at the last switch-off.

If the deviation exceeds the value specified in this parameter, the error message "F2074 Actual position value 1 outside the absolute encoder window" is output. When switching-off the drive, the current encoder data of the absolute motor encoder is stored in "P-0-0177, Absolute encoder buffer motor encoder)".

S-0-0052, Reference value

In this parameter, the desired actual position value is inserted with regard to a certain axis position (reference position). With the command "P-0-0012, C0300 Command: Set absolute position", the value inserted in parameter S-0-0052 becomes effective at the reference position as actual position value.

Thus, a certain position reference system can be established for an axis or an existing reference system can be replaced by a new one (switching).

S-0-0051, Actual position value encoder 1

The actual position value encoder 1 displays the current position of the motor encoder. After the drive was switched on, the communication phases are run through and the actual position value is initialized with the command "S-0-0128, C0200 Preparation of switching to phase 4", i.e. it is set to its initial value.

Furthermore, the following parameters are involved

- S-0-0403, Actual position value status
- S-0-0147, Homing parameter

Optional Encoder



This dialog box is available for the interpolation in the drive and in the control.

The dialog is intended for setting the dimensional reference of an optional encoder.

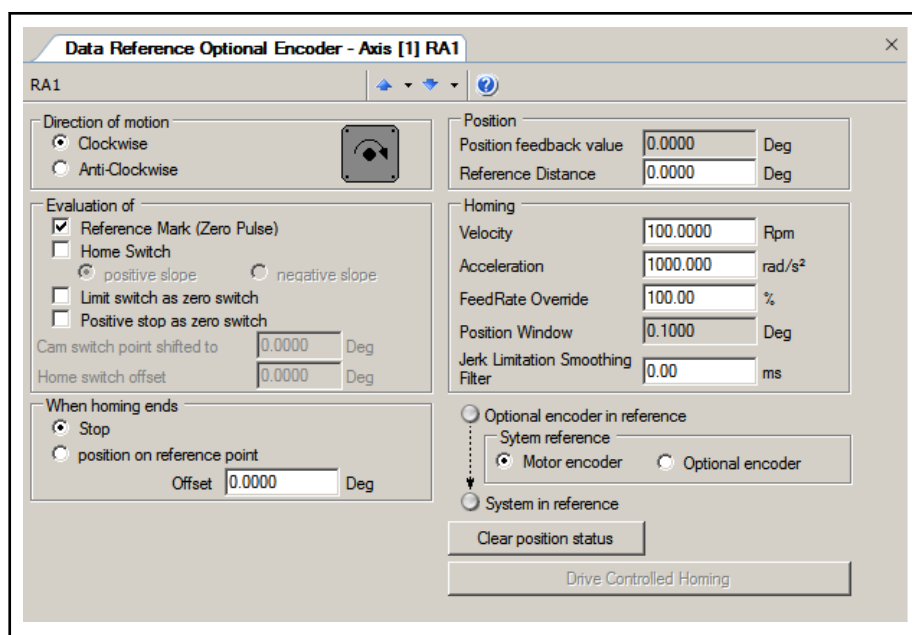


Fig. 8-65: Dialog of the dimensional reference optional encoder

P-0-0096, Absolute encoder monitoring window for optional encoder

When switching-on a drive with absolute encoder, it is checked whether the actual position value deviates from the actual position value during the last switch-off.

If the deviation exceeds the value specified in this parameter, the error message "F2075 Actual position value 2 outside the absolute encoder window" is output. When switching-off the drive, the current encoder data of the absolute optional encoder is stored in "P-0-0178, Absolute encoder buffer, optional encoder".

S-0-0054, Reference value 2

In this parameter, the desired actual position value is inserted with regard to a certain axis position (reference position). With the command "P-0-0012, C0300 command: Set absolute position", the value inserted in parameter S-0-0053 becomes effective at the reference position as the actual position value.

Thus, a certain position reference system can be established for an axis or an existing reference system can be replaced by a new one (switching).

S-0-0053, Actual position value encoder 2

The actual position value encoder 2 displays the current position of the optional encoder. After the drive was switched on, the communication phases are run through and the actual position value is initialized with the command "S-0-0128, C0200 Preparation of switching to phase 4", i.e. it is set to its initial value.

Furthermore, the following parameters are involved

- S-0-0403, Actual position value status
- S-0-0147, Homing parameter

8.7.13 Error Reaction

Drive



This dialog is available for IndraDrive for interpolation in the drive and in the control.

This dialog configures the reaction in case of error in the drive.

Axes

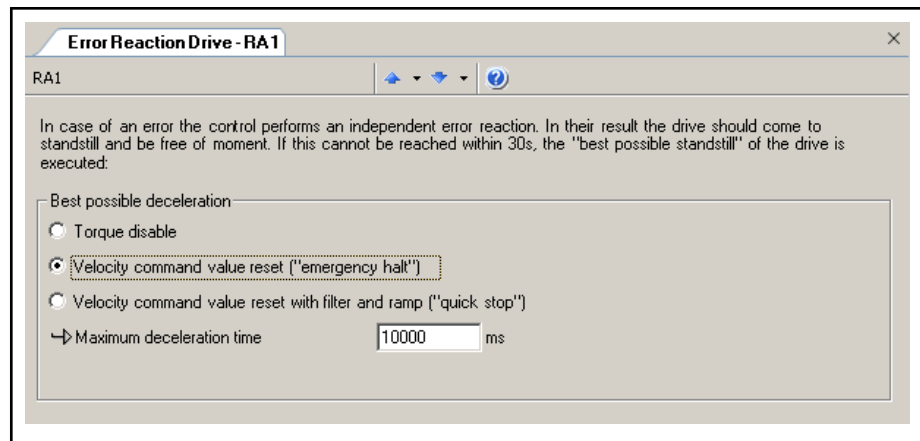


Fig. 8-66: "Error Reaction Drive" dialog

For each error, the type of the "Optimum deceleration procedure" as well as the maximum braking time can be indicated.

Error reaction can be switched off

For drive errors, the automatic error reaction set by default can be switched off individually for each drive. Thus, even a drive with errors can be synchronized and delayed with a master axis. If the error reaction is switched off, there is a period of 30 seconds starting from the error detection to decelerate the drive. After this period, the best possible deceleration is automatically executed by the drive.

For the **interpolation in the drive** the switch-off becomes effective for all **F2/F3** drive errors. F2/F3 drive errors are evaluated as F0 errors on the control and thus there is no error reaction from the control. Irrespective of that fact, the drive executes an autarkic error reaction starting from error class 4 (e.g. sercos ring interruption) according to the drive documentation.

For the **interpolation on the control** the switch-off becomes effective for **all** drive error classes. All drive errors are evaluated as F0 errors on the control and thus there is no error reaction from the control. Irrespective of that fact, the drive executes an autarkic error reaction starting from error class 4 (e.g. sercos ring interruption) according to the drive documentation.



In case of a switched off error reaction, the drive does not automatically delay if a non-fatal error occurs. This behavior can damage the machine. The drive has to be set to standstill in the PLC user program.

An automatic error reaction for drive errors set by default can be switched off as follows:

- In the IndraWorks drive dialog "Error reaction --> Error reaction drive", enable the option "30s NC external NC reaction, then best possible deceleration (P-0-0117 Bit 0 = TRUE)
- Set the additional axis parameter A-0-0651 Bit 0 = TRUE. Thus, no automatic error reaction is executed for drive errors by the control (For details, refer to the parameter documentation of the A-0-0651)
- The PLC user program is responsible for the deceleration of the axis containing errors. It is recommended to always query the error bit of the drive (e.g. AxisData[].Axis_Error) cyclically and to start an error reaction in case of a set bit

Example:

Axes

For non-fatal drive errors of a real slave axis, only the master axis is delayed in the PLC user program via the MC_Stop FB. The slaves axis remains synchronized even in case of an error and delays with the master axis. After these 30 seconds, the best possible deceleration starts on the slave axis (the synchronous link is canceled in standstill). After the error cause has been cleared, the error is deleted by the "MB_Command" (C-0-1030) FB and the PLCopen Motion FB of the slave axis (e.g. MB_GearInPos) has to be restarted via a positive edge at the "Execute" input.



If the error reaction is switched off, the drive error of the PLCopen state remains (the "ErrorStop" state remains inactive). Furthermore, the "Error" output is not set at the Motion function blocks (e.g. MC_MoveVelocity). Irrespective of that fact, the "Error" bit **AxisData[].Axis_Error** and A-0-0021 are set.

Since the PLCopen state "Errorstop" does not become active in case of error, the drive errors cannot be deleted via the PLCopen function block "MC_Reset". An alternative is the command "A-0-1030, Command: Clear all axis errors" or "C-0-1030, Command: Clear all control error" via the MB_Command function block.

Power Section



This dialog is available for IndraDrive for interpolation in the drive and in the control.

This dialog configures the reaction in case of error in the power section.

Fig. 8-67: "Error Reaction Power Section" dialog

The following parameters are displayed:

P-0-0460, Module link, control word

In this parameter, the information sent from a device (inverter or converter) to the module bus is displayed. Due to the hierarchical structure of the module bus, only the information with the highest priority becomes effective (read only)!

Axes

P-0-0461, Module network, status word

In this parameter, the currently active information of the module bus is displayed.

P-0-0118, Power supply, configuration

In this parameter, the settings regarding error messages and error reactions for drives are made, which are connected with each other via the DC intermediate circuit and the module bus (drive package). Furthermore, the handling of the DC intermediate circuit undervoltage is determined.



The DC intermediate circuit voltage (power bus) is not switched off in case of non-fatal errors when the "Drive package" is in operation!

8.7.14 E-Stop Function



This dialog box is available for the interpolation in the drive and in the control.

The activation of the E-Stop function and the determination of the drive response is carried out via the parameter "P-0-0008, Activation of E-Stop function".

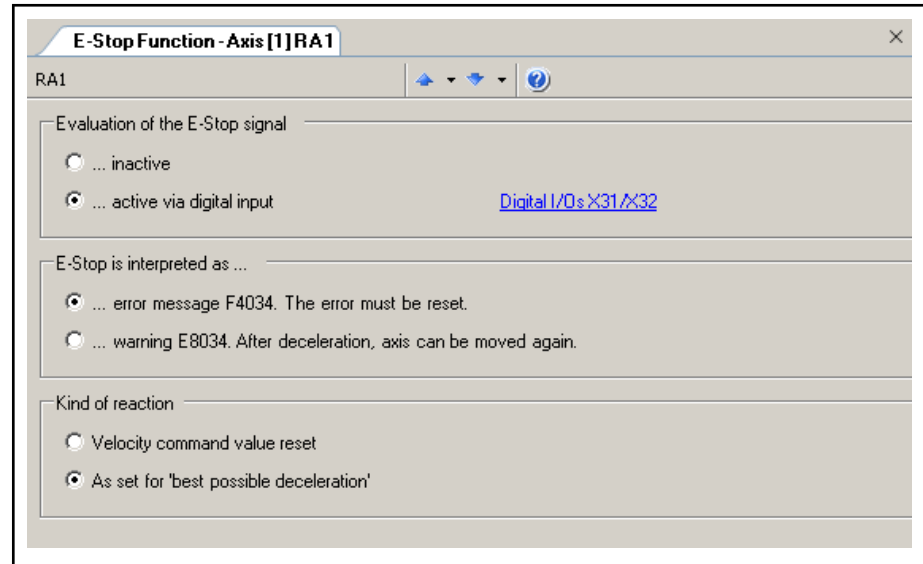


Fig.8-68: E-Stop functionality

Through the activation of "P-0-0008, E-Stop input (Bit 0 = 1)" and the allocation of the bit 0 condition of "P-0-0223, E-Stop input" to a digital input, the drive is initiated to execute the response to shut down the drive defined via P-0-0008, if the E-Stop input has a 0V level.

"P-0-0223, E-Stop input" has to be assigned via parameters "P-0-0300, Digital I/Os, allocation list" and "P-0-0301, Digital I/Os, bit numbers" to a digital input. The definition of the input can be influenced via the dialog box that is located behind the "Digital I/O X31/X32" link.

The data of this window is stored in the following S- and P-parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
E-Stop input		P-0-0008
E-Stop input		P-0-0223

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Digital I/O assignment list	P-0-0223	
Digital I/O bit numbers	P-0-0301	

For details see /10/ and /11/, [Required and Supplementing Documentations](#), page 14, "E-Stop function" or online help.



The input is always "0" active!

8.7.15 Motion Limit Values



This dialog box is available for the interpolation in the drive and in the control.

The dialog provides an overview on the preset motion limit values of the real axis.

Fig. 8-69: Motion limit values



Switching to the corresponding drive dialog (see [fig. 8-72 "Motion Limit Values - Corresponding drive dialog"](#) on page 291) is only possible with interpolation in the control.

The dialog provides the respective parameters for the control (A-parameters) as well as for the drive (S- and P-parameters) as table.

The parameters for the drive are not displayed if the drive is disabled.

Axes

The parameters for "Settings Drive" cannot be changed for an IndraDrive with interpolation in the drive, since the A-parameter values are automatically transferred to the respective S- and P-parameters. During interpolation in the control, there is no automatic transmission. Thus, the A-parameter as well as the S- and P-parameters can be set separately in this case.

The buttons for copying the values are only displayed if the settings can be made for the drive.

The following parameters are evaluated:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Position	x						
Position limit value monitoring	A-0-0029/S-0-0055		A-0-0029				
Actual position value	A-0-0100/S-0-0051		A-0-0100				
Position limit value, positive	A-0-0030/S-0-0049		A-0-0030				
Position limit value, negative	A-0-0031/S-0-0050		A-0-0031				
Travel range limit switch setting	A-0-0028/P-0-0090		A-0-0028				
Velocity	x						
Positive velocity limit value	A-0-0032/S-0-0038		A-0-0032				
Negative velocity limit value	A-0-0033/P-0-0090		A-0-0033				
Bipolar velocity limit value	S-0-0091						
Acceleration	x						
Bipolar acceleration limit value	A-0-0034/S-0-0138		A-0-0034				
Jerk	x						
Bipolar jerk limit value	A-0-0036/S-0-0349		A-0-0036				
Torque/force	x						
Torque/force limit value, bipolar	A-0-0038/S-0-0092		A-0-0038				

Monitoring the travel range exceedance

The value of "A-0-0029, Bit 4 or S-0-0055 Bit 4 position polarities" decides whether the position limit monitoring of the axis is active (Bit 4 = TRUE).

If the position limit monitoring is active, "Actual position value (A-0-0100)" displays the position of the axis and the adjustable "A-0-0030, Positive position limit value", "A-0-0031, Negative position limit value" show the valid motion limits.

The user can now decide, whether a "Fatal Warning" is output with subsequent shutdown of the drive or whether an "Error" is triggered according to the error reaction preset in "P-0-0119, Optimum standstill procedure".

Actual position value	Smaller than negative position limit value	Higher than positive position limit value
Warning	E0112053	
Error	F6116030	F6116029

Fig. 8-70: Diagnostics as a response to the exceeding of the position limit values

Monitoring limit values

Irrespective of the activation of the position limit value monitoring, the other limit values can either be accepted with their default setting or changed user-

Axes

specifically. Bipolar limit value means that the same value is effective in positive as well as in negative direction.

If said limits are exceeded in either direction, a warning is output (see following table).

Actual value	Smaller than negative limit value	Higher than positive limit value
Velocity	E0112063	
Acceleration	E0112039	

Fig. 8-71: Warnings as a response to the exceeding/falling below of the limits

The jerk and torque/force limit values do neither trigger a warning nor an error message, but they limit the respective command values [for detailed information, see parameter description (A-0-0036) and (A-0-0038)].



The jerk limit value is only effective in the operating states "Discrete" and "Continuous" Motion.

Corresponding drive dialog

Fig. 8-72: Motion Limit Values - Corresponding drive dialog



Further dialogs to configure the drive can be opened (e.g. axis control, current limitation) in the corresponding drive dialog.

8.7.16 Initial Values



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive, virtual axis, controller axis.

The dialog box displays the initial values used for the PLCopen function blocks. Starting point is the state after reboot:

Axes

Initial Values - RA1

RA1

Status
No movement is performed, the displayed values are currently NOT used

Parameter

	Initial Values	Effective command values	
Position	These values are predetermined by the protective circuit in the Motion-FBs in the PLC program	0	Deg
Target position		0	Deg
Velocity		0.1666667	Rpm
Acceleration		0.01745329	rad/s ²
Deceleration		0.01745329	rad/s ²
Jerk		0	rad/s ³

Quick stop parameter

Quickstop-Delay: 50 rad/s²

Emergency jerk: 0 rad/s³

Direction

(only effective at MC_MoveAbsolute)

☐ Positive direction
☐ Negative direction
☒ Shortest distance

Window shortest distance: 1 Deg

Fig.8-73: Initial values after a reboot

The following parameters are evaluated:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Position	x						
Command position	A-0-2200						
Target position	A-0-2201						
Velocity	x						
Command velocity	A-0-2202						
Direction of motion	A-0-0203						
Acceleration	x						
Command acceleration	A-0-2203						
Rapid stop deceleration	A-0-0224						
Jerk	x						
Jerk	A-0-0216						
Jerk limit value, bipolar	A-0-0036						
Rapid stop jerk	A-0-0225						

The values of the parameters given in the parameter monitor result from the initialization at startup. They are of no relevance, since no motion has yet been made.

The values displayed are overwritten with the values of the respectively active PLCopen function block.

Axes

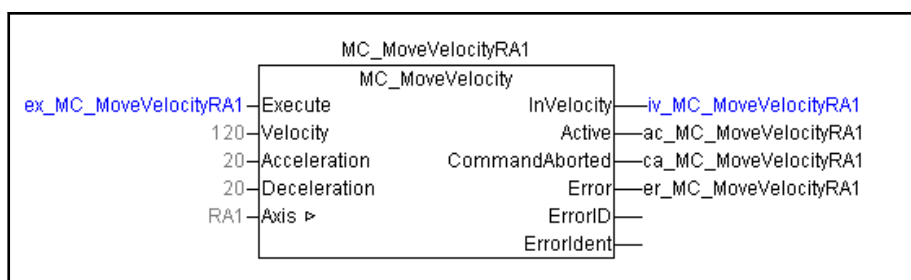


Fig.8-74: Active PLCopen FB: MC_MoveVelocity

Parameter	Initial Values	Effective command values
Position		0 Deg
Target position	These values are predetermined by the protective circuit in the Motion-FBs in the PLC program	274.0431 Deg
Velocity		120 Rpm
Acceleration		20 rad/s ²
Deceleration		20 rad/s ²
Jerk	0	0 rad/s ³

Quick stop parameter

Quickstop-Delay: 50 rad/s²

Emergency jerk: 0 rad/s³

Direction (only effective at MC_MoveAbsolute):

- ☐ Positive direction
- ☐ Negative direction
- ☒ Shortest distance

Window shortest distance: 1 Deg

Fig.8-75: Command values based on the initial values and the function block

The velocity (120 min⁻¹), acceleration (20 rad s⁻²) and deceleration (20 rad s⁻²) values were applied by the function block.

The previous position and distance values were kept. They change e.g. with an active MC_MoveRelative.

8.7.17 Compensation Functions/Corrections

Encoder Correction



This dialog is available for IndraDrive for interpolation in the drive and in the control.

The quality of processing and accuracy the axis follows a preset path is affected by several factors.

One factor is the accuracy used by a measuring system to identify the position of an axis or a shaft. IndraDrive controller devices provide the option to improve system-dependent inaccuracies of position-measuring systems with sinusoidal signals. Thus, the following behavior of a drive or an axis regarding the position command values and velocity command values is improved. Frequently, the bandwidth of the control loops can also be improved with a higher adjustable control loop amplification.

Axes

The basic principles of the "Compensation functions, encoder correction" are described product-specifically in /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Encoder Correction".

Support is provided by the implemented online help.

Backlash on Reversal Correction



This dialog is available for IndraDrive for interpolation in the drive and in the control.

By means of the backlash on reversal correction, a clearance can be easily corrected in the axis mechanics.

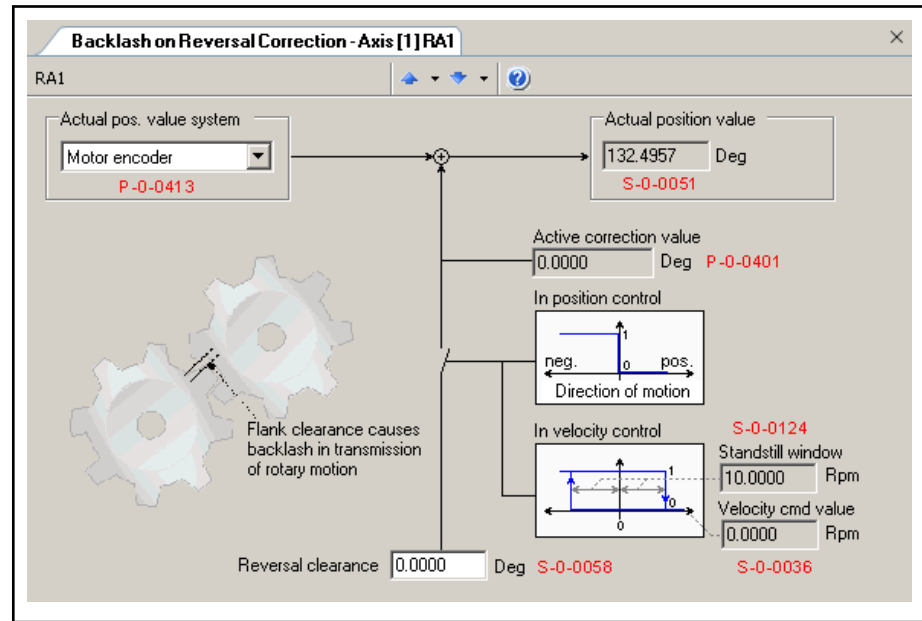


Fig.8-76: "Backlash on reversal correction" dialog

Parameters used:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
AxisCorr. Control word		P-0-0413
AxisCorr. active correction value		P-0-0401
Actual position value encoder 1		S-0-0051
Backlash		S-0-0058
Standstill window		S-0-0124
Velocity command value		S-0-0036

Further information on the parameters to be used are provided in the IndraDrive firmware description.

8.7.18 Drive-integrated Command Value Generator



This dialog is available for IndraDrive for interpolation in the drive and in the control.

The basic principles of the "drive-integrated command value generator" are described product-specifically in /10/, /11/, [Required and Supplementing Doc-](#)

umentations, page 14, chapter "Drive-Integrated Command Value Generator".

Support is provided by the implemented online help.

8.7.19 Cam Table

General Information



This dialog box is only available if the checkmark before the option "Interpolation in Drive" was **not** removed during the creation of the real axis.

If a real axis operates as slave axis in the MC_CamIn Motion function block, the drive runs in the operating state "cam table with real/virtual master axis". In this operating state, there is a fixed relation between a given master axis position and a slave axis. Master axes (masters) can either be virtual axes or encoder axes.

Based on the virtual master axis position (actual position value, master), the effective master axis position (actual position value in the actual value cycle) is generated with "Master Axis Position Offset", "Gear" and "Dynamic Synchronization".

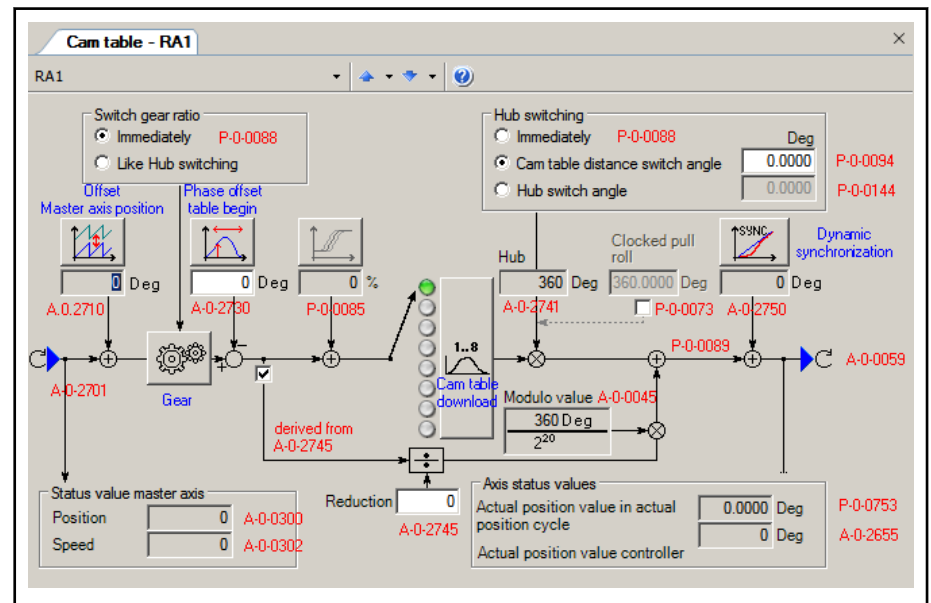


Fig. 8-77: "Cam Table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos-Drive	VA	RA	EncA	LnkA
Control word, synchronization operating states	P-0-0088						
ELS, Phasing, additive position command value	A-0-2710						
ELS, PhasingTableAngle, cam table beginning	A-0-2730						
Dynamic phase offset	P-0-0085						

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Switching phase, cam, distance	P-0-0144						
Switching phase, cam	P-0-0094						
ELS, CamIn, cam distance	A-0-2741						
ELS, PhasingSlave, additive position command value	A-0-2750						
ELS, CamIn, cam table preselection	A-0-2740						
Master axis, actual position value	A-0-0300						
Master axis, actual velocity value	A-0-0302						
Actual position value in the actual value cycle	P-0-0753						
ELS, position command value	A-0-2655						
ELS, gear reduction	A-0-2745						
Scaling type for position data	A-0-0059						

The cam table dialog displays the master axis parameters when the drive is in the operating state "Cam table with real/virtual master axis". The parameters are modified from the PLC program by the MC_CamIn Motion function block.

Gear switching Definition of the gear switching either "immediately" or "like distance switching" (P-0-0088).

Distance switching Definition of the distance switching either "immediately" or definition of the cam table switching phase or the distance switching phase (P-0-0088).

Master axis position offset Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value " as well as "A-0-2600, Additive master axis position, process controller".

Gear settings Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Cam table - Phase offset Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller".

Dynamic phase offset This functionality is currently not supported by the IndraMotion MLC.

Cam table selection Selection and download of a cam table created with the CamBuilder to a cam table of the drive.

Dynamic synchronization – Phase synchronization Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2761, ELS, PhasingSlave Vel, additive command velocity" and the "A-0-2615, ELS, additive velocity command value, process controller".

Master Axis Position Offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position

Axes

command value " as well as "A-0-2600, Additive master axis position, process controller". With the "P-0-0750, Master axis revolutions per master axis cycle", the modulo range of the master axis position is specified.

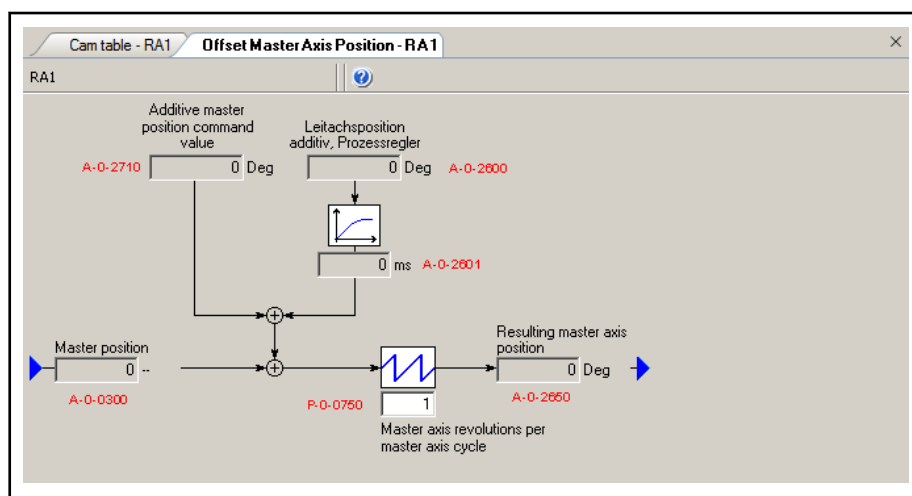


Fig. 8-78: "Offset Master Axis Position" dialog in the "Cam table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value	A-0-2710						
Master axis, actual position value	A-0-0300						
Additive master axis position, process controller	A-0-2600						
ELS, filter time constant, additive master axis position, process controller	A-0-2601						
ELS, resulting master axis position	A-0-2650						
Master axis revolutions per master axis cycle	P-0-0750						

Gear Settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

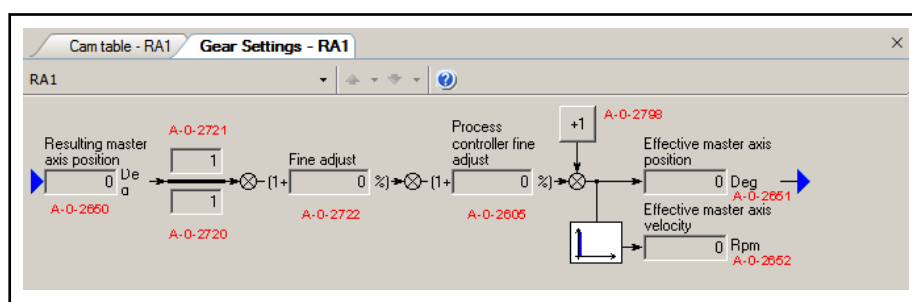


Fig. 8-79: "Gear Settings" dialog in the "Cam table" dialog

Axes

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, resulting master axis position	A-0-2650						
ELS, master axis gear, output revolutions	A-0-2721						
ELS, master axis gear, input revolutions	A-0-2720						
ELS, gear ratio fine adjustment	A-0-2722						
ELS, gear ratio fine adjustment, process controller	A-0-2605						
Polarity of master axis position	A-0-2798						
ELS, effective master axis position	A-0-2651						
ELS, effective master axis velocity	A-0-2652						

Cam Table Phase Offset

Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller".

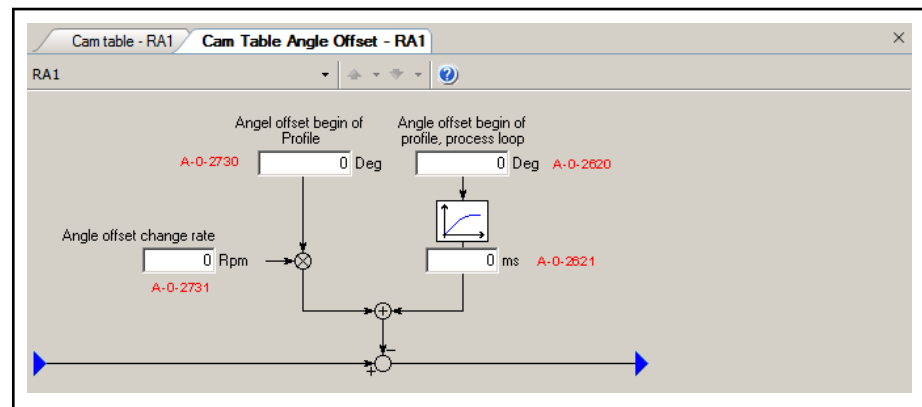


Fig. 8-80: "Cam Table Phase Offset" in the "Cam table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, PhasingTableAngle, cam table beginning	A-0-2730						
ELS, PhasingTableAngle, change velocity	A-0-2731						
ELS, phase offset of table beginning, process controller	A-0-2620						
ELS, filter time constant, phase offset of table, process controller	A-0-2621						

Dynamic Phase Offset

This functionality is currently not supported by the IndraMotion MLC.

Cam Table Selection

Selection and download of a cam table created with the CamBuilder to a cam table of the drive. It is possible to load a different cam table in any of the eight cam tables. The selection of the cam table to be processed is carried out on the MC_CamIn Motion function block. Furthermore, for each cam table, either linear and cubic spline interpolation can be selected.

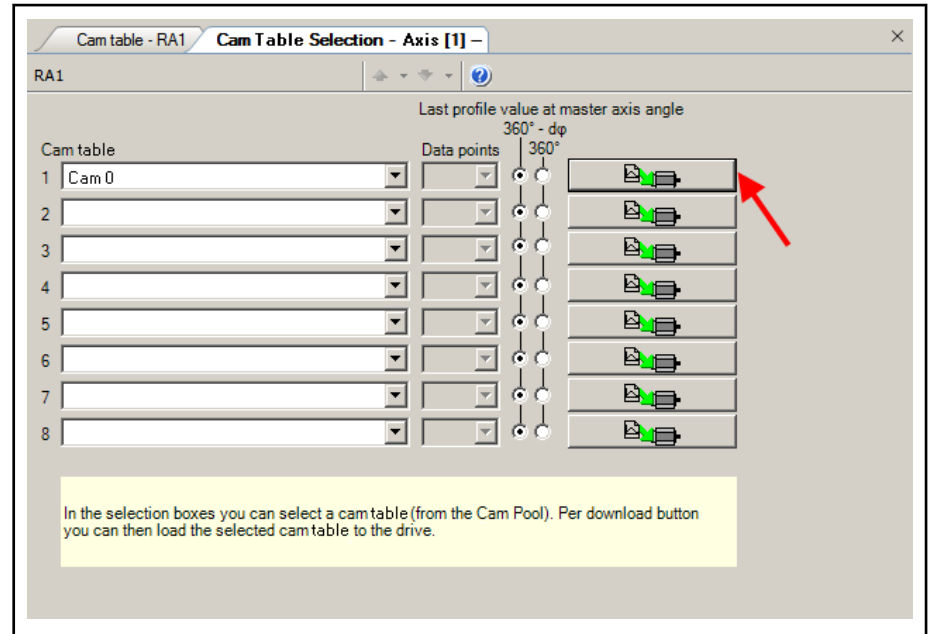


Fig. 8-81: "Cam Table Selection" dialog in the "Cam table" dialog

Dynamic Synchronization - Phase Synchronization

Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2761, ELS, PhasingSlaveVel, additive command velocity" and the "A-0-2610, ELS, additive position command value, process controller".

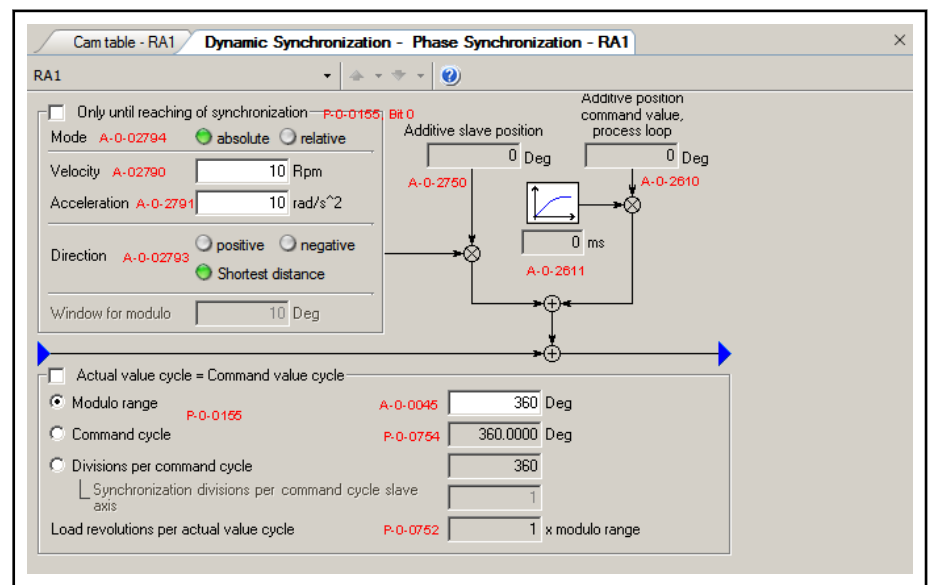


Fig. 8-82: "Dynamic Synchronization - Phase Synchronization" dialog in the "Cam table" dialog

Axes

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Synchronization mode	P-0-0155	A-0-2794					
ELS, synchronization mode (SercosDrive and virtual axes)	A-0-2794						
ELS, synchronization velocity	A-0-2790						
ELS, synchronization acceleration	A-0-2791						
ELS, synchronization direction	A-0-2793						
Scaling type for position data	A-0-0059						
Modulo value	A-0-0045						
Command value cycle	P-0-0754						
Load revolutions per actual value cycle of the slave axis	P-0-0752						
ELS, PhasingSlave, additive position command value	A-0-2750						
ELS, additive position command value, process controller	A-0-2610						
ELS, filter time constant, additive position command value	A-0-2611						

8.7.20 MotionProfile

General Information



This dialog is available for all axis types with interpolation on the control. These are: IndraDrive (IPO control), SercosDrive, virtual axis, controller axis.

If a real axis (IndraDrive) operates as slave in the "MB_MotionProfile" Motion function block, the drive runs in the operating state "MotionProfile with real/virtual master axis". In this operating state, there is a fixed relation between a given master axis position and a slave axis. Master axes can be real, virtual or encoder axes.

Based on the virtual master axis position (actual position value, master), the position command value is created for the real axis "Master Axis Position Offset", "Gear" and "Dynamic Synchronization".

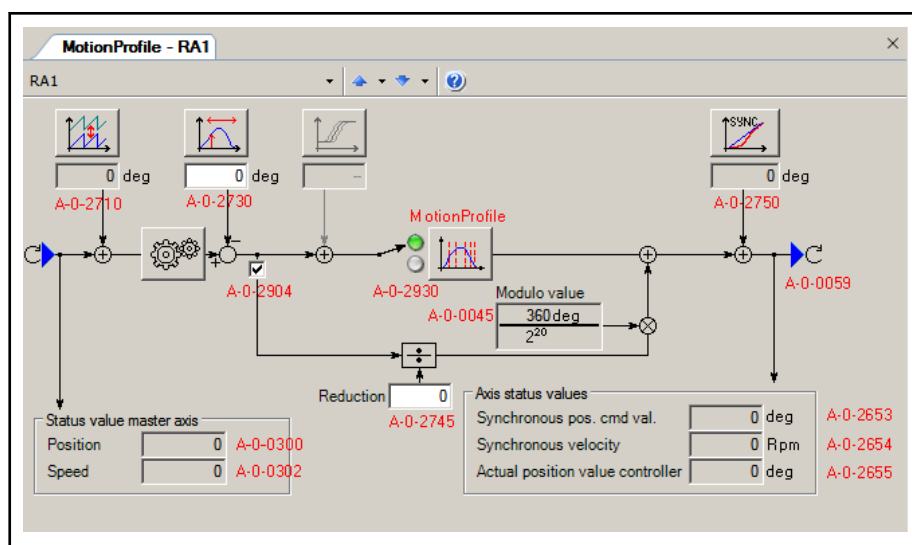


Fig.8-83: "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, master axis, axis number			A-0-2710				
ELS, Phasing, additive position command value			A-0-2710				
ELS, PhasingTableAngle, cam table beginning			A-0-2730				
ELS, synchronous position command value			A-0-2750				
ELS, synchronous position command value			A-0-2653				
ELS, synchronous velocity			A-0-2654				
ELS, position command value			A-0-2655				
Master axis, actual position value			A-0-0300				
Master axis, actual velocity value			A-0-0302				
MotionProfile, basic configuration			A-0-2904				
Scaling type for position data			A-0-0059				

The MotionProfile dialog displays the relevant parameters when the control is in the operating state "MotionProfile with real/virtual master axis". The parameters can be modified by the settings made in the PLC program for the MB_MotionProfile Motion function block.

Master axis position offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value" as well as "A-0-2600, Additive master axis position, process controller".

Gear settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Axes

- Cam table - Phase offset** Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller".
- Dynamic phase offset** This functionality is currently not supported by the IndraMotion MLC.
- MotionProfile overview** Selection and download of a MotionProfile created with a CamBuilder on the real axis.
- Dynamic synchronization – Phase synchronization** Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2750, ELS, PhasingSlave, additive position command value" and the "A-0-2610, ELS, additive position command value, process controller".

Master Axis Position Offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value " as well as "A-0-2600, Additive master axis position, process controller". With the "P-0-0750, Master axis revolutions per master axis cycle", the modulo range of the master axis position is specified.

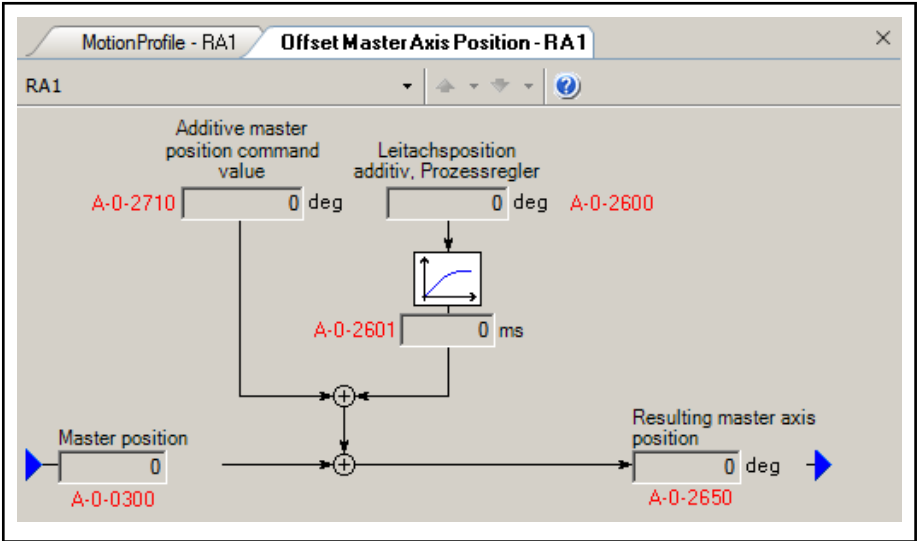


Fig.8-84: "Offset Master Axis Position" dialog in the "MotionProfile" dialog
Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value			A-0-2710				
ELS, filter time constant, additive master axis position, process controller			A-0-2601				
ELS, resulting master axis position			A-0-2650				
Master axis, actual position value			A-0-0300				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Additive master axis position, process controller		A-0-2600					
Master axis revolutions per master axis cycle		P-0-0750					

Gear Settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

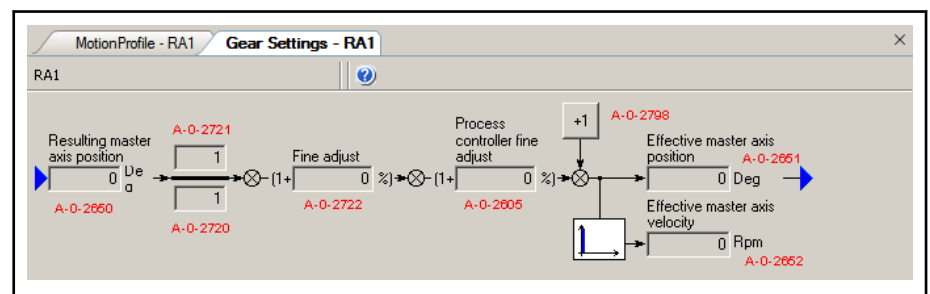


Fig. 8-85: "Gear Settings" dialog in the "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, resulting master axis position		A-0-2650					
ELS, master axis gear, input revolutions		A-0-2720					
ELS, master axis gear, output revolutions		A-0-2721					
ELS, gear ratio fine adjustment		A-0-2722					
ELS, gear ratio fine adjustment, process controller		A-0-2605					
Polarity of master axis position		A-0-2798					
ELS, effective master axis position		A-0-2651					
ELS, effective master axis velocity		A-0-2652					

Cam Table - Phase Offset

Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller" and "A-0-2621, ELS, filter time constant, phase offset of table, process controller".

Axes

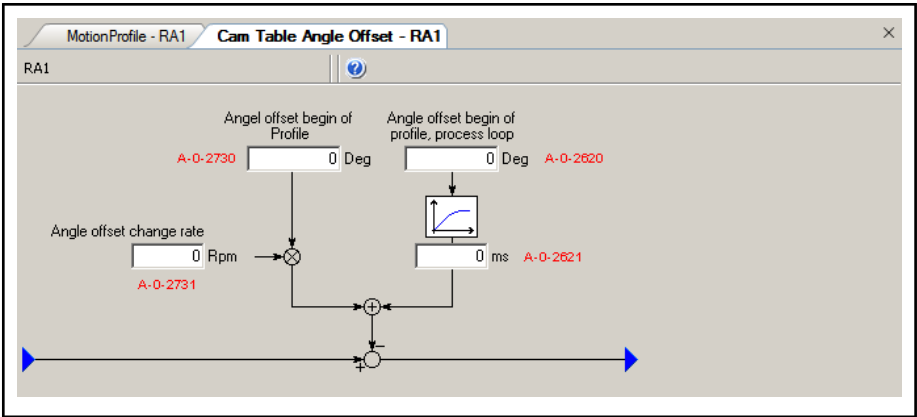


Fig.8-86: "Cam Table Phase Offset" dialog in the "MotionProfile" dialog
Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, PhasingTableAngle, cam table begin- ning		A-0-2730					
ELS, PhasingTableAngle, change velocity		A-0-2731					
ELS, phase offset of table beginning, proc- ess controller		A-0-2620					
ELS, filter time constant, phase offset of ta- ble, process controller		A-0-2621					

Dynamic Phase Offset

This functionality is currently not supported by the IndraMotion MLC.

MotionProfile Overview

Selection and download of a MotionProfile created with CamBuilder to the axis. For each axis, a maximum of two MotionProfiles can be stored. With the dialog, MotionProfiles already stored on the axis can return to the CamBuilder. The MotionProfile to be processed is selected in the MB_MotionProfile Motion function block.

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
MotionProfile, master axis switching position		A-0-2900					
MotionProfile, motion step 1, initial slave axis position		A-0-2901					
MotionProfile, control word		A-0-2930					
MotionProfile, status word		A-0-2931					
MotionProfile 0		x					
Number of motion steps		A-0-2910					
Master axis velocity		A-0-2911					
List of initial master axis positions		A-0-2912					
List of motion laws		A-0-2913					
List of distances		A-0-2914					
List of slave axis velocities		A-0-2915					
MotionProfile 2		x					

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Number of motion steps			A-0-2920				
Master axis velocity			A-0-2921				
List of initial master axis positions			A-0-2922				
List of motion laws			A-0-2923				
List of distances			A-0-2924				
List of slave axis velocities			A-0-2925				

Dynamic Synchronization - Phase Synchronization

Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2750, ELS, PhasingSlave, additive position command value" and the "A-0-2610, ELS, additive position command value, process controller".

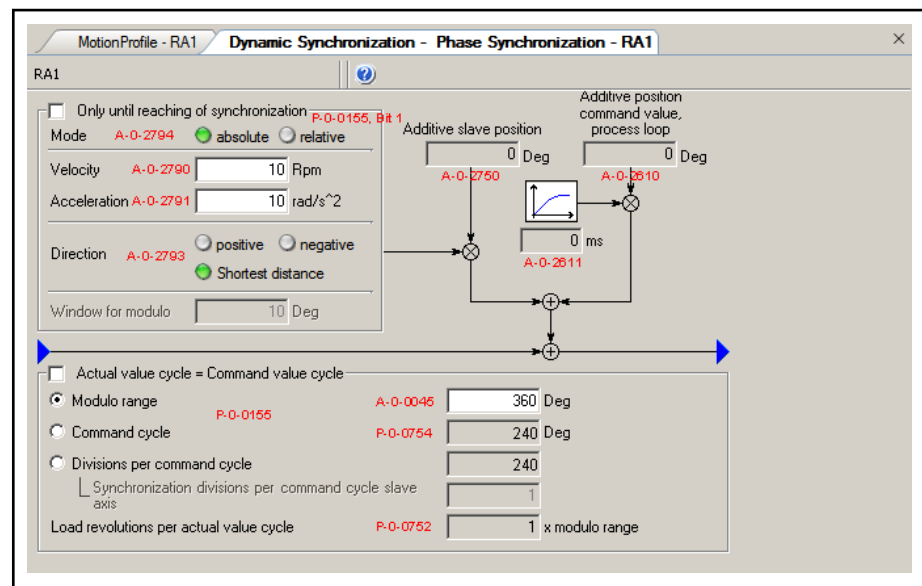


Fig. 8-88: "Dynamic Synchronization – Phase Synchronization" dialog in the "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Synchronization mode		P-0-0155					
Synchronization mode			A-0-2794				
ELS, synchronization velocity			A-0-2790				
ELS, synchronization acceleration			A-0-2791				
ELS, synchronization direction			A-0-2793				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, PhasingSlave, additive position command value		A-0-2750					
ELS, additive position command value, process controller		A-0-2610					
ELS, filter time constant, additive position command value		A-0-2611					
Modulo value		A-0-0045					
Command value cycle		P-0-0754					
Load revolutions per actual value cycle of the slave axis		P-0-0752					

8.7.21 FlexProfile

General Information



This dialog is available for all axis types with interpolation on the control. These are: IndraDrive (IPO control), SercosDrive, virtual axis, controller axis.

In addition to the MotionProfile, the "FlexProfile" is available on the control for 04VRS and higher. Apart from a series of new motion laws, different switching and synchronization options as well as links to events are provided. In addition to the master axis as position master, the time master is also available. Up to four of these new FlexProfiles can be created per axis and thus allow a faster switching to the prepared profiles.

The basic use of FlexProfile is described in /23/, [Required and Supplementing Documentations, page 14](#).

If a real axis (IndraDrive) operates as slave in the MB_FlexProfile Motion function block, the drive runs in the operating state "FlexProfile". In this operating state, there is a fixed relation between a given master axis position and a slave axis. Master axes can be real, virtual or encoder axes.

Based on the master axis position (actual position value, master), the position command value is created for the real axis "Master Axis Position Offset", "Gear" and "Dynamic Synchronization".

Axes

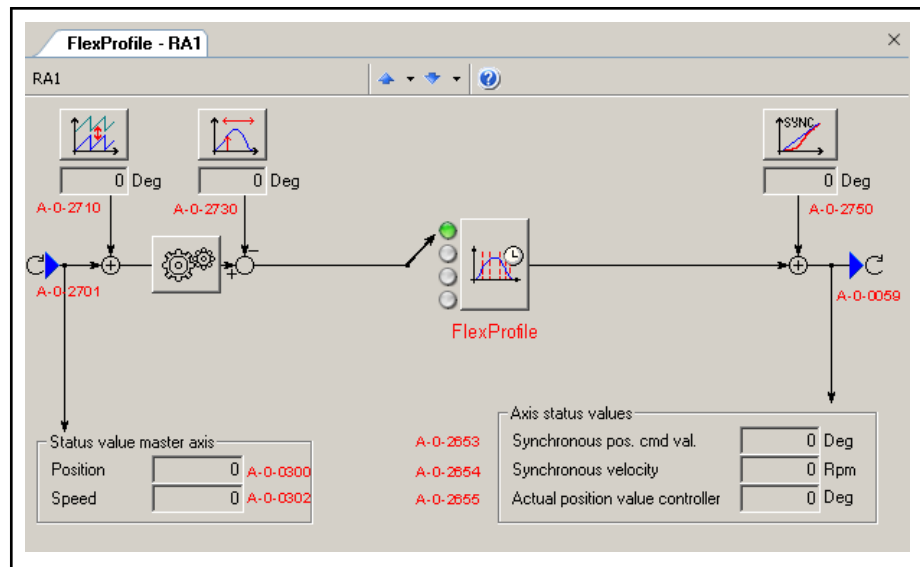


Fig.8-89: "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Scaling type for position data			A-0-0059				
ELS, master axis, axis number			A-0-2710				
ELS, Phasing, additive position command value			A-0-2710				
ELS, PhasingTableAngle, cam table beginning			A-0-2730				
ELS, synchronous position command value			A-0-2750				
ELS, synchronous position command value			A-0-2653				
ELS, synchronous velocity			A-0-2654				
ELS, position command value			A-0-2655				
Master axis, actual position value			A-0-0300				
Master axis, actual velocity value			A-0-0302				

The FlexProfile dialog displays the relevant parameters when the control is in the operating state "FlexProfile". The parameters can be modified by the settings made in the PLC program for the ML_FlexProfile Motion function block.

Master axis position offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value" as well as "A-0-2600, Additive master axis position, process controller".

Gear settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Cam table - Phase offset

Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change ve-

FlexProfile overview

Dynamic synchronization – Phase synchronization

locity" as well as "A-0-2620, ELS, phase offset start of table, process controller".

Selection and download of a FlexProfile to a real axis.

Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2750, ELS, PhasingSlave, additive position command value" and the "A-0-2610, ELS, additive position command value, process controller".

Master Axis Position Offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value" as well as "A-0-2600, Additive master axis position, process controller". With the "P-0-0750, Master axis revolutions per master axis cycle", the modulo range of the master axis position is specified.

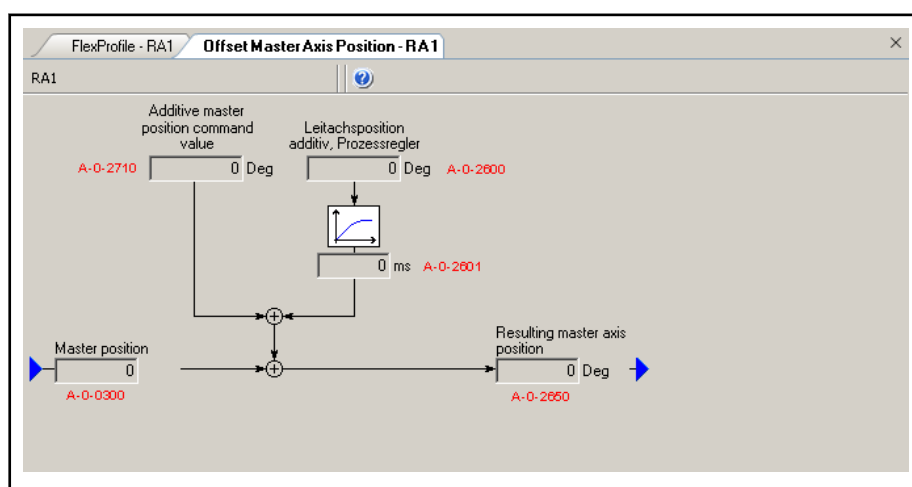


Fig. 8-90: "Offset Master Axis Position" in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value		A-0-2710					
Additive master axis position, process controller		A-0-2600					
ELS, filter time constant, additive master axis position, process controller		A-0-2601					
ELS, resulting master axis position		A-0-2650					
Master axis, actual position value		A-0-0300					

Gear Settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Axes

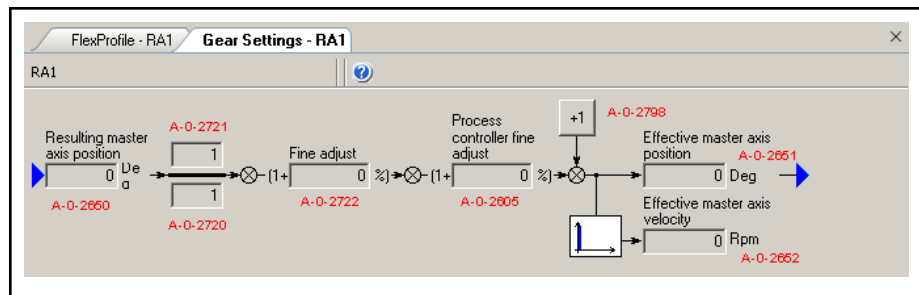


Fig. 8-91: "Gear Settings" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, resulting master axis position			A-0-2650				
ELS, master axis gear, input revolutions			A-0-2720				
ELS, master axis gear, output revolutions			A-0-2721				
ELS, gear ratio fine adjustment			A-0-2722				
ELS, gear ratio fine adjustment, process controller			A-0-2605				
Polarity of master axis position			A-0-2798				
ELS, effective master axis position			A-0-2651				
ELS, effective master axis velocity			A-0-2652				

Cam Table Phase Offset

Definition of the phase offset with "A-0-2730, ELS, PhasingTableAngle, cam table beginning" and "A-0-2731, ELS, PhasingTableAngle, offset change velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller" and "A-0-2621, ELS, filter time constant, phase offset of table, process controller".

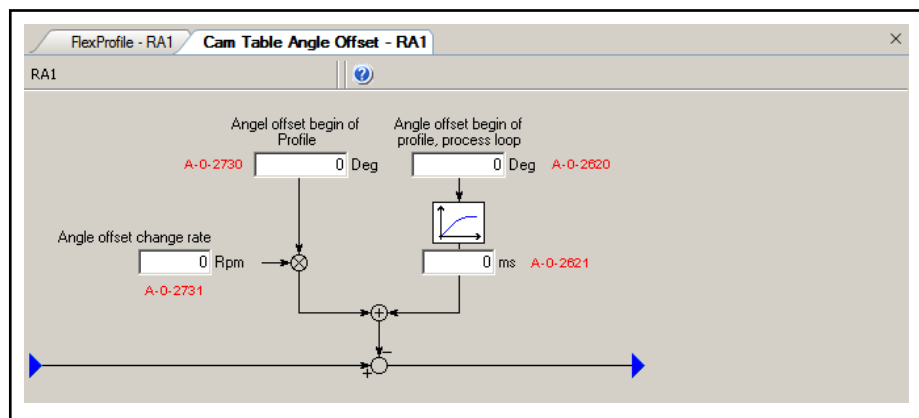


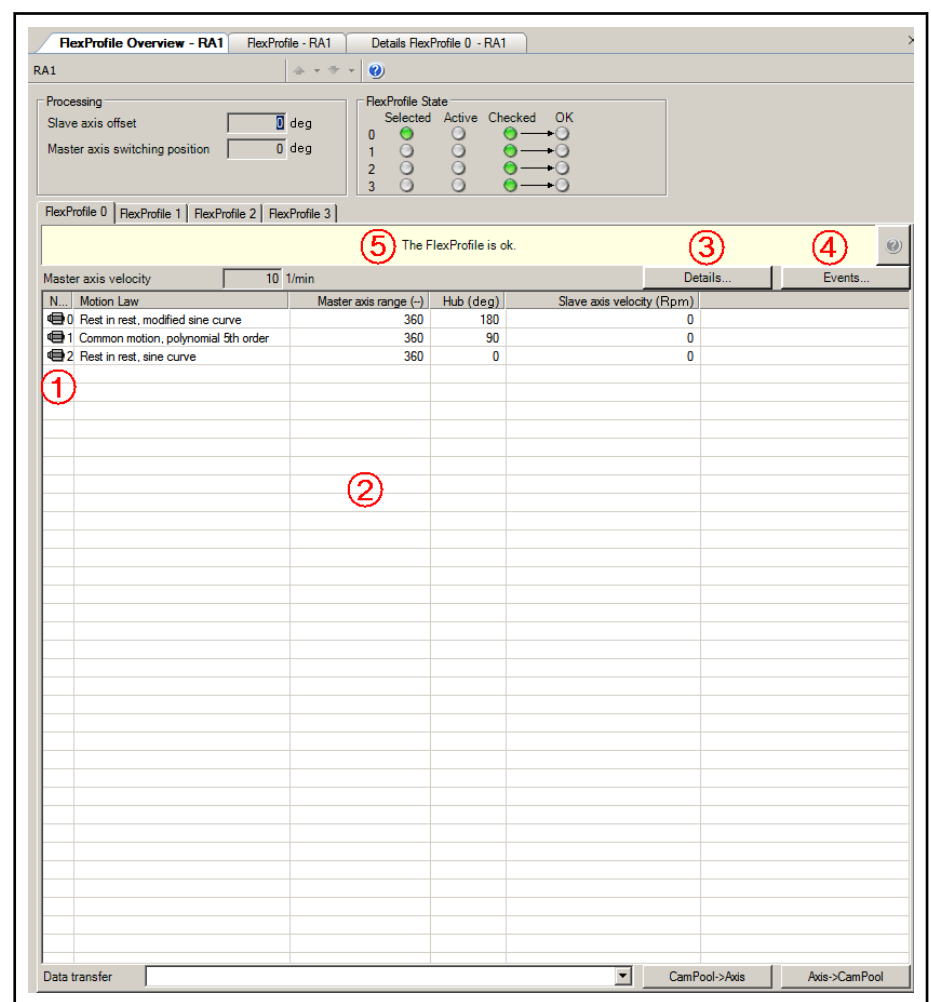
Fig. 8-92: "Cam Table - Phase Offset" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, PhasingTableAngle, cam table beginning		A-0-2730					
ELS, PhasingTableAngle, change velocity		A-0-2731					
ELS, phase offset of table beginning, process controller		A-0-2620					
ELS, filter time constant, phase offset of table, process controller		A-0-2621					

FlexProfile Overview

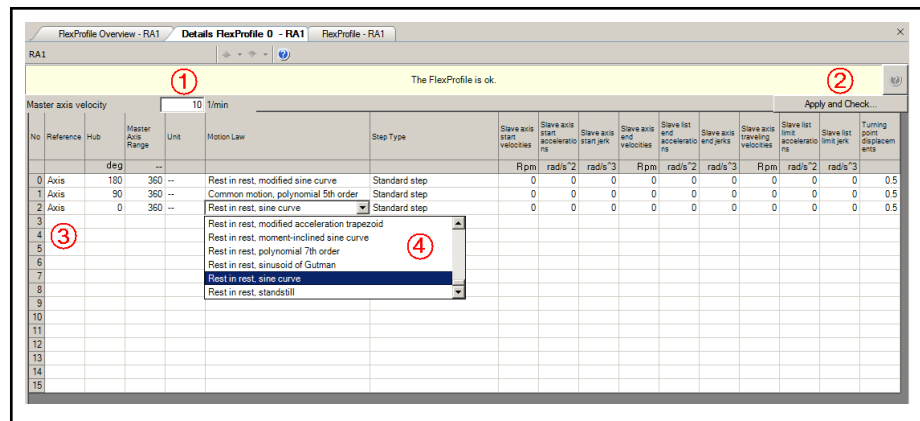
Selection and download of a FlexProfile to the axis. For each axis, up to four FlexProfiles can be stored. The FlexProfile to be processed is selected in the ML_FlexProfile Motion function block.



- ① Symbolic representation of the segment type (axis or time)
- ② Configuration display
- ③ FlexProfile configurator
- ④ Event configurator
- ⑤ Status display

Fig. 8-93: "FlexProfile Overview" dialog in the "FlexProfile" dialog

Axes



- ① Master axis velocity
- ② Apply to axis and check configuration
- ③ Master axis source selection
- ④ Motion laws can be selected from a list

Fig.8-94: Configuration of FlexProfile

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
FlexProfile, master axis switching position			A-0-3002				
FlexProfile, slave axis, additive position command value			A-0-3003				
FlexProfile 0			x				
List of master axis sources			A-0-3021				
List of slave axis distances			A-0-3022				
List of master axis ranges			A-0-3023				
List of motion laws			A-0-3025				
List of slave axis end velocities			A-0-3030				
FlexProfile 1			x				
List of master axis sources			A-0-3051				
List of slave axis distances			A-0-3052				
List of master axis ranges			A-0-3053				
List of motion laws			A-0-3055				
List of slave axis end velocities			A-0-3060				
FlexProfile 2			x				
List of master axis sources			A-0-3081				
List of slave axis distances			A-0-3082				
List of master axis ranges			A-0-3083				
List of motion laws			A-0-3085				
List of slave axis end velocities			A-0-3090				
FlexProfile 3			x				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
List of master axis sources			A-0-3111				
List of slave axis distances			A-0-3112				
List of master axis ranges			A-0-3113				
List of motion laws			A-0-3115				
List of slave axis end velocities			A-0-3120				

Dynamic Synchronization - Phase Synchronization

Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2750, ELS, PhasingSlave, additive position command value" and the "A-0-2610, ELS, additive position command value, process controller".

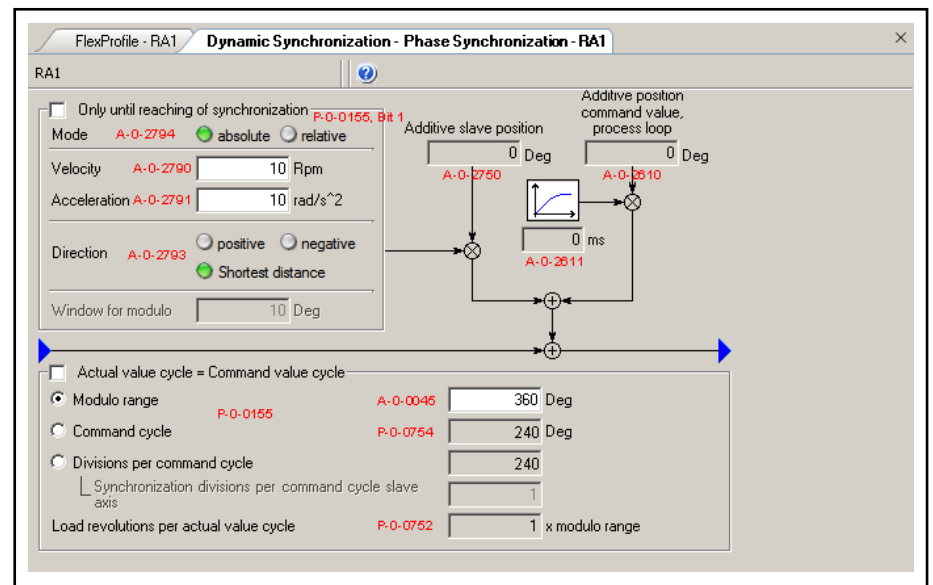


Fig. 8-95: "Dynamic Synchronization – Phase Synchronization" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Synchronization mode		P-0-0155					
Synchronization mode			A-0-2794				
ELS, synchronization velocity			A-0-2790				
ELS, synchronization acceleration			A-0-2791				
ELS, synchronization direction			A-0-2793				
ELS, PhasingSlave, additive position command value			A-0-2750				

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, additive position command value, process controller			A-0-2610				
ELS, filter time constant, additive position command value			A-0-2611				
Modulo value			A-0-0045				
Command value cycle		P-0-0754					
Load revolutions per actual value cycle of the slave axis		P-0-0752					

8.7.22 Gear with Velocity Synchronization

General Information



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive, virtual axis, controller axis.

If a real axis operates as a slave axis in the MC_GearIn Motion function block, the drive runs in the operating state "Velocity synchronization with real/virtual master axis". In this operating state, the drive follows a given master axis velocity velocity-synchronously. Master axes can be real, virtual or encoder axes.

Based on the master axis velocity "A-0-0302, Master axis, actual velocity value", the velocity command value "A-0-2657, Effective velocity command value" is created for the real axis using "Offset Master Axis Position", "Gear" and "Dynamic Synchronization".

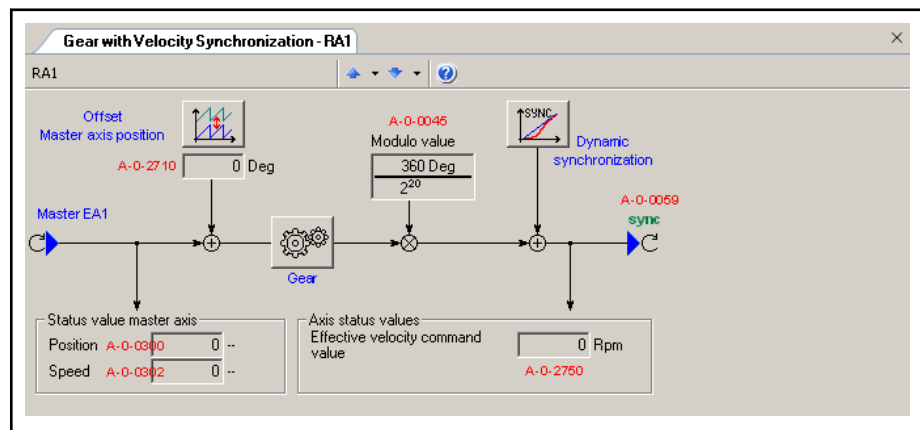


Fig. 8-96: "Gear with Velocity Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value			A-0-2710				
Master axis, actual position value			A-0-0300				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Scaling type for position data	A-0-0059						
Modulo value	A-0-0045						
Master axis, actual velocity value	A-0-0302						
ELS, effective master axis velocity	A-0-2657						

The "Gear with Velocity Synchronization" displays the effective parameters when the drive is in the operating state "Velocity synchronization with real/virtual master axis". The parameters can be modified by the settings made in the PLC program for the MB_GearIn Motion function block.

Master axis position offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value " as well as "A-0-2600, Additive master axis position, process controller".

Gear settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Dynamic synchronization

Definition of the "A-0-2791, ELS, synchronization acceleration", the "A-0-2761, ELS, PhasingSlaveVel, additive command velocity" and the "A-0-2615, ELS, additive velocity command value, process controller".

Application

The velocity synchronization is used with simple feed rolls of printing presses for example. The drive runs synchronously to the master axis velocity. The web velocity at the perimeter of the feed roll or of a winding support is preset by the electric gear. A defined tension can be set via the gear fine adjustment.

Master Axis Position Offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value " as well as "A-0-2600, Additive master axis position, process controller".

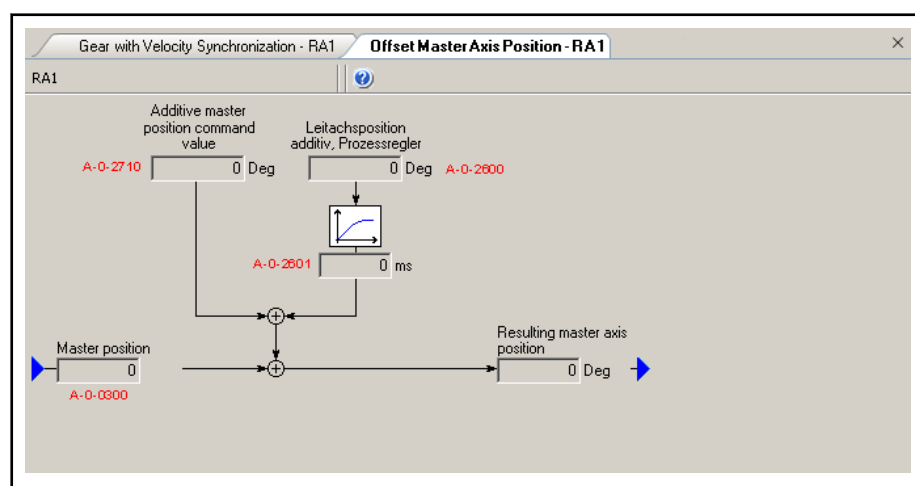


Fig. 8-97: "Offset Master Axis Position" subdialog in the "Gear with Velocity Synchronization" dialog

Parameters used:

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Master axis, actual position value	A-0-0300						
ELS, Phasing, additive position command value	A-0-2710						
Additive master axis position, process controller	A-0-2600						
ELS, filter time constant, additive master axis position, process controller	A-0-2601						
ELS, resulting master axis position	A-0-2650						
Modulo value	A-0-0045						

Gear Settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as "A-0-2722, ELS, Gear ratio fine adjustment" and determination of the "A-0-2652, ELS, effective master axis velocity".

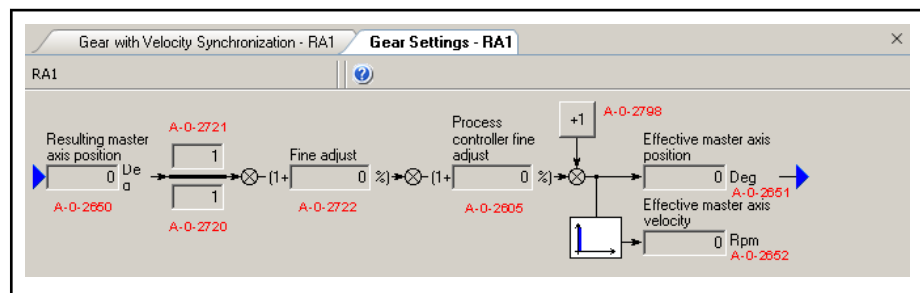


Fig.8-98: "Gear Settings" dialog in the "Gear with Velocity Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, resulting master axis position	A-0-2650						
ELS, master axis gear, input revolutions	A-0-2720						
ELS, master axis gear, output revolutions	A-0-2721						
ELS, gear ratio fine adjustment	A-0-2722						
ELS, gear ratio fine adjustment, process controller	A-0-2605						
Polarity of master axis position	A-0-2798						
Effective master axis position	A-0-2651						
Effective master axis velocity	A-0-2652						

Dynamic Synchronization

Definition of the "A-0-2791, ELS, synchronization acceleration", the "A-0-2761, ELS, PhasingSlaveVel, additive command velocity" and the "A-0-2615, ELS, additive velocity command value, process controller"

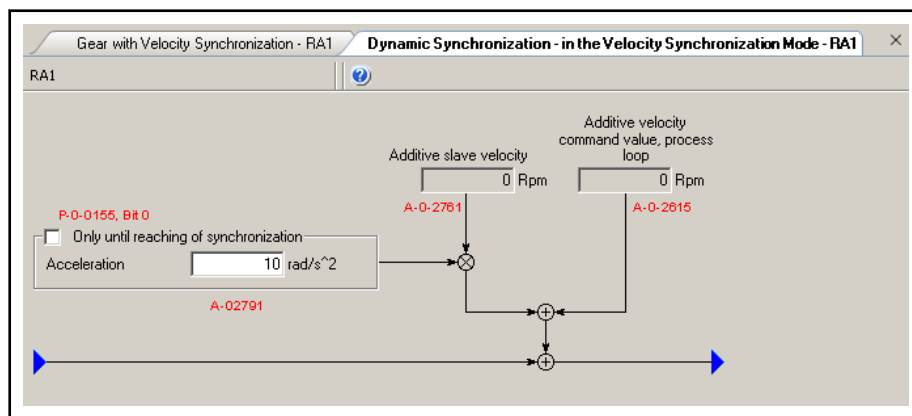


Fig. 8-99: "Dynamic Synchronization in the Velocity Synchronization Mode" dialog in the "Gear with Velocity Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Synchronization mode	P-0-0155						
ELS, synchronization acceleration	A-0-2791						
ELS, PhasingSlaveVel, additive position command value	A-0-2761						
ELS, additive velocity command value, proc- ess controller	A-0-2615						

8.7.23 Gear with Phase Synchronization

General Information



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive, virtual axis, controller axis.

If a real axis operates as slave axis in the MC_GearInPos Motion function block, the drive runs in the operating state "Phase synchronization with real/virtual master axis". In this operating state, the drive follows a given master axis position either absolutely or relative phase-synchronously. Master axes can be real, virtual or encoder axes.

Based on the master axis position "A-0-0300, Master axis, actual position value", the effective position command value "A-0-2655, ELS position command value" is generated for the real axis with "Master Axis Position Offset", "Gear" and "Dynamic Synchronization".

Axes

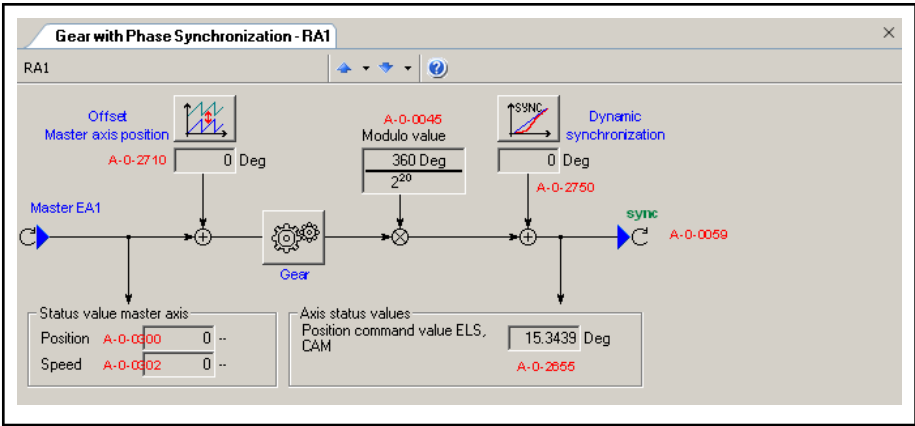


Fig.8-100: "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value		A-0-2710					
Master axis, actual position value		A-0-0300					
Scaling type for position data		A-0-0059					
Modulo value		A-0-0045					
ELS, position command value		A-0-2655					
ELS, PhasingSlave, additive position command value		A-0-2750					
Master axis, actual velocity value		A-0-0302					

The "Gear with Phase Synchronization" displays the relevant parameters when the drive is in the operating state "Phase synchronization with real/virtual master axis".

The parameters can be modified by the settings made in the PLC program for the MB_GearInPos Motion function block.

Master axis position offset	Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value " as well as "A-0-2600, Additive master axis position, process controller".
Gear settings	Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as" A-0-2722, ELS, Gear ratio fine adjustment" and determination of the "A-0-2652, ELS, effective master axis velocity".
Dynamic synchronization	Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2761, ELS, PhasingSlave Vel, additive command velocity" and the "A-0-2615, ELS, additive velocity command value, process controller".

Master Axis Position Offset

Determination of the resulting master axis position based on "A-0-0300, Master axis, actual position value" and "A-0-2710, ELS, phasing, additive position command value" as well as "A-0-2600, Additive master axis position, process controller". With the "P-0-0750, Number of master axis revolutions per master axis cycle", the modulo range of the master axis position is specified.

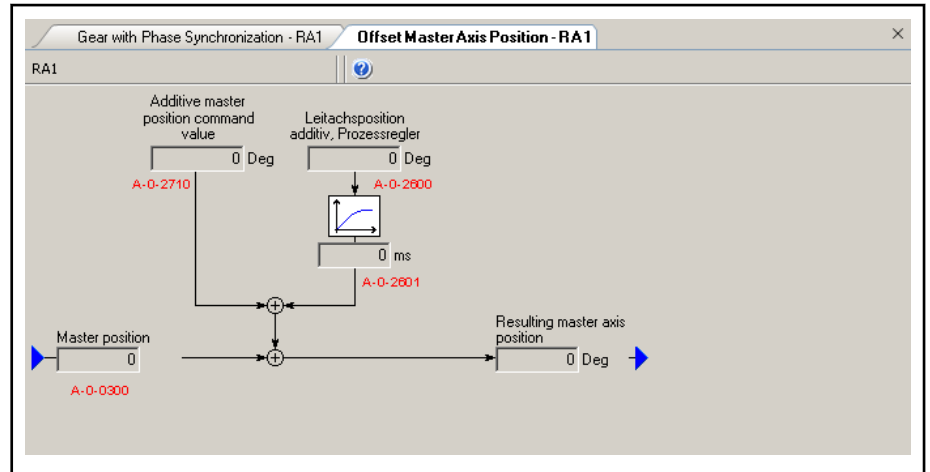


Fig. 8-101: "Offset Master Axis Position" dialog in the "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, Phasing, additive position command value		A-0-2710					
Master axis, actual position value		A-0-0300					
ELS, Phasing, additive position command value		A-0-2710					
Additive master axis position, process controller		A-0-2600					
ELS, filter time constant, additive master axis position, process controller		A-0-2601					
ELS, resulting master axis position		A-0-2650					

Gear Settings

Definition of the gear transmission ratio of the master axis gear with "A-0-2720, ELS, master axis gear, input revolutions" and "A-0-2721, ELS, master axis gear, output revolutions" as well as gear fine adjustment and determination of "A-0-2652, ELS, effective master axis velocity".

Axes

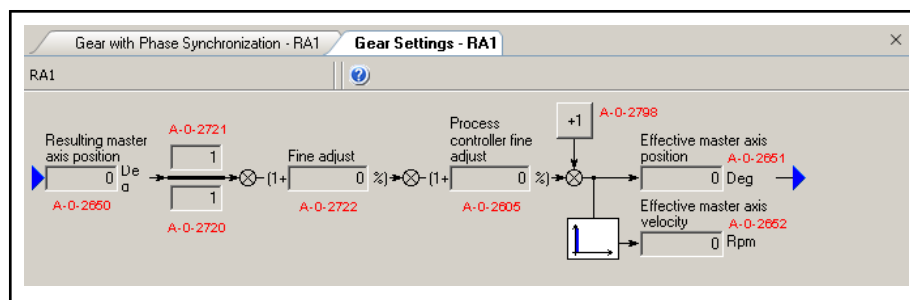


Fig.8-102: "Gear Settings" dialog in the "Gear with phase synchronization" dialog
Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
ELS, resulting master axis position		A-0-2650					
Polarity of master axis position		A-0-2798					
ELS, effective master axis position		A-0-2651					
ELS, master axis gear, input revolutions		A-0-2720					
ELS, master axis gear, output revolutions		A-0-2721					
ELS, gear ratio fine adjustment		A-0-2722					
ELS, gear ratio fine adjustment, process controller		A-0-2605					
ELS, effective master axis velocity		A-0-2652					

Dynamic Synchronization

Definition of the "A-0-2794, ELS, synchronization mode", the "A-0-2790, ELS, synchronization velocity", the "A-0-2791, ELS, synchronization acceleration" as well as the "A-0-2793, ELS, synchronization direction", the "A-0-2750, ELS, PhasingSlave, additive position command value" and the "A-0-2610, ELS, additive position command value, process controller".

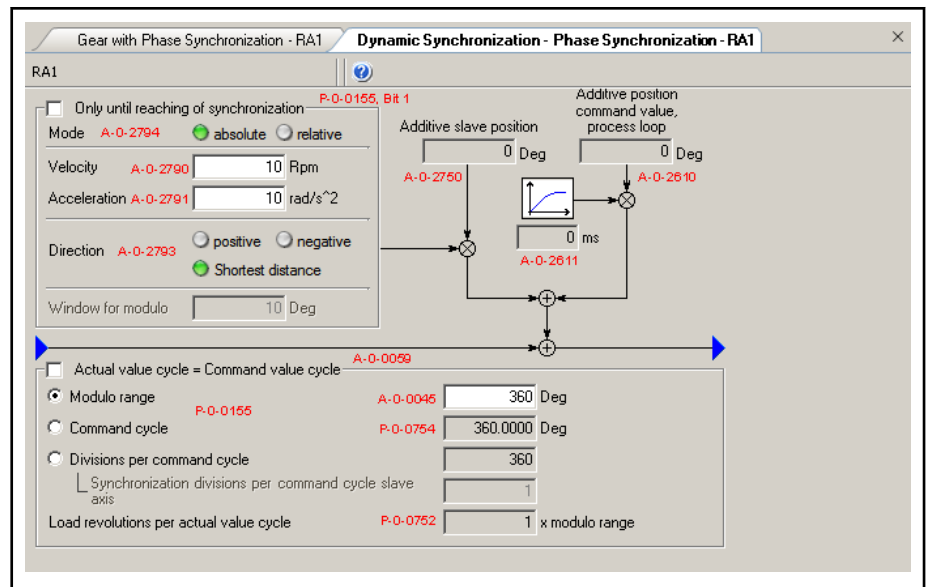


Fig. 8-103: Dynamic synchronization in the "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Synchronization mode	P-0-0155						
ELS, synchronization mode	A-0-2794						
ELS, synchronization velocity	A-0-2790						
ELS, synchronization acceleration	A-0-2791						
ELS, synchronization direction	A-0-2793						
Scaling type for position data	A-0-0059						
Modulo value	A-0-0045						
Command value cycle	P-0-0754						
Load revolutions per actual value of the slave axis	P-0-0752						
ELS, PhasingSlave, additive position command value	A-0-2750						
ELS, additive position command value, process controller	A-0-2610						
ELS, filter time constant, additive position command value	A-0-2611						

8.7.24 Axis Configuration



This dialog is available for encoder and link axes.

The encoder axis and link axis are configured via the following dialog.

Axes

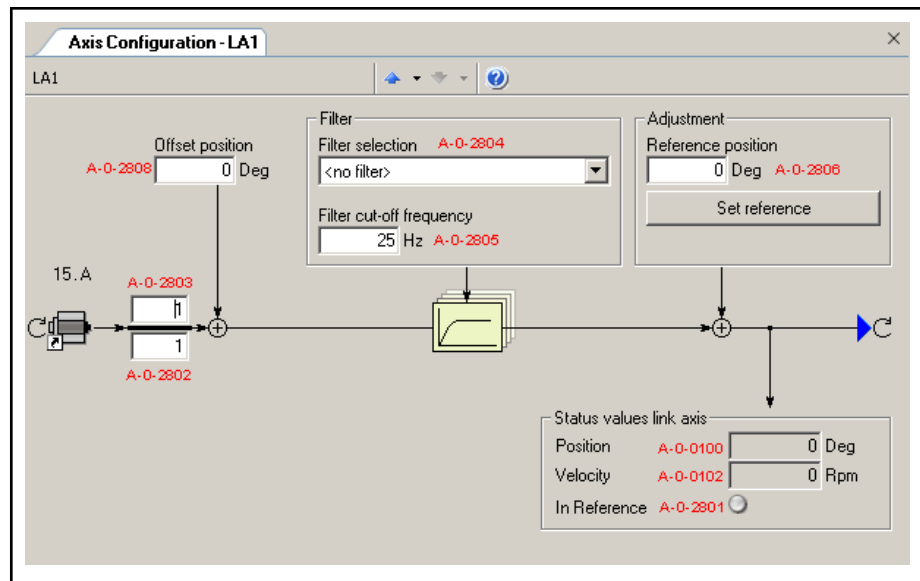


Fig.8-104: Dialog "Axis Configuration"

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Master axis encoder						x	
Configuration						A-0-2800	
Status						A-0-2801	
Input revolutions						A-0-2802	
Output revolutions						A-0-2803	
Filter type						A-0-2804	
Filter cutoff frequency						A-0-2805	
Reference position						A-0-2806	
Position command value, additive						A-0-2808	
Position/velocity						x	
Actual position value						A-0-0100	
Actual velocity value						A-0-0102	
Scaling type for position data						A-0-0059	
Actual velocity value of the measuring en- coder						P-0-0332	
Scaling exponent for torque/force data						P-0-0052	
Commands						x	
Setting Absolute Measuring (Position)						A-0-2807	

The transmission of the electronic measuring gear results from the ratio "A-0-2803, Master axis encoder, output revolutions" to "A-0-2802, Master axis encoder, input revolutions", i.e. RatioNumerator / RatioDenominator.

The adjustment range is between 65,535 : 1 and 1 : 65535.

The position value **after** the measuring gear is increased by "A-0-2808, Master axis encoder, additive position command value".

A filter - selectable and configurable by the user - is displayed. Via "A-0-2804, Master axis encoder - filter type", the following selection can be made:

Value	Filter type
0	Filter deactivated
1	40 ms average value filter
2	1st order low-pass filter
3	2nd order low-pass filter
4	2nd order Butterworth
5	2nd order low-pass filter with velocity feedforward (dead time compensation)
6	3rd order low-pass filter
7	3rd order low-pass filter with velocity and acceleration feedforward (dead time compensation)
8	3rd order Butterworth

Fig. 8-105: Filter type specification

The parameter "A-0-2805, Master axis encoder, filter cutoff frequency" to store the filter cutoff frequency completes the component.

Finally, the position value can be set to a desired value by the reference command ("A-0-2806, Master axis encoder, reference position", "A-0-2807, Command: Set absolute position", "A-0-2801, Master axis encoder status").

Successful referencing is indicated in "A-0-2801, Master axis encoder, status (Bit 0 = 1)".

As with the other axis types (encoder axis, real axis and virtual axis), the actual values for the link axis can also be used from the following parameters:

- "A-0-0100, Actual position value"
- "A-0-0102, Actual velocity value"

8.7.25 Drive-integrated Safety Engineering



This dialog is available for IndraDrive for interpolation in the drive and in the control.

If the drive is provided with the "Safety engineering " option module, the respective functions in the drive can be configured via this dialog.

The hardware functions of the safety engineering is described in the manual "Rexroth IndraDrive Drive Controllers Control Sections CSB01, CSH01, CDB01" (see /35/, [Required and Supplementing Documentations](#), page 14).

8.7.26 Position Switching Point



This dialog is available for IndraDrive for interpolation in the drive and in the control.

This dialog configures a virtual position-dependent switch.

Axes

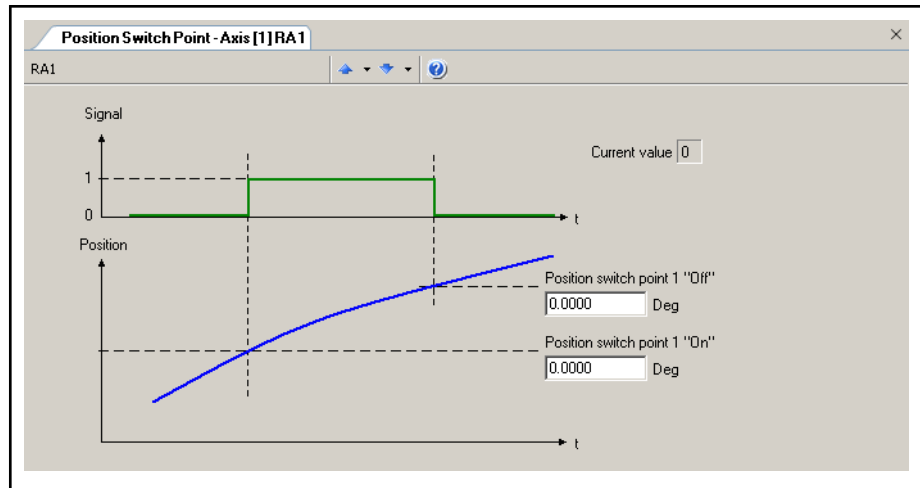


Fig.8-106: Position switching point

The following parameters are analyzed:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Position switching point 1 On	S-0-0060	
Position switching point 1 Off	S-0-0460	
Position switching point parameter	S-0-0059 (Bit 0)	

The switching positions for the drive are preset in "S-0-0060, Position switching point 1 "On"" and "S-0-0460, Position switching point 1 "Off"".

Between these positions, the virtual switch is in the state "1"; outside the two positions, the virtual switch is in the state "0".

The value can be read from the parameter "S-0-0059, Position switching point parameter".

The basic principles of the "Position switching points""Dynamic programmable limit switch" are described for individual products in /10/, /11/, [Required and Supplementing Documentations, page 14](#), section "Dynamic Programmable Limit Switch".

Support is provided by the implemented online help.

8.7.27 Touch Probe



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Touch probes measure absolute actual position values as well as actual position value differences and record time intervals between measuring signals.

Subject to the hardware design of the controller unit, up to two touch probe inputs can be defined per axis.

Measuring signals can either be actual position values of motor encoders, external encoders and measuring encoders or master axis position values or cam table values.

A measurement can be triggered by positive and/or negative touch probe signal edges.

Axes

Single shot measurements or continuous measurements are possible as well as counting the measuring events during the continuous measurement.

Furthermore, a position value range ("expectation window") can be defined for each touch probe, in which measurements can be executed, including the activation of a "Failure counter" when running through the expectation window without measuring event.

The sample time for touch probe signals is approx. 0.05 ms (for sufficient noise immunity, a signal edge change has to continue at least 4.0 ms in order to be detected!).



The function is only active in the operating mode (BB).

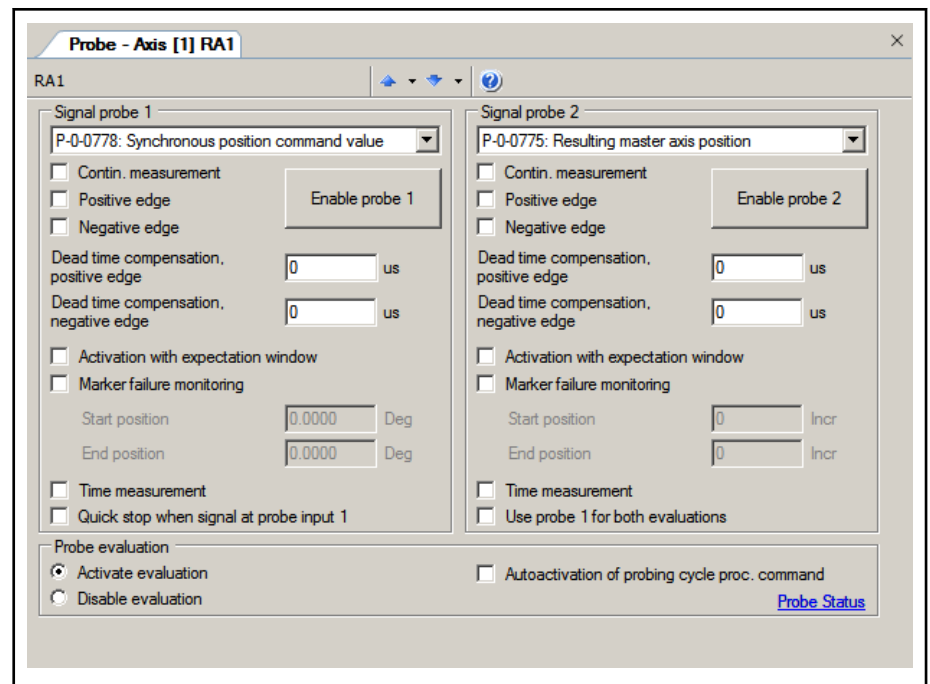


Fig. 8-107: Dialog "Touch Probe"

The following parameters are analyzed:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Touch probe, control parameter	S-0-0169	
Touch probe, extended control word	P-0-0226	
Command touch probe cycle	S-0-0170	
Touch probe 1	x	
Signal selection	S-0-0426	
Starting position active	P-0-0204	
End position active	P-0-0205	
Enabled	S-0-0405	
Touch probe 2	x	
Signal selection	S-0-0427	

Axes

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Starting position active		P-0-0200
End position active		P-0-0201
Enabled		S-0-0406

Working in the dialog starts and ends with enabling or disabling the touch probe evaluation.

For enabling, the parameter "S-0-0170, Command touch probe cycle"" is used to set the command in the drive and, at the same time, the command execution is enabled.

For disabling, the command in the drive is deleted and, at the same time, the command execution is interrupted (Fig. "Touch Probe" dialog, right bottom).



Only if the touch probe evaluation is activated, the remaining areas of the dialog are active.

Selecting signal sources

The signal sources are selected via a selection window. The selected sources are determined in the parameters "S-0-0426, Signal selection touch probe 1" or "S-0-0427, Signal selection touch probe 2".

Basic settings

The remaining basic settings, single shot measurement/continuous measurement, positive and/or negative edge, are stored in the parameters "S-0-0169, Touch probe control parameter" and "P-0-0226, Touch probe, extended control word" for both sources.

Activating the touch probe evaluation

If the touch probe evaluation (S-0-0170) is activated, the touch probes 1 and 2 can be enabled/disabled separately: "S-0-0405, Touch probe 1 enabled" or "S-0-0406, Touch probe 2 enabled".

The settings can be extended for one or the other touch probe:

- **Activating an expectation window:** The expectation window restricts the position range of an axis or shaft in which touch probe signal edges are used for the measured value acquisition.

Therefore, the starting and end positions of the respectively selected touch probe are stored in the parameters "P-0-0204, Starting position, touch probe function 1 enabled"/"P-0-0205, End position, touch probe function 1 enabled" or "P-0-0200, Starting position, touch probe function 2 enabled"/"P-0-0201, End position, touch probe function 2 enabled".

- **Marker failure monitoring:** Recording marker failures if expectation window is activated:

If the actual position value is outside the range of expected values, touch probe signal edges do not lead to a measured value acquisition!

If no "marker" was detected after a complete run (exceeding of both position limits) which triggered a measured value, this status can be stored in the parameter "P-0-0224, Touch probe, number of marker failures" and be indicated. If repeated, the value of (P-0-0224) increases incrementally. When registering a measured value trigger in the "expectation window", the value in the parameter (P-0-0224) is deleted.



The "expectation window" cannot be used for the time measurement!

The additional "Touch Probe Status" dialog that can be selected in the touch probe dialog, provides information on both touch probes.

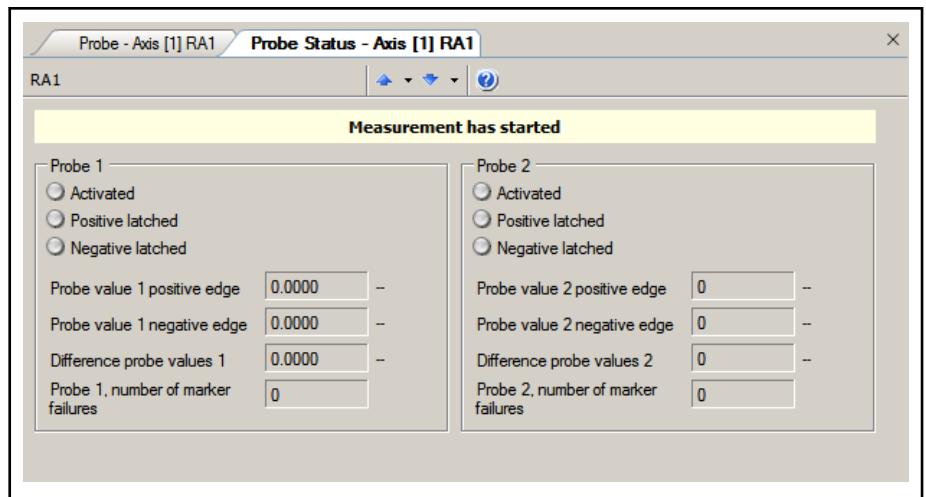


Fig. 8-108: "Touch Probe Status" dialog in the "Touch Probe" dialog

Further basic principles of the touch probe function are described product-specifically in /10/, /11/, [Required and Supplementing Documentations](#), page 14, section "Touch Probe Function".

Support is provided by the implemented online help.

8.7.28 Jogging



This dialog is available for all IndraDrive axes as well as virtual axes.

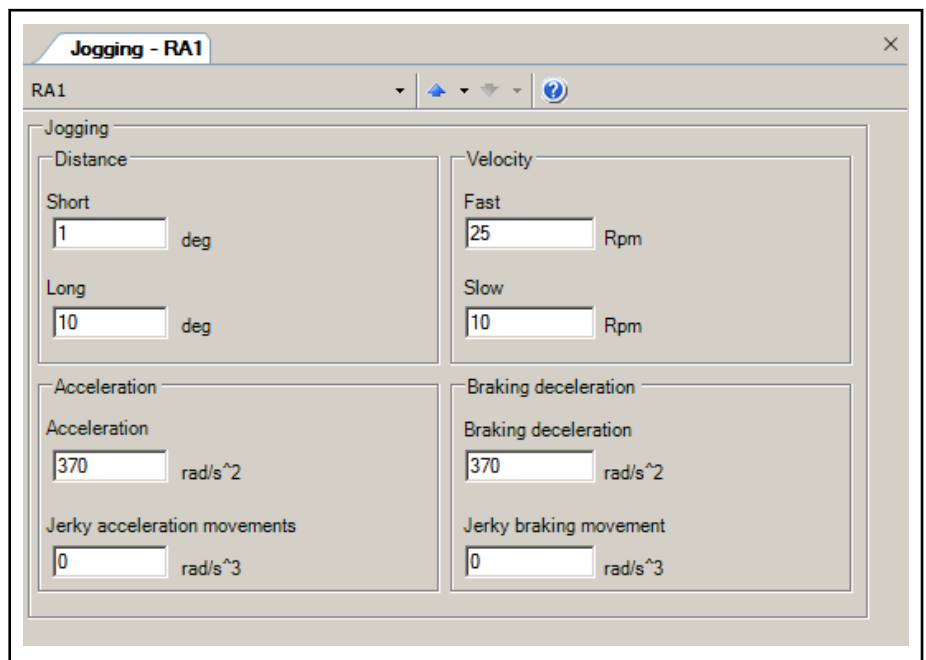


Fig. 8-109: "Jogging" dialog

This dialog is used to set the

- Velocity
- Acceleration
- Acceleration jerk
- Deceleration

Axes

of an axis for jogging (Manual mode).

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Jogging	x			x			
Short distance	A-0-0080			A-0-0080			
Long distance	A-0-0081			A-0-0081			
Low velocity	A-0-0082			A-0-0082			
High velocity	A-0-0083			A-0-0083			
Acceleration	A-0-0084			A-0-0084			
Deceleration	A-0-0085			A-0-0085			
Acceleration jerk	A-0-0086			A-0-0086			
Deceleration jerk	A-0-0087			A-0-0087			

The parameter units depend on the settings made for "Advanced Scaling / Units" in the dialog (see [Scaling/Units Extended, page 278](#)).

8.7.29 Initial Commissioning

For the initial commissioning of an axis, the following dialogs are processed by a wizard:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Loading Basic Parameters, page 334	x	x	x	x	x	x	x
Settings, Motor Encoder, page 274	x	x	x				
Settings of Optional Encoder, page 275	x	x	x			x	
Settings: Scaling/Units, page 277	x	x	x	x	x		x
Scaling/Units Extended, page 278	x	x	x				x
Mechanical Gear, page 280	x	x	x				
Brake, page 273	x	x	x				
Motion Limit Values, page 289	x	x	x		x		
Initial Values, page 291	x	x	x	x	x		
E-Stop Function, page 288	x	x	x				
Communication, page 336	x	x	x				
Settings of Status Messages, page 281	x	x	x	x	x		
Axis Configuration, page 321						x	



Data changed while editing in the dialog is directly transferred to the control/drive, irrespective of whether a subsequent date is still changed or whether the dialog is exited with **Next** or **Cancel**.

8.7.30 Exchanging the Drive/Firmware



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Introduction

Since it might become necessary to exchange a drive, a partially automated sequence is available for this procedure.

There are many reasons to exchange a drive, e.g.:

- Upgrade to a new drive of the same type with a newer firmware version
- Exchanging the drive for a new drive type
- Exchanging a defective device



The mechanism for exchanging the drive firmware depends on the type of drive and cannot be performed here. Using the drive commissioning tool is required here.

Depending on the status of control and drive (online/offline), there are three different methods:

1. The control and the drive to be exchanged are available online. The parameters to be backed up are saved online by the drive and the control in a parameter file. After an successful exchange, the control returns to "Online" state. This method is described below.
2. The configuration of the drive ring is available; the connection to the control is switched offline.
3. No drive ring configuration is available (no drive present); the connection to the control is switched offline or the control is not present.

In general, the sequence consists of three steps:

1. Backing up existing settings and parameters
2. Exchanging the drive (physically and in the control project)
3. Rewriting the saved settings and parameters

Selecting the drive to be exchanged

In the context menu of the desired real axis, select **Change Drive/Firmware....** A wizard opens to guide through the exchange procedure.



The following description corresponds to the first method described above. For the other two methods, fewer settings and dialogs can be provided.

Axes

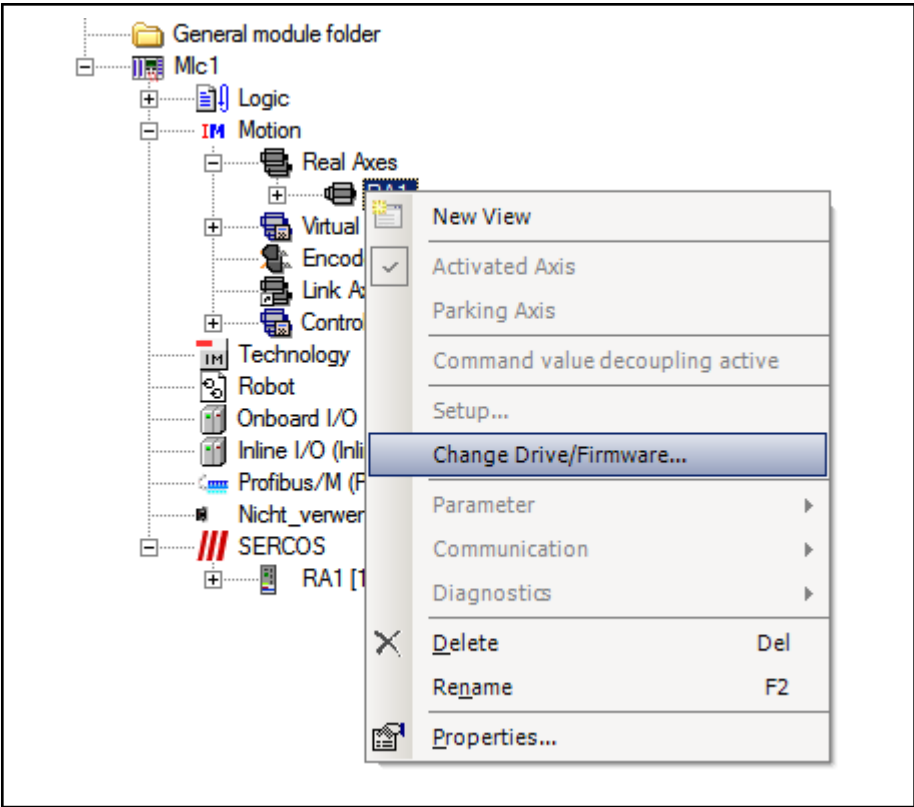


Fig.8-110: Selecting the drive to exchange using an example of a drive in an IndraMotion MLC project

Settings for parameter backup

In the first step, the parameters are backed up. Based on the availability of parameter data, this can be carried out with online or offline data. These settings can be carried out separately for axis data in the control (A-parameters) and the S-parameters and P-parameters in the drive. The parameters are saved internally during further processing. In addition, they can be backed up in a general parameter file (*.par) for later use.

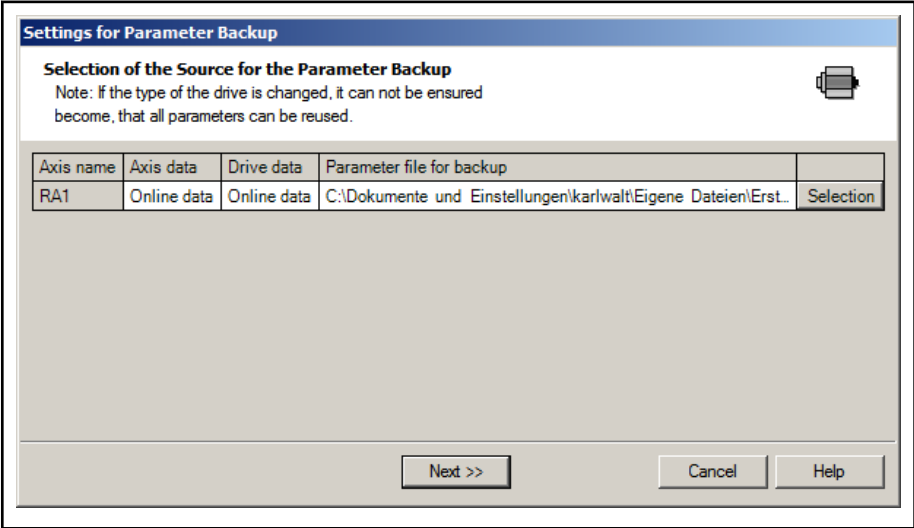


Fig.8-111: Settings for parameter backup

Parameter backup

The drive parameters A, S and P are saved in a backup file on the PC.

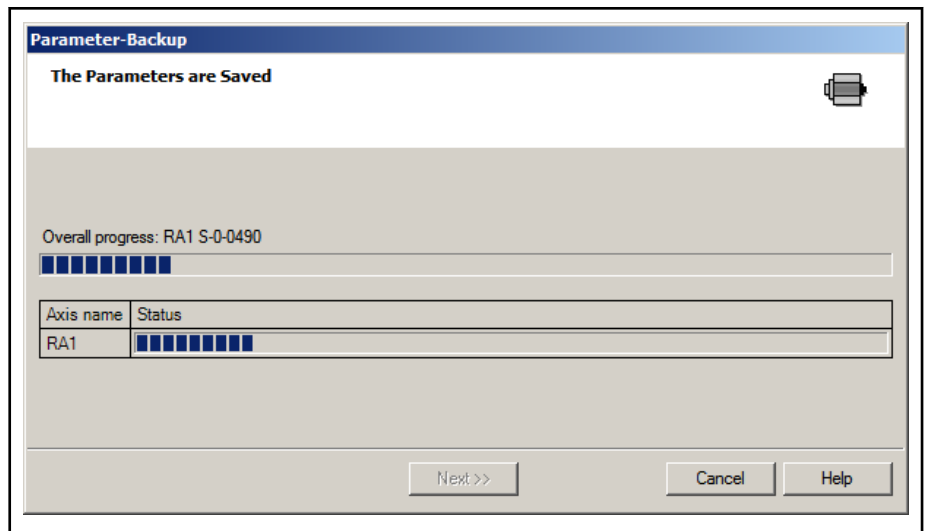


Fig. 8-112: Parameter backup

Preparing for drive exchange

Press **Confirmation** to set the control to a safe state and to disable the drive (only in online mode).

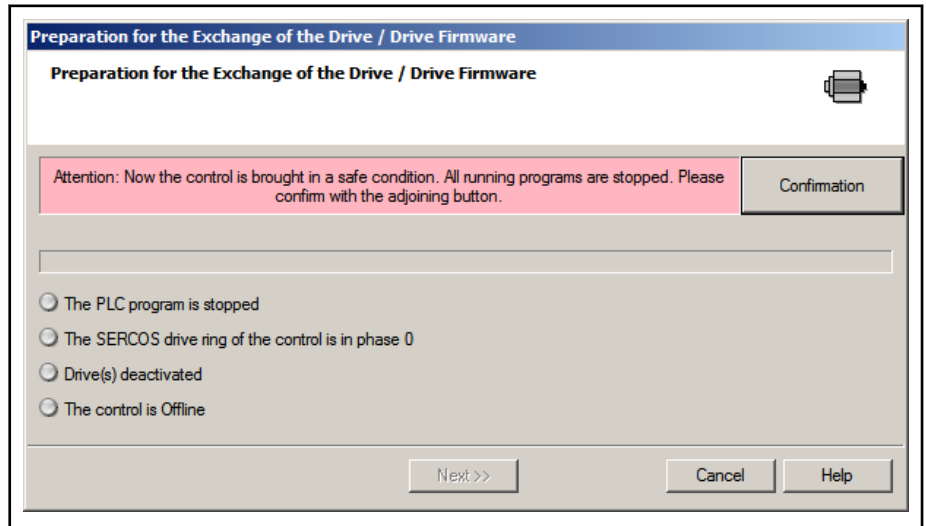


Fig. 8-113: Preparing for drive exchange

Saving drive parameters in the non-volatile memory

Some drives save their current parameters (optionally) only in the volatile memory. Therefore, when going offline, an option is available to save these parameters in the non-volatile memory of the drive.

Axes

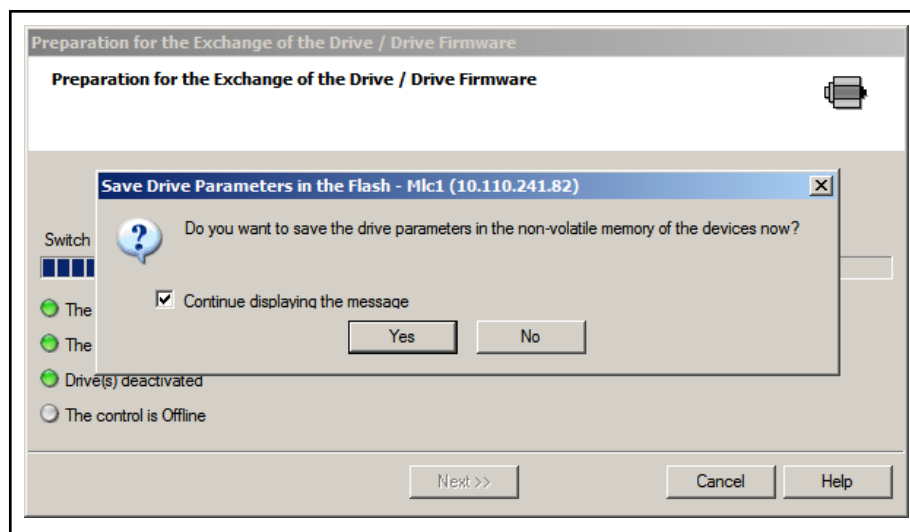


Fig.8-114: Saving drive parameters in the non-volatile memory

Exchanging the drive The drive can now be exchanged.

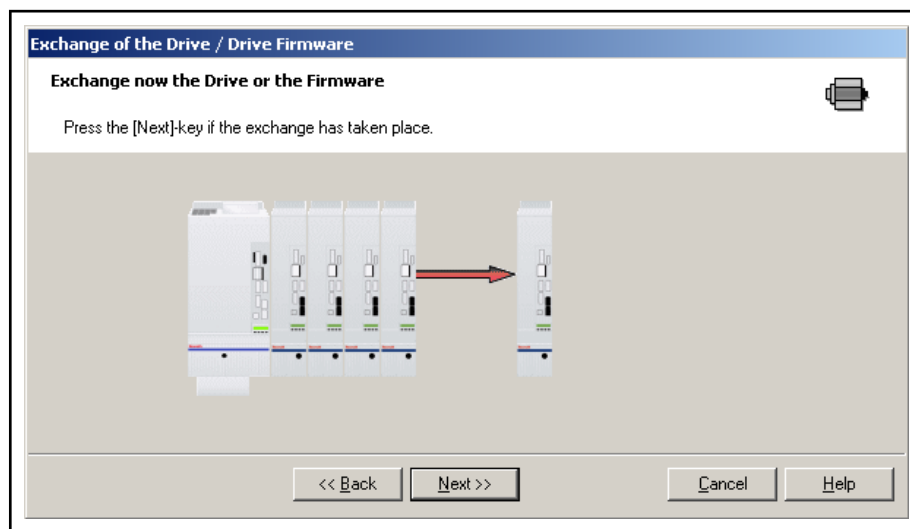


Fig.8-115: Exchanging the drive

Specifying a new drive In this dialog, the new drive can be determined online or entered offline. If there is an online connection, a selection of all drives in the sercos bus of the control is provided. In the "Enter Offline" mode the new drive can be freely determined. If it is an IndraDrive for example, it can also be specify where the interpolation is to take place (in the drive or the control).

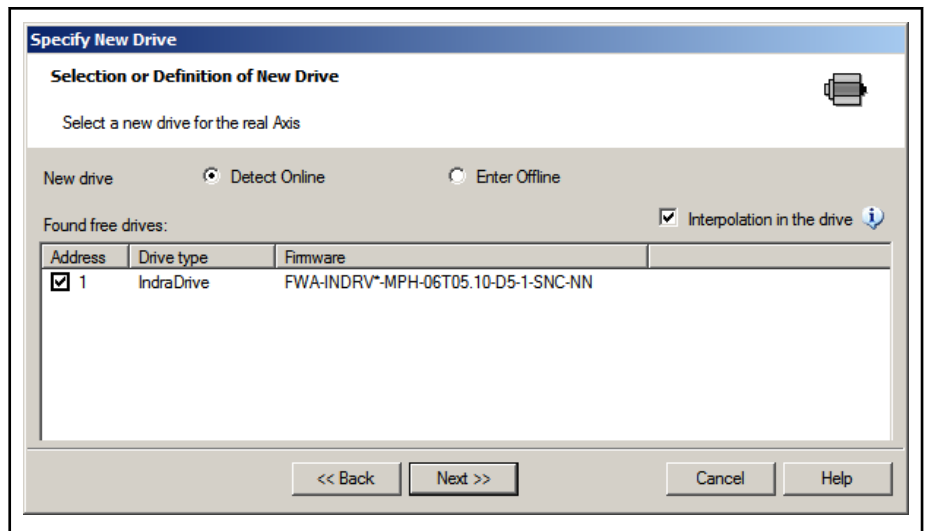


Fig. 8-116: Selecting the new drive

Adaptations in the project

The settings in the project are now adapted.

Only now, actual changes are made to the project! An abort in the preceding steps, leaves the project unchanged.

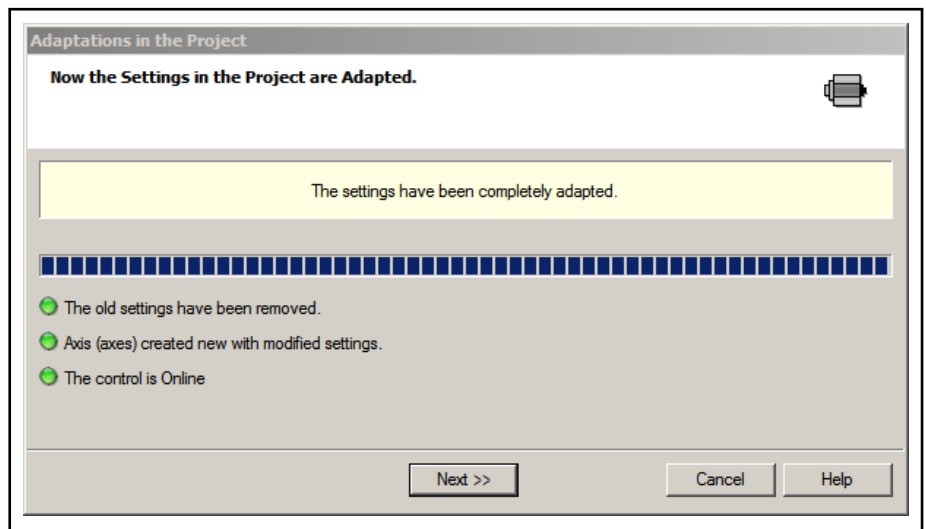


Fig. 8-117: Settings in the project are adapted

Parameters - Restoration

The saved parameters are imported.



If the drive type or the drive was exchanged by one with a newer firmware, it is "normal" that the saved parameters no longer completely match with those currently present. It is likely that several parameters cannot be written into the drive, i.e. the attempt is acknowledged with a sercos error message. All the parameters affected are output in a list together with the error message in the lower section of the dialog.

Axes

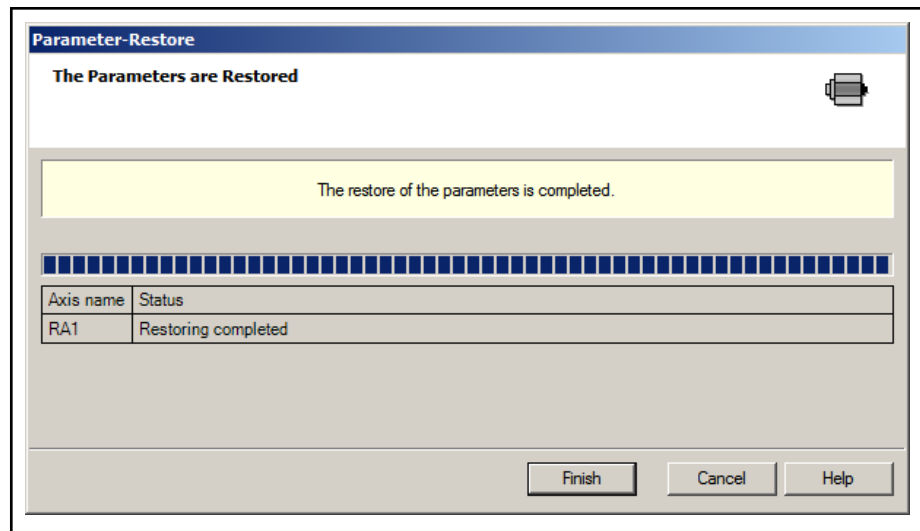


Fig.8-118: Parameters - Restoration

Finish completes the exchange of the drive.



If a parameter file was created in the first two steps, it is also preserved after the Wizard was ended.

8.7.31 Parameters

Loading Basic Parameters

Restore the basic function of a drive with "Load basic parameters".

Practically, the axis is removed and newly created. All parameters directly located in the axes (stored in the S- and P-parameters of the drive) are assigned with default values like all other A-parameters in the control.

Creating a new axis:

When creating a new axis, note that the drive as well as the control have to have certain presettings.

1. Creating new axis (completely new)
 - Dragging drive to the Project Explorer
 - Going online
 - Initial commissioning "Load Basic Parameters" or via parameter handling
 - A-0-1000, Command: Load basic parameters of the axis: sets the required A-parameters for the drive to default values
Triggering: 0x11, parameter is reset automatically
 - P-0-4090, Configuration of default values, S-0-0262, C07_x
Command "Initial Loading"
Triggering: P-0-4090 0x1, S-0-0262 0x11; P-0-4090 0x0, S-0-0262 0x00
2. Creating new axis (drive already optimized)
 - Dragging drive to the Project Explorer
 - Going online
 - Parameter handling

- A-0-1000, Command: Load basic parameters of the axis: sets the required A-parameters for the drive to default values, not the S- and P-parameters

Triggering: 0x11, parameter is reset automatically

3. Creating new axis (axis was deleted before)



This procedure is not recommended!

- Dragging drive to the Project Explorer
- Going online
- "Load basic parameter" is not executed

All old parameter assignments are applied

Drive Password - Change Password

IndraDrive control devices provide the possibility to protect parameter values from unintentional or non-authorized changing by the password.

Based on a default or known password, this menu item allows to enter a new changed customer password and to confirm it with the repeated entry.

Regarding the write protection, there are three groups of writable parameters:

- The parameters always write-protected, such as motor parameters, hardware identification parameters, encoder parameters, error memory, etc. ("administration parameters"). The values of these parameters ensure the correct functioning and performance of the drive.
- Parameters which can be grouped for a customer and protected with a customer password. It is thus possible to protect parameter values which adapt the drive to the axis after they have been defined.
- All remaining writable parameters which are not included in one of the aforementioned groupings. They have no write-protection.

Password types:

The drive firmware allows to activate or deactivate the write protection for parameter values by means of three hierarchically different passwords:

- Customer password - The values of the parameters of a customized parameter group can be protected.
- Control password - Parameters protected by a customer password are writable. "Administration parameters" remain write-protected.
- Master password - All writable parameters, including "Administration parameters" and parameters protected by customer password can be changed.

Save Mode

IndraDrive drives allow different save modes after a parameter value change. The dialog for the menu item allows presetting the mode, including the loading, saving and selective backing up of the user memory as well as the composition of lists for special handling.

The basic principles of "loading, saving and backing up Parameters" are described product-specifically in in /10/, /11/, [Required and Supplementing Documentations, page 14](#), section "Loading, Storing and Backing Up Parameters".

Support is provided by the implemented online help.

Axes

Saving Parameters in the Drive

Via this menu item, saving the content (parameter values) of the volatile user memory in the non-volatile Flash memory of the drive can be triggered at any point in time.

The basic principles of "loading, saving and backing up Parameters" are described product-specifically in in /10/, /11/, [Required and Supplementing Documentations, page 14](#), section "Loading, Storing and Backing Up Parameters".

8.7.32 Communication

Cyclic sercos Data Channel



This dialog is available for IndraDrive for interpolation in the drive and in the control.

All drives in the sercos ring communicate with the control in each sercos cycle. The most important components are the "master data telegram" (MDT) and the "drive telegram" (AT).

The MDT is the control telegram. Command values are specified. The AT is the response telegram. Actual values are received. Both telegrams have a specified number of places for the parameter specification. The majority of these places is covered by the control firmware to control the drives. However, six places each can be freely assigned.

The term "Cyclic sercos data channel" is thus to be understood as an option to modify the content of the MDT and the AT to a certain extent, i.e. to be able to transfer selected parameters into the sercos cycle. Find these parameters in the PLC program via the "functional variables" of each axis.

Cyclical SERCOS Channel - RA1

RA1

Configuration cyclic write access (MDT)

<axis>.UserCmdDataA	Parameter	Value	Unit
<axis>.UserCmdDataA	S-0-0124.0.0: Standstill window	10.0000	Rpm
<axis>.UserCmdDataB	S-0-0108.0.0: Feedrate override	100.00	%
<axis>.UserCmdDataC	S-0-0036.0.0: Velocity command value	0.0000	Rpm
<axis>.UserCmdDataD	S-0-0104.0.0: Position loop Kv-factor	1.00	1000/min
<axis>.UserCmdDataE	S-0-0041.0.0: Homing velocity	100.0000	Rpm
<axis>.UserCmdDataF	S-0-0037.0.0: Additive velocity command v _r	0.0000	Rpm

Configuration of cyclic read access (AT)

<axis>.UserActualDataA	Parameter	Value	Unit
<axis>.UserActualDataA	S-0-0193.0.0: Positioning Jerk	0.000	rad/s ³
<axis>.UserActualDataB	S-0-0189.0.0: Following distance	0.0000	Deg
<axis>.UserActualDataC	S-0-0052.0.0: Reference distance 1	0.0000	Deg
<axis>.UserActualDataD	S-0-0082.0.0: Torque/force limit value posit	100.0	%
<axis>.UserActualDataE	S-0-0083.0.0: Torque/force limit value nega	-100.0	%
<axis>.UserActualDataF	S-0-0347.0.0: Velocity error	0.0000	Rpm

Fig.8-119: Cyclic sercos data channel

In the selection lists of the dialog, one parameter each can be chosen for the six available places. Only reasonable parameters are provided. The only restriction: Ensure not to assign the same parameter more than once.

If a parameter is selected, its value and - if required - its unit are displayed on the right.



Settings in this dialog can only be made in the parameterization mode. If settings are changed, switching to parameterization mode might be provided.

Example: Access to the parameter data in the PLC program

In the PLC program, the parameters can be accessed (read/write) (for MDT) via the specified functional variables.

In the following example, the value for the feedrate override parameter (S-0-0108) is increased by 5 for the axis "RA1" with the axis number "1":

The variables can be retrieved as index in an array under the respective axis number.

The data set is a structure of the "ML_AXISDATA_SM" type.

Example:

AxisData[1].dwUserCmdDataB_q.INT means axis 1, MDT data container B, interpreted as an INT.

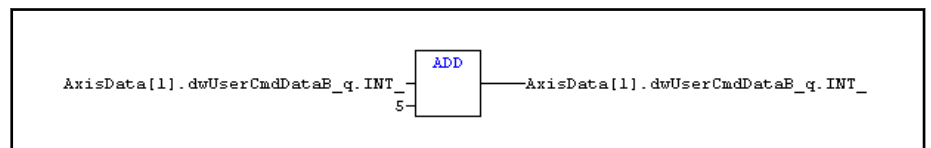


Fig. 8-120: Using the cyclic sercos data channel for writing

For the data exchange, the following parameters

- A-0-0500, Configuration of user-defined command value A
- A-0-0502, Configuration of user-defined command value B
- A-0-0504, Configuration of user-defined command value C
- A-0-0506, Configuration of user-defined command value D
- A-0-0508, Configuration of user-defined command value E
- A-0-0510, Configuration of user-defined command value F

and use of the parameters for the read access

- A-0-0520, Configuration of actual user-defined value A
- A-0-0522, Configuration of actual user-defined value B
- A-0-0524, Configuration of actual user-defined value B
- A-0-0526, Configuration of actual user-defined value D
- A-0-0528, Configuration of actual user-defined value E
- A-0-0530, Configuration of actual user-defined value F



Use the documented access via AxisData[n] for the axis with the logical axis number "n".

Access via direct variable

<AxisName>_DAT.dwUserCmdDataB_q requires approx. the ten-fold access time (15 µs) and is therefore to be avoided!

Signal Status Word



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Axes

The "Signal Status Word" dialog is part of the master communication. It is possible to define up to four bit queries which are read in in each sercos cycle.

The bit values can be accessed in the PLC program via the "functional variables" of the axis.

In order to define a bit query, it is usually necessary to determine a parameter and a bit from its value.

To ensure that these window settings can be transferred consistently to the control, the values are not immediately transferred after editing, but only after the explicit command to accept them with **Apply**. All settings changed, but not yet applied, are highlighted in blue.

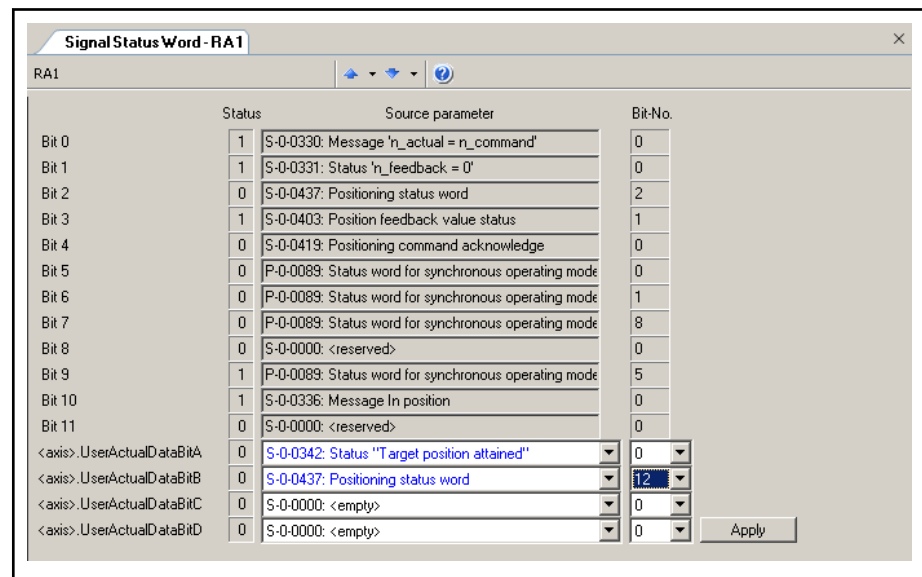


Fig.8-121: Signal status word

In addition to the freely configurable four bit queries, this dialog also displays the other bit queries preconfigured by the control firmware. (Incorrect) multiple definitions of the same bit queries can thus be avoided.



When selecting the bit number, a selection 0..31 is always provided. However, according to the parameter, it is possible that the real bit width is only 0..15.

No test regarding a valid bit number is executed.

Example: In the PLC program, the bit values can be accessed in read-only mode via the specified functional variables. In the following example, it is determined whether the value "S-0-0342, Target position reached (Bit 0)" is set.

The bit values can be accessed in the PLC program via the functional variable of the axis

The variables can be retrieved as index in an array under the respective axis number.

The data set is a structure of the "ML_AXISDATA_SM" type.

Example:

AxisData[1].wUserActualDataBitA_i refers to axis 1, data container A, WORD.



In order to receive a Boolean value, the functional (WORD) variable has to be masked with 0x0001:

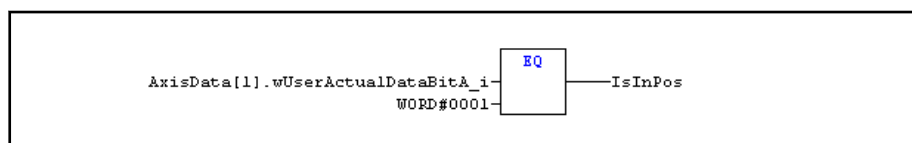


Fig. 8-122: Using the signal status word for reading

Task	Container A	Container B	Container C	Container D
Defines the drive parameter	A-0-0560	A-0-0563	A-0-0566	A-0-0569
Defines the bit in the drive parameter	A-0-0561	A-0-0564	A-0-0567	A-0-0570
Contains the value read from the parameter bit	A-0-0562	A-0-0565	A-0-0568	A-0-0571

Fig. 8-123: Parameters of the signal status word used



Use the documented access via AxisData[n] for the axis with the logical axis number "n".

Access via direct variable

<AxisName>_DAT.wUserActualDataBitA_i requires approx. the tenfold access time (15 µs) and is therefore to be avoided!

Signal Control Word



This dialog is available for IndraDrive for interpolation in the drive and in the control.

The "Signal Control Word" dialog is part of the master communication. It is possible to define up to four bit in parameters written in each sercos cycle.

In the PLC program, the bit values can be accessed (read/write) via the "functional variable" of the axis.

In order to define such a cyclically written bit value, it is necessary to determine a parameter and a bit in its value.

To ensure that these window settings can be transferred consistently to the control, the values are not immediately transferred after editing, but only after the explicit command to accept them with **Apply**. All settings changed, but not yet applied, are highlighted in blue.

Axes

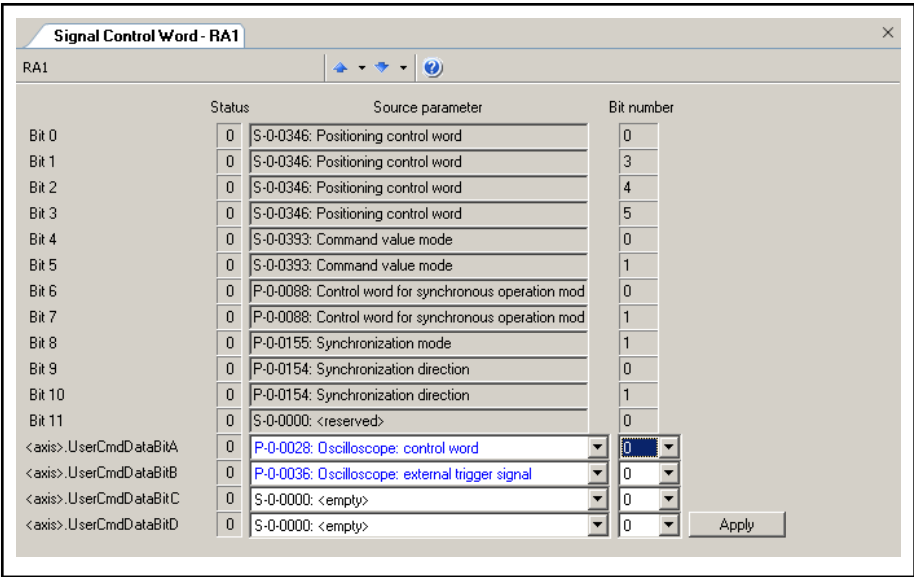


Fig.8-124: Signal control word

In addition to the freely configurable four bit values, this dialog also indicates the other bit values preconfigured by the control firmware. (Incorrect) multiple definitions of the same bit values can thus be avoided.

When selecting the bit number, a selection 0..31 is always provided. However, according to the parameter, it is possible that the real bit width is only 0..15.

No test regarding a valid bit number is executed.

Example: In the PLC program, the parameters can be accessed (read/write) via the specified "functional variables".

The variables can be retrieved as index in an array under the respective axis number.

The data set is a structure of the "ML_AXISDATA_SM" type.

Example:

AxisData[1].wUserCmdDataBitA_q refers to axis 1, data container A, WORD.

In the following example, the oscilloscope recording starts program-controlled (bit 0 of P-0-0028 is set to 1):

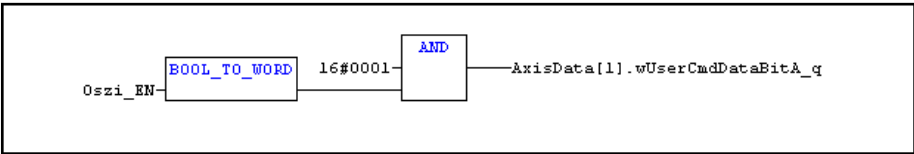


Fig.8-125: Using the signal control word for writing

Task	Container A	Container B	Container C	Container D
Defines the drive parameter	A-0-0540	A-0-0543	A-0-0546	A-0-0549
Defines the bit in the drive parameter	A-0-0541	A-0-0544	A-0-0547	A-0-0550
Contains the value to be written to the parameter bit	A-0-0542	A-0-0545	A-0-0548	A-0-0551

Fig.8-126: Parameters of the signal control word used



Use the documented access via `AxisData[n]` for the axis with the axis number "n".

Access via direct variable

<AxisName>_DAT.wUserActualDataBitA_i requires approx. the tenfold access time (15 μ s) and is therefore to be avoided!

8.7.33 Diagnostics



This dialog is available for all axis types.

With the dialog, the status information on the respective axis is graphically prepared and displayed. In the given example, this is a real axis in an endless motion. The window is split in two, the upper half containing general information, whereas the lower half contains detailed supplements.

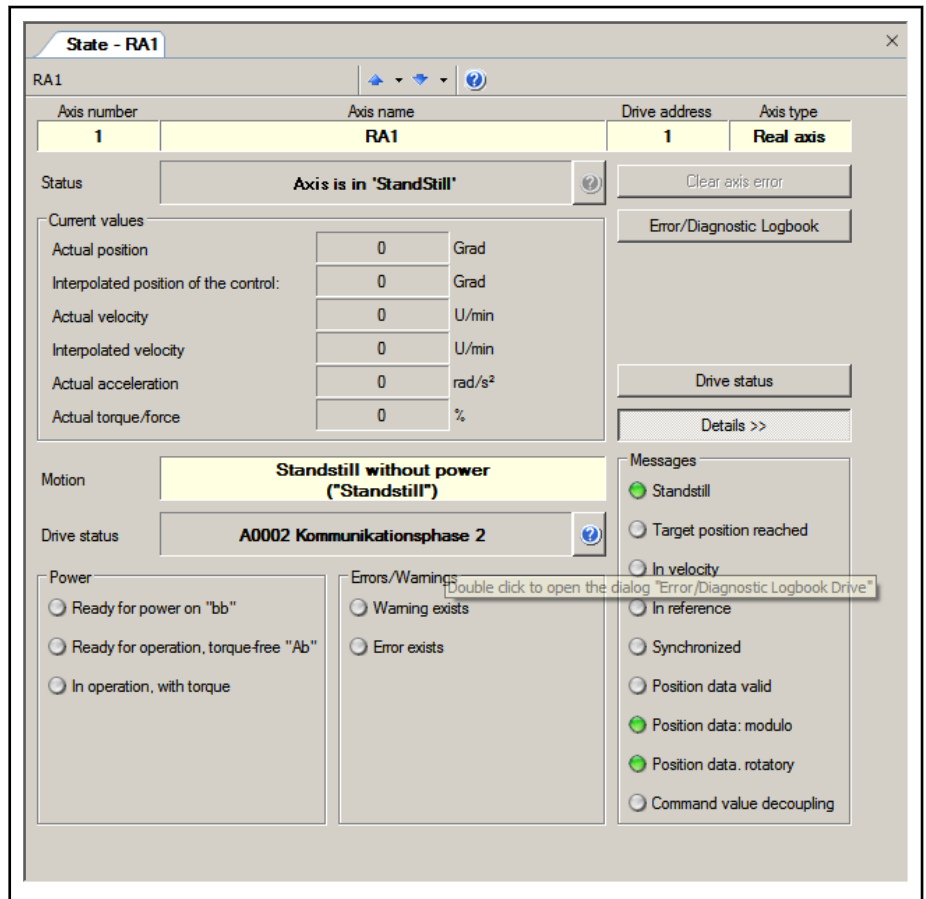


Fig.8-127: Axis status of a real axis

To identify the axis:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Axis ID	x						
Logic axis number	A-0-0001						

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	RA	EncA	LnkA
Axis name	A-0-0002						
Axis type	A-0-0004 Bit 3 - Real axis, IndraDrive or HNC device Bit 4 - Virtual axis Bit 5 - Encoder axis Bit 6 - Link axis Bit 7 - Controller axis						
Drive address, here: sercos address of the drive	A-0-0005					A-0-0005	
Master axis selection							A-0-0008
Axis diagnostic message	A-0-0020						
Command: Clear axis errors	A-0-1030						
Current values	x						
Actual position value	A-0-0100						
Actual velocity value	A-0-0102						
Actual acceleration value	A-0-0103						
Actual torque/force value	A-0-0110						
Axis status	A-0-0021						
Extended axis status	A-0-0022						



The **Drive Status** button starts the status display for the drive belonging to the axis (see online help there).

The detailed information is encoded as bit arrays in the parameters

- A-0-0021, Axis status and
- A-0-0022, Extended axis status



Also double click to the "Axis status" on the desired axis in the "Device status" of the control.

Clearing Errors



This dialog is available for all axis types.

Selecting this context menu item triggers "A-0-1030, Command: Clear all axis errors" and deletes the axis errors in the drive and on the control.

Error/Diagnostic Memory



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive, virtual axis, controller axis.

The dialog displays the operating hour counter and the error memory/diagnostic memory of the drive belonging to the axis.

Double-click on the respective error line/diagnostic line to open the help for the error/warning or message.

Average Value Filter Display



This dialog is available for IndraDrive for interpolation in the drive and in the control.

In the dialog, it is possible to select two drive signals from the parameter "P-0-0396, Average value filter display: Signal selection list" for the parameter "P-0-0395, Average value filter display: signal selection".

The function is disabled if "S-0-0000" is selected as signal.

Both the filters are not interconnected and can therefore be separately enabled or disabled. The parameter is of the Ident number list type and is provided with exactly two values (Ident numbers) for both filters.



Caution: For the oscilloscope, etc., the parameterization sequences have to be followed!

If the signal for the average value filter display is preset first and if the same signal is then selected as measuring signal in the oscilloscope, the filtered value is recorded in the oscilloscope.

If the sequence is reversed, the value not filtered is recorded.

If the signal for the average value filter display is changed or deleted, there is no automatic switching to the oscilloscope function. The signal has to be selected again as measuring signal.

The basic principles are described product-specifically in /10/, /11/, [Required and Supplementing Documentations, page 14](#), section "Velocity Loop (Including the Respective Filters)".

Support is provided by the implemented online help.

9 Motion Functionality

9.1 General Information

This chapter describes the various operation modes of the axes. First, the single-axis modes are discussed such as positioning and velocity control. Subsequently, the synchronous modes, such as velocity and synchronous axis motions and then cams and complex MotionProfiles are described. In the following, the functionality of the programmable limit switch and the touch probe are described.

For a more detailed introduction into the programming of axis motions, please refer to the manuals /23/, "Rexroth IndraLogic XLC IndraMotion MLC 12VRS PLCopen Libraries" and /27/, "Rexroth IndraLogic XLC IndraMotion MLC 12VRS Generic Application Template" (see [Required and Supplementing Documentations](#), page 14).

9.2 Motion Mode

The Motion mode indicates the operating state of the control from the point of view of the drive technology and is closely related to the sercos communication phases P0, P2 and P4, although the focus is more on Engineering.

Download mode (P0): This operating state is used to load a new firmware to the control. Additionally, it is automatically switched to this state if the sercos bus is interrupted. Only very few control parameters can be modified in this operation mode and parameters of sercos devices cannot be accessed at all.

Parameterization mode (P2): This operating state is used to parameterize Motion and control functions (that means that nearly all parameters can be modified). During this period, no motion commands are sent to the drives. From the view of the sercos buses, the acyclic communication is established, but the cyclic communication has not yet been established.

Operating mode (BB): This operating state runs the Motion functionality (that is the drives process the motion commands of the control). These motion commands are normally controlled by the PLC and, in this case, it has to be in the RUN state to send useful motion commands to the drives. Since the parameterized Motion functions can be executed in the operating mode (BB), only very few parameters can be modified. Due to the similarity with the sercos bus behavior, this state is sometimes named "Phase 4" or "P4". From the view of the sercos bus, acyclic as well as cyclic communication are established.

9.3 Parameters

The control and the drives communicate - with very few exceptions - using parameters.

Parameters are used to:

- Determining the configuration
- Parameterizing the control loop
- Triggering and controlling drive functions and commands
- Transferring command and actual values (demand-controlled, cyclic or acyclic)

Parameters can be read and transferred if necessary. If it can be written on parameters by the user depends on the properties of the respective parame-

Motion Functionality

ters and the current communication phase. Certain parameter values (operating data) are checked for validity by the control.

The following tools

- [Parameter Editor, page 133](#)
- [Parameter Group, page 135](#)
- [Parameters - Export, page 136](#)
- [Parameters - Import, page 139](#)

are used to read out, modify and save the parameters in a file or to load the file and write the parameters.

Accesses to parameters are only possible in:

- Online mode; it refers to the real parameters of the control or axes/drives
- Offline parameterization mode; it refers to the parameters of the emulated control or emulated axes/drives

9.4 Modes for Project Editing

9.4.1 Introduction

Offline Mode

In the offline mode, all data saved on your PC can be specified in the project. That is data not requiring any data exchange with a control or drive. Firmware is only downloaded to the control in offline mode.

If a project is opened, IndraWorks is automatically in offline mode.

Online Mode

In the online mode, data requiring communication between the programming device and the control or drive can be changed. This data is stored at least partially in the control or drive or is read from them and is, therefore, only available in case of an established communication. Devices can be created or deleted in offline mode, but not in offline mode.

The online mode is started via the context menu of the control. It is closed when switching to offline mode again.

Access to control or drive data depends on the device state (**P0/P2** parameterization mode/**BB** operating mode) of the Motion component and (**STOP/RUN**) the PLC.

The following rules apply in **online mode**:

1. If no axes are active, the PLC can be stopped and restarted.
2. If at least one axis is active, stopping the PLC causes a forced stop of all active axes.
3. If no axes are active, the user can switch between operating and parameterization mode.
4. If axes are active, the control is in the operating mode. Switching to parameterization mode is no possible.

Many parameters of a control or drive can only be changed if a certain device state is present, e.g. only in "operating mode **BB**" or "parameterization mode **P2**" (also refer to "Parameter Attributes").

Offline Parameterization

If the configured devices (controls or drives) are not available, their parameters can be entered via the "offline parameterization" mode and saved on the programming device.

The "offline parameterization" mode is started via the context menu of the control (as the online mode). Subsequently, all control parameters can be changed as if IndraWorks was in online mode. To change the drive parameters, the respective axis has to be enabled first. Due to reasons of performance, it is recommended to disable all axes that are completely parameterized.

The end the "offline parameterization" mode, switch back to offline mode.

IndraWorks dialogs and behavior are as in online mode. A parameter export or import is also possible at any time.

The changed parameters are automatically saved in the project and can be adjusted with the control or drive parameters via the context menu item **Load Offline Parameters to Device**. Also refer to chapter [Loading Offline Parameters to the Device](#) page 356.

9.4.2 Transitions Between the Modes

Mode Switching

It is possible to switch between the three modes as shown in the following figure:

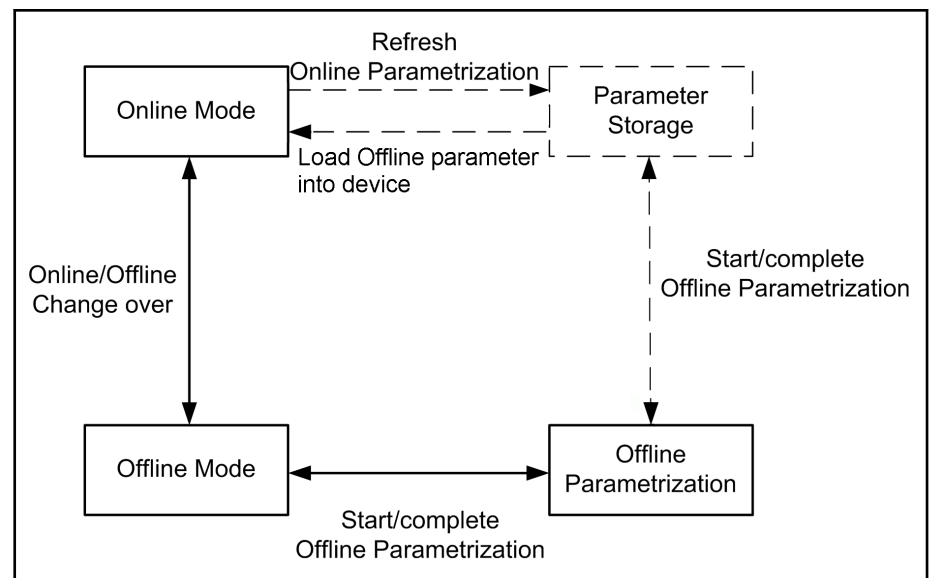


Fig.9-1: Transitions between the modes

- The online/offline switching is executed via the toolbar or via the control context menu item **Switch Online** or **Switch Offline**
- Start the [offline parameterization](#) via the toolbar or via the control context menu item
- Additionally, it is possible to back up the current parameter state in the devices of the programming device in [online mode](#) via the context menu item **Refresh Online Parameterization**.

The following prerequisites have to be fulfilled to connect online to the control:

Motion Functionality

- The type of the projected control has to match the real control. Thus, it is not possible to switch a project online with a projected control of type IndraMotion MLC L45 to an actually connected control of the type IndraLogic XLC L45
- The projected firmware version has to match the actually connected control. Thus, it is for example not possible to switch a project online with a projected firmware version MLC 12VRS to an actually connected control with a firmware MLC 11VRS.

If there are deviations, the projected control has to be adjusted to the actual conditions. See also section ["Device type" on page 57](#)

If parameters are saved in the project, the parameters can be synchronized in online mode with actually connected devices from the control via the context menu item **Load Offline Parameters to Device**. Also refer to chapter [Loading Offline Parameters to the Device page 356](#).

Adjustment of the Command/Actual Configuration

To adjust the command and actual configuration of the sercos devices in the ring, use the dialog "Adjustment of sercos Device Configuration". This dialog can be opened in online and offline mode via the context menu of the sercos node. Additionally, it is displayed when going online if the sercos configuration check outputs a warning (refer to the following figure).

Case 1: Going online

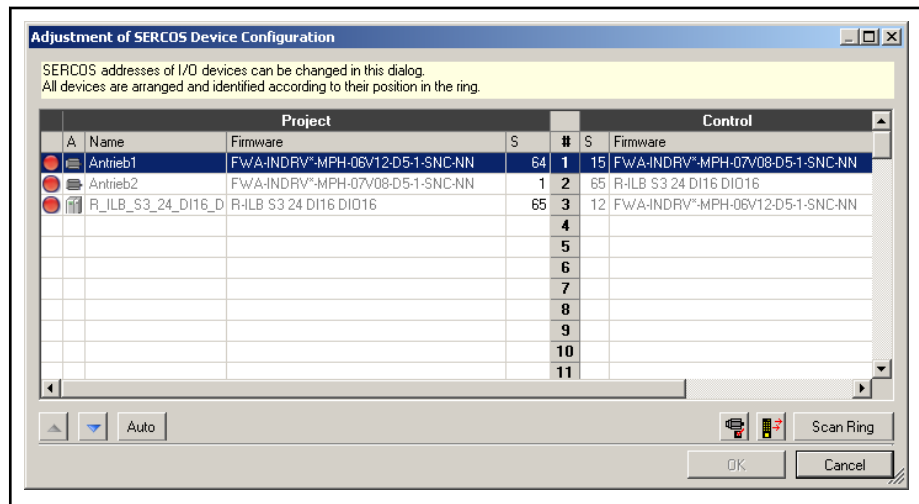


Fig.9-2: Example of the sercos configuration dialog when going online

The left part of the table displays all projected sercos devices and the right part displays all sercos devices connected to the sercos bus.

To solve the conflicts shown above, proceed as follows:

1. If the sercos bus cabling has been changed in the meantime, the right side of the table can be updated via the **Scan Ring** button, see [chapter "Scanning the Ring" on page 352](#).
2. Click on **Auto**. IndraWorks tries to adapt the projected devices to the actually found devices to match them the best possible (see [fig. 9-3 "sercos Device Configuration" dialog after pressing the Auto button on page 349](#)). All unsuitable projected I/O devices are disabled. In the first column, traffic light symbols are used to show how well the command configuration matches the actual configuration. The traffic light symbols have the following meaning:

Motion Functionality

- Green light: There are no warnings and errors during the sercos device configuration
- Yellow light: There are discrepancies in the configuration (e.g. a projected drive is not connected), but it can be switched online. It might not be possible to switch subsequently to the operating mode
- Red light: The differences between the projected and connected sercos devices is so high that it cannot be switched online, since for example the firmware version does not fit. Adjust the configuration to switch online

The detailed explanations are also provided in a tooltip on the light.

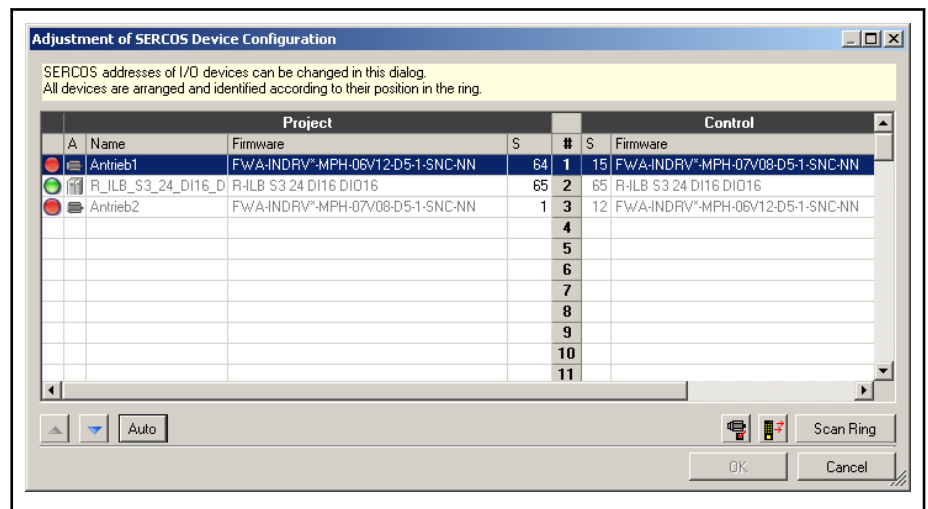


Fig.9-3: "sercos Device Configuration" dialog after pressing the **Auto** button

- The **arrow up** and **arrow down** buttons can adjust the manually projected sequence of the sercos devices (topology address) to the actually connected devices. In the example, it is possible to uniquely assign the drives using the firmware.

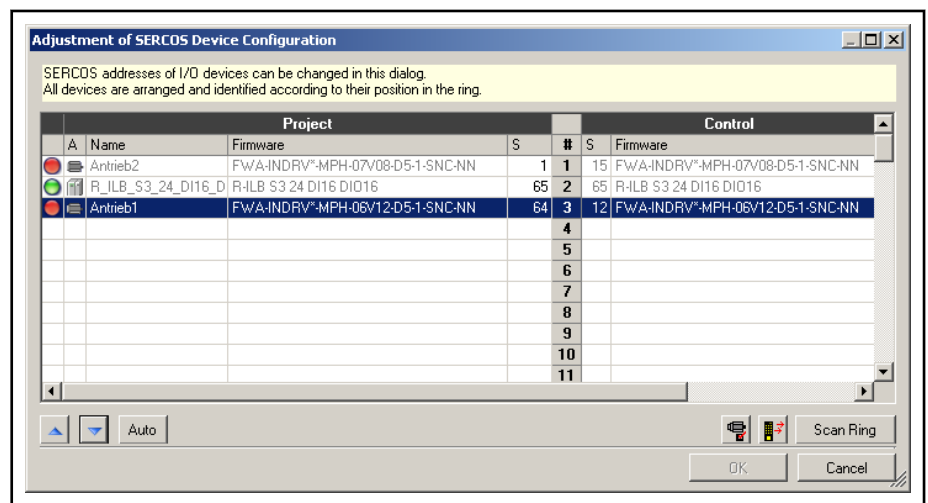


Fig.9-4: "sercos Device Configuration" dialog after shifting the devices (drives)

- Press the **Remote address assignment** button (on the left next to the **Scan Ring**). This writes the projected sercos addresses to the devices (see the following figure).

Motion Functionality

- Press **OK** to save the configuration, close the dialog and continue switching online. The button is only available if the sercos configuration is correct.

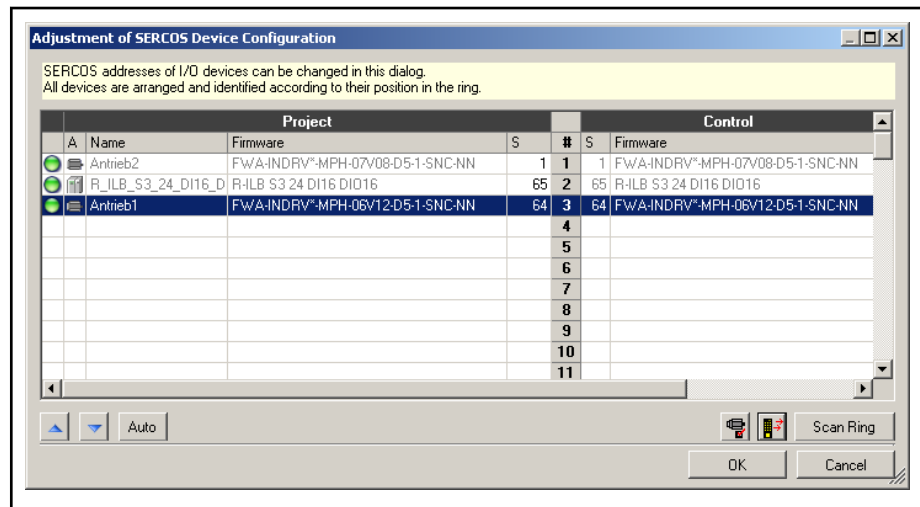


Fig.9-5: Correct sercos device configuration

- Remark: The "Adjustment of sercos Device Configuration" dialog also provides the following function not shown in this display:
 - Click on the icon in column "A" (activation) to activate or deactivate I/O devices.
 - The configured sercos addresses of I/O devices can be changed in the "S" column.
 - The **Activate all connected axes** button (on the left next to the **Remote address assignment** button) ensures that all real axes actually connected to the control are automatically activated. This function is often a lot faster than the manual activation in the Project Explorer.

Case 2: Online

In this case, the command configuration can no longer be changed. The dialog provides only an overview.

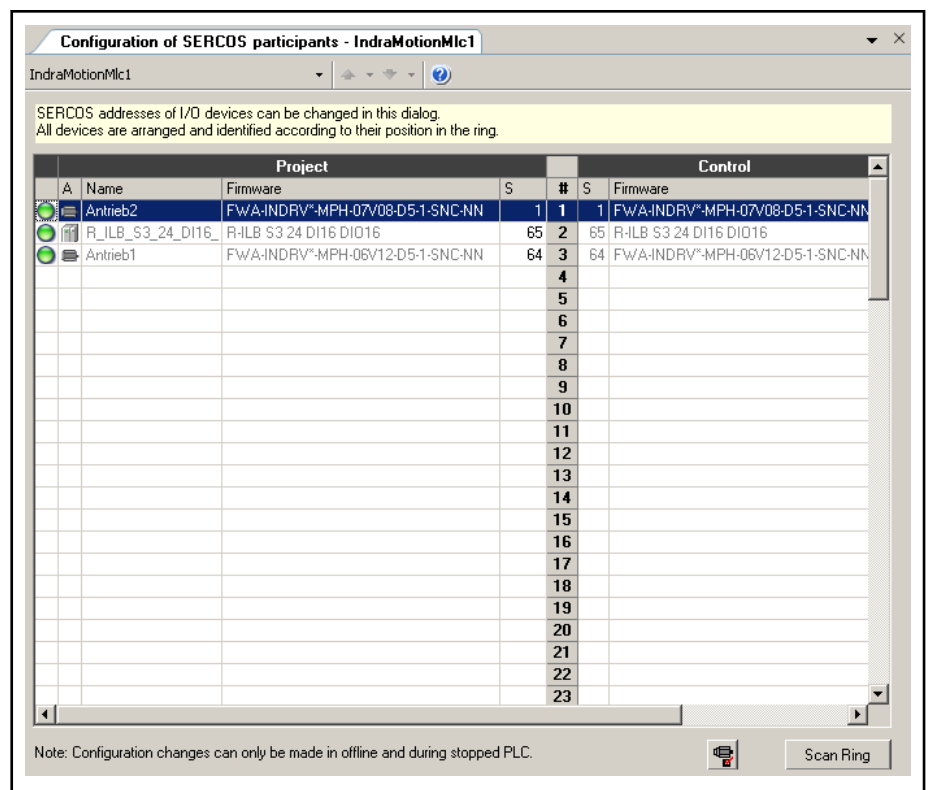


Fig. 9-6: "sercos Device Configuration" dialog in online mode

Case 3: Offline

In this case, the dialog is as in case 1 "Going online". The essential difference is that there are no **OK** and **Cancel** buttons and thus all changes are applied immediately. If the dialog is opened, the right side of the table is empty at first. It is filled when the **Scan Ring** button is pressed.

Motion Functionality

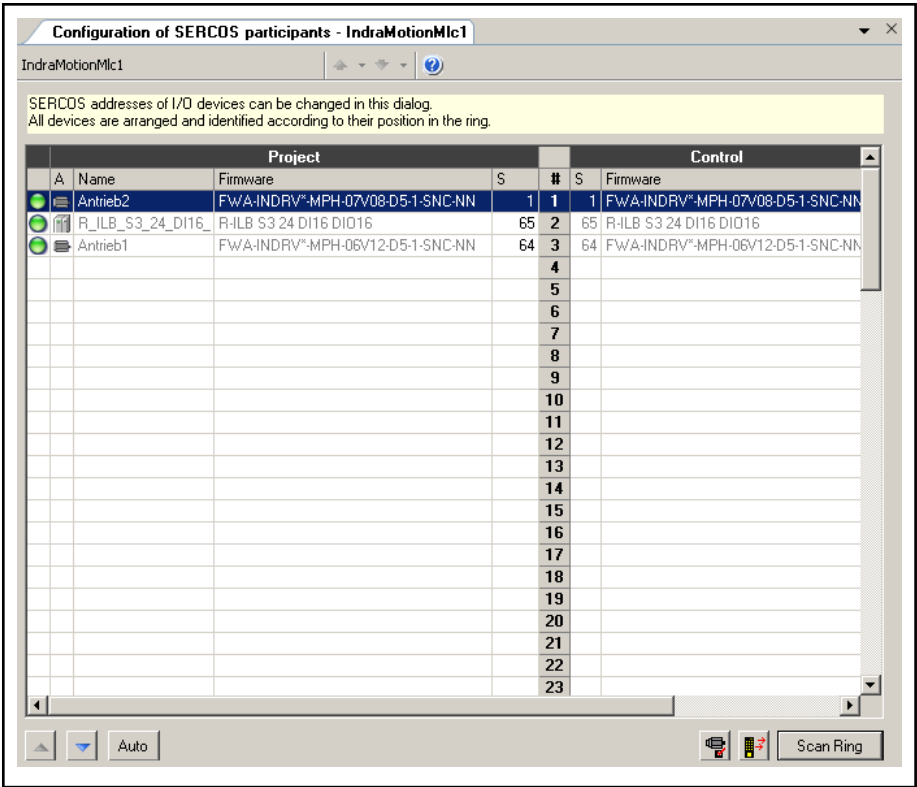
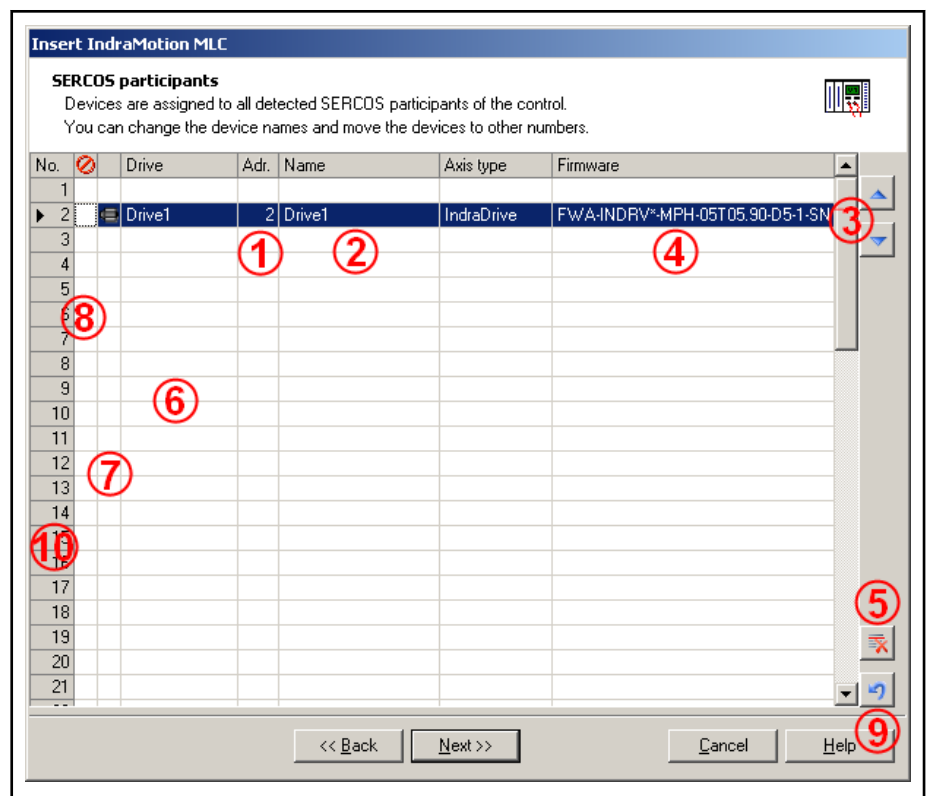


Fig. 9-7: "sercos Device Configuration" dialog in offline mode

Scanning the Ring

sercos II dialog

If the sercos II axes were found, the following dialog is displayed:



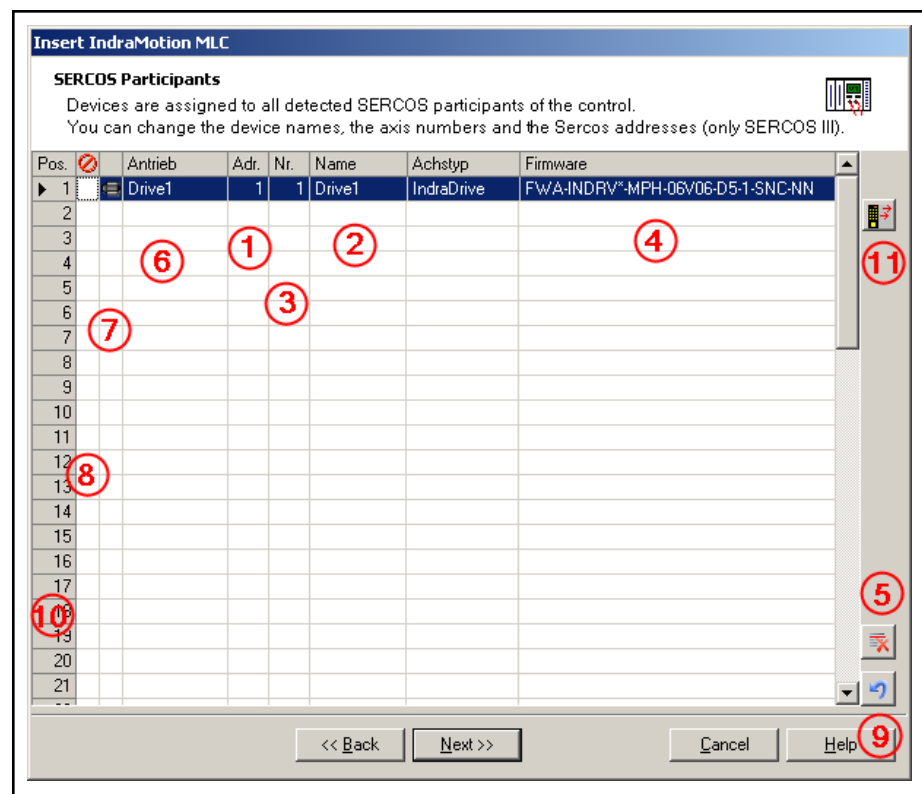
- ① sercos II address
- ② Cannot be changed, axis name corresponds to drive name, exception: Double axis. Here, the distinction is made via the annex
- ③ Move line or the logic address
- ④ Firmware identification detects supported firmware or sercos drive
- ⑤ Delete line/drive
- ⑥ Column to rename the default drive names
- ⑦ Switch between calculation in the control or in the drive if supported by the firmware
- ⑧ Column to deactivate the drives
- ⑨ Return to initial state
- ⑩ Logic address

Fig. 9-8: "Scan Devices sercos II" dialog

sercos III dialog

In case of sercos III, the following dialog is displayed (after scanning the devices):

Motion Functionality



- ① sercos III address (can be changed)
- ② Cannot be changed, axis name corresponds to drive name, exception: Double axis. Here, the distinction is made via the annex. For sercos III I/O, the firmware version is displayed
- ③ Axis number (logic address, can be changed). For sercos III I/O, this column is empty
- ④ Firmware identification detects supported firmware or sercos drive
- ⑤ Delete line
- ⑥ Column to rename the default drive names. For sercos III I/O, this column is empty
- ⑦ Switch between calculation in the control or in the drive if supported by the firmware
- ⑧ Column to deactivate the drives and I/Os
- ⑨ Return to initial state
- ⑩ Topology address (position in the ring)
- ⑪ Enable remote address assignment

Fig. 9-9: Scan sercos III devices dialog

Changed behavior compared to sercos II:

- The dialog shows the devices according to their topology address (for sercos II: sorted according to their axis number)
- The sercos addresses and axis numbers can be changed (for sercos II: only the axis number can be changed by moving the entry)
- The set sercos addresses can be transferred to the sercos devices via the remote address specification
- If several sercos addresses or axis numbers are entered twice, a respective error message occurs as tool tip and the dialog cannot be exited via the **Back** or **Next** buttons.

Axis Adjustment Dialog

Going Online

The axis synchronization dialog allows the user to adjust the axis configuration between IndraWorks and the control when going online.

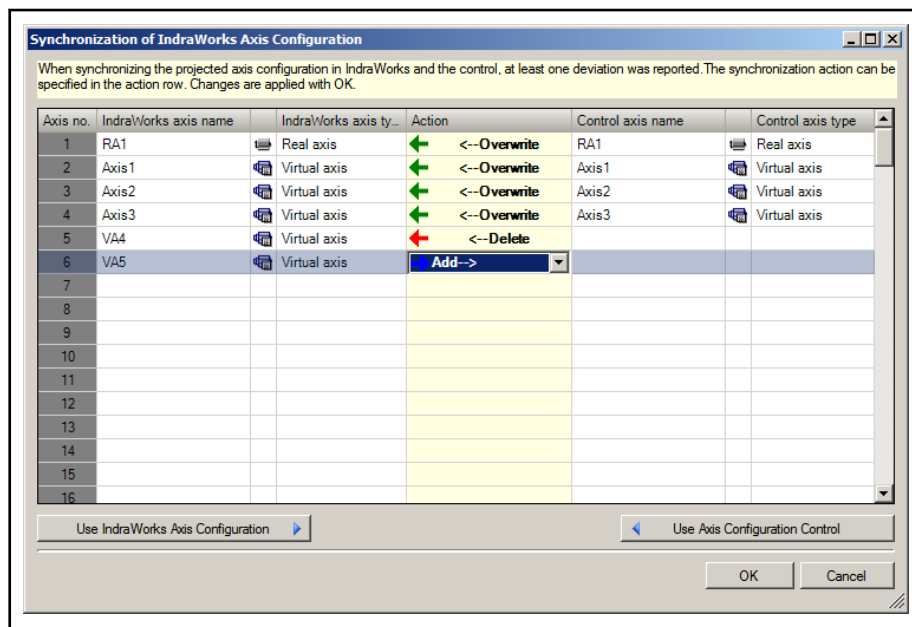


Fig.9-10: Axis synchronization dialog

Axes can be overwritten, deleted and added to the control or to the IndraWorks project using this dialog.

In the "Action" column, the user can separately select the adjustment action for each axis. In case of multi-axes, all axes are seen as entity to ensure a consistent configuration.

Detailed information can be retrieved for each difference via the tooltip in the "Action" column.

Via the **Use IndraWorks Axis Configuration** or **Use Axis Configuration Control** buttons, select all adjustment actions in a way that either the complete IndraWorks configuration or the control configuration is used.

Changes are made in the IndraWorks project or in the control if the dialog was exited with **OK**.

If the project was added to version control (see /5/ [Required and Supplementing Documentations](#), page 14) and changes are to be applied to the IndraWorks project, all axes and drives are checked out after the user query.

NOTICE

If an axis is deleted in IndraWorks via this dialog or overwritten by a control, the axis parameters are not backed up.

Back up the axis parameters before!

Adapting the Behavior of the Axis Adjustment Dialog

The behavior of the axis synchronization dialog can be adjusted via the IndraWorks menu item **Tools ► Options**.

Motion Functionality

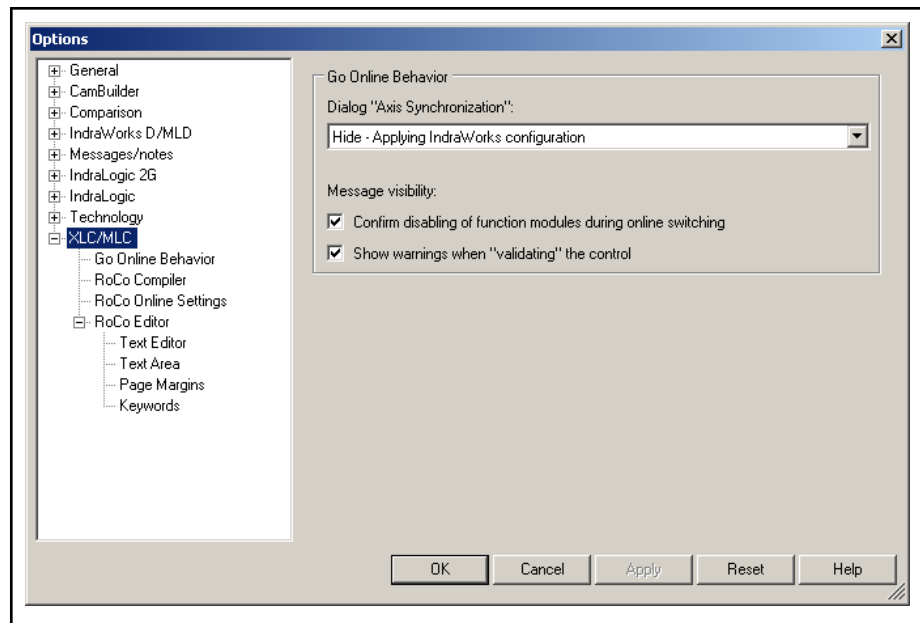


Fig.9-11: IndraWorks menu item Tools/Options

The following options are available:

1. Hide: The dialog is not shown when going online. The axis configuration saved in IndraWorks is written to the control.
2. Show and pre-assign with IndraWorks configuration: The dialog is shown and all axes are preset so that the axis configuration saved in IndraWorks is written to the control with **OK**.
3. Show and pre-assign with the control configuration: The dialog is shown and all axes are preset so that the axis configuration saved in the control is written to the IndraWorks project with **OK**.

Option for disabling function modules when going online:

The following prompt can be controlled via the option "Confirmation prompt before disabling the function modules" when switching online.

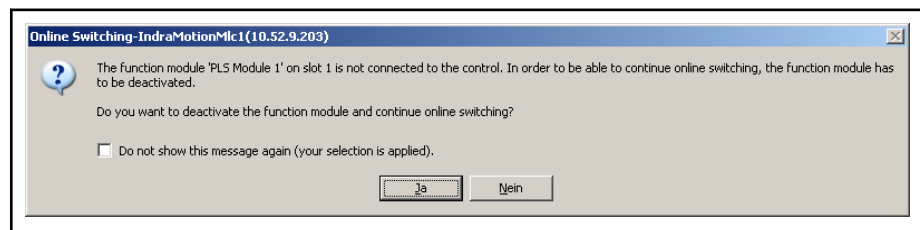


Fig.9-12: Disabling dialog for function modules

This prompt is displayed if a configured, but not equipped function module (except for a master communication module) is used to switch online. This allows the user to switch online even if not all function modules are plugged into the control.

9.4.3 Loading Offline Parameters to the Device

The "Load offline parameters to device" menu item transfer the modified parameters to the real environment (controls, axes, drives and kinematics) during the [offline parameterization](#). For this function, go to the respective menu item in the context menu of the control:

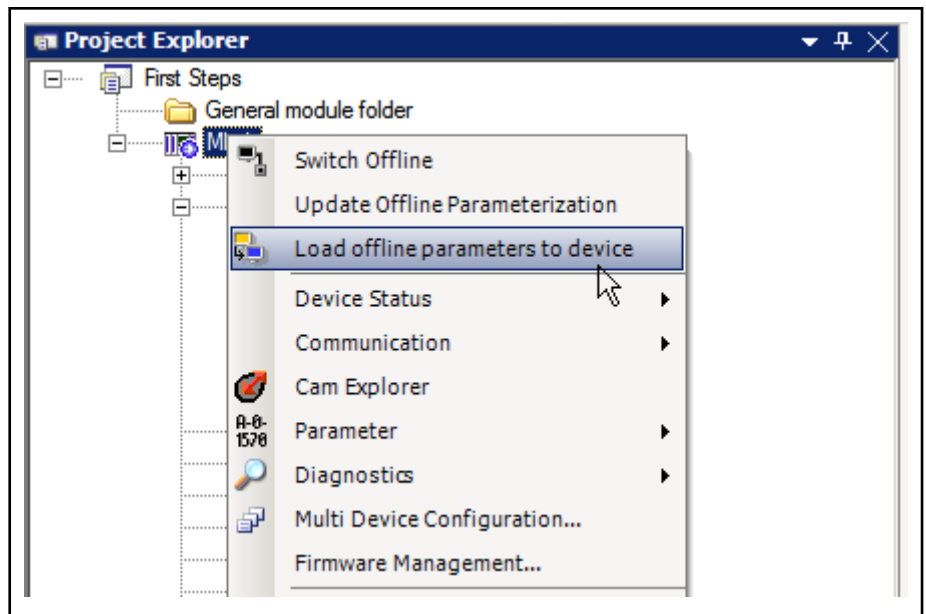


Fig.9-13: Context menu of a control



This menu item is only available if offline parameters exist.

Subsequently, it is queried whether the synchronization (adjustment) is to be executed and if yes, whether a preview is to be displayed. It is also displayed when the offline parameters were created.

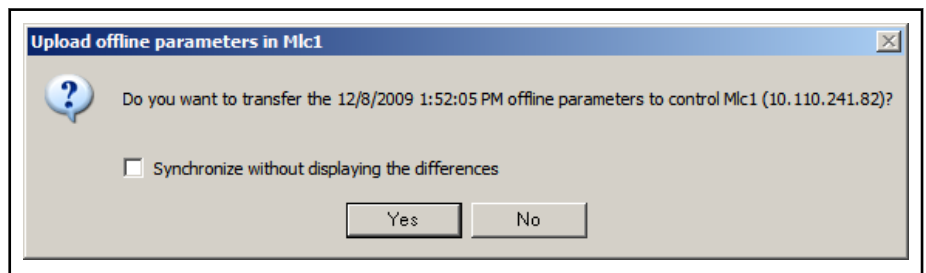


Fig.9-14: Query for synchronization (adjustment)

The offline parameters are then compared with the ones of the control. Differences are displayed if desired:

Motion Functionality

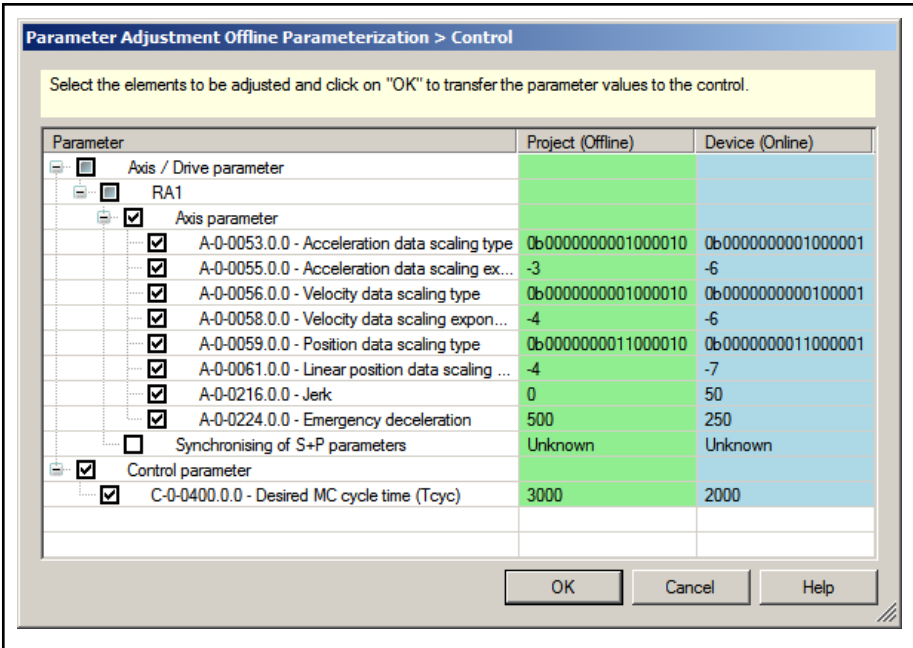


Fig.9-15: *Displaying control parameter differences*

Differences in the parameters are listed in the dialog. The parameters to be transferred from the project to the online device can be selected.

For drives supporting the offline parameterization - currently only drives of the type "IndraDrive" - it is possible to select whether S- and P-parameters are to be adjusted. This option is always displayed irrespective of the fact whether there are differences in the control parameters. The drive parameters are adjusted after the dialog was closed. The differences of S- and P-parameters for each drive are determined and displayed. The parameters to be transferred from the project to the online device can also be selected here.

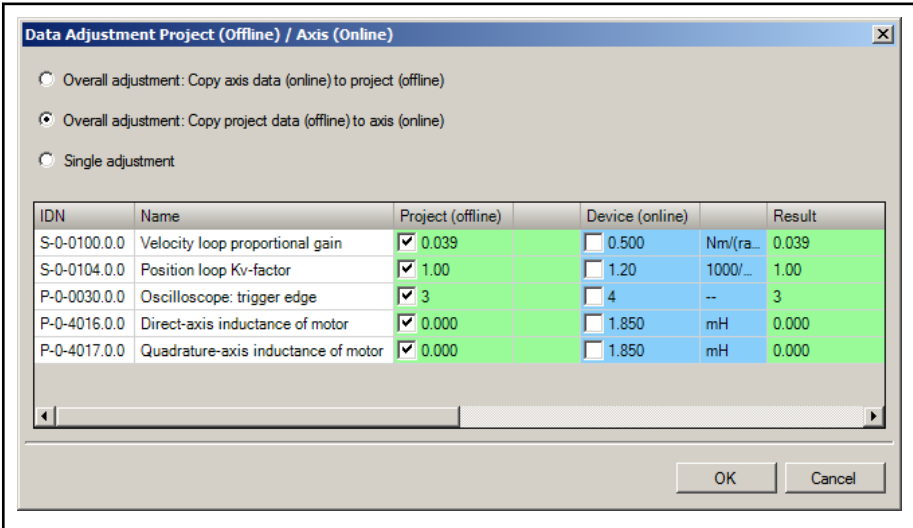


Fig.9-16: *Displaying drive parameter differences*

If the parameters are loaded to the device without displaying the differences, the control parameters as well as the drive parameters are transferred to the respective devices.

The setting whether the synchronization is to be executed with or without displaying the differences is saved and automatically preset when calling the function for the next time.

9.5 Modes to Operate a Real Axis

9.5.1 Deactivated Axis

Disconnecting point: Between control and drive electronics.

Properties The real axis is created in the control project. The "disabled" property can be assigned via the context menu and is then highlighted on the user interface with  in the Project Explorer.

Its A-parameters can be accessed.

For sercos II applies:

- The drive does not have to exist physically

For sercos III applies:

- The drive may not exist physically

The drive is always in **P0**, even if the sercos ring is in **BB**.

Virtual axes and encoder axes cannot be "deactivated".

If the drive electronics is removed from the encoder axis by "deactivating" a real axis, it is also labeled as "deactivated".

- Use cases**
- During commissioning, only certain drives should be moved. The other drives are configured and also programmed in the PLC program, but they do not yet exist physically or are currently built.
 - Master project: A machine manufacturer designs a scalable device series with different stages of expansion. To allow an effective software production for the individual devices, the manufacturer creates a master project with the maximum state of expansion ($\leq 32/64$ axes). On devices with a lower stage of expansion, the non-existing axes are configured, but not activated. Thus, a subsequent upgrading of the device is possible without software modifications.
 - During the [offline parameterization](#), only individual drives are parameterized. The drives or real axes are thus deactivated at first. To start their offline parameterization, the real axes have to be activated.

9.5.2 Parking Axis

Disconnecting point: Between drive electronics and motor.

Properties The real axis is created in the control project. The "parked" property can be assigned via the context menu and is then highlighted on the user interface with  in the Project Explorer.

The axis is active and provided with the "Parking" property.

The drive has to exist physically. The motor does not have to exist. The drive command "Parking Axis" is used.

The A-, S- and P-parameters can be accessed.

The drive follows the sercos phase. No error messages are generated by the drive.

Virtual axes and encoder axes cannot be "parked".

- Use cases**
- It is prevented that a drive controller generates an error when switching to **BB** without a connected motor
 - The S-parameters and P-parameters remain reachable (e.g. writing in **P2** from the PLC program)

Motion Functionality

9.5.3 Notes on How to Use Deactivated and Parking Axes

Parking and deactivating is only possible in **P2**:

- When parking the axes, no phase switching to **P0** is required
- When deactivating the axes, a phase switch to **P0** is required (automatically executed by the firmware)



With the active axes in **P0**, the PLC program can temporarily not access S- and P-parameters.

To all axes applies, whether activated, parked or deactivated, that they

- are provided with an axis number and a sercos address
- are provided with a global axis variable in the PLC referencing the respective axis number

Axis	Can be traveled	Access to A-parameters	Access to S- /P-parameters	Phase switching	Commands possible	Function block response upon call in the PLC program	Parameter check in the drive
Active	Yes	Yes	Yes	Follows sercos	Yes	Normally without error, "Done" when command value is reached	Activated
Parking	No	Yes	Yes	Follows sercos	Partially	Error message	Deactivated
Deactivated	No	Yes	No	Always in P0	No	Error message	Always in P0

Fig.9-17: Properties

9.5.4 Command Value Decoupling

Mode of Operation

The command value decoupling allows the execution of traversing commands for a real axis with interpolation in the control but without a motion carried out in the drive. The command value decoupling is activated via the parameter "A-0-0024, Axis state".

The present states

- 0 - Axis activated
- 1 - Axis parked
- 2 - Axis deactivated

were extended by the following states:

- 4 - Axis activated without command value processing in the drive
- 5 - Axis activated with command value calculation. Drive is parked
- 6 - Axis activated with command value calculation. Drive deactivated (not in the ring)

The axis is subject to the normal PLCopen state machine with active command value decoupling for a real axis. This means that the MC_Power function block has to be executed before executing a motion command. This does not cause the drive to be switched to **AF**. The **AB** state in the drive is not required for the execution of the function block.

In case of an active command value decoupling, the position command value calculated by the control are still written cyclically to the parameter S-0-0047. Since the drive is not in **AF**, the command values are not processed. The actual position value of the axis (A-0-0100) is not updated. The calculated command value of the axis can be read out via the parameter "A-0-0151, Interpolated position of the control".

An axis operated with an active command value decoupling

- can be used as master axis
- set bit 10 "Actual position value is relative/is referenced" in the parameter A-0-0021.
- set bit 14 "Ab - Control and power element ready for operation" in the parameter A-0-0021
- set bit 31 "Command value decoupling active" in the parameter A-0-0021.

If the command value decoupling is disabled, the parameter "A-0-0151, Interpolated position of the control" is set to the current actual value of the drive. This causes a jump in position.

Difference: MLC 04VRS/09VRS vs. MLC 10VRS:

- MLC 04VRS/09VRS:
If a slave axis follows a master axis operating with an active command value decoupling in a parked or deactivated state, the error F9180003 "Invalid axis registered" is generated when the command value decoupling is disabled. Thus, all slave axes have to be decoupled before deactivating the command value decoupling.
- MLC 10VRS:
If a slave axis follows a master axis operating with an active command value decoupling either in the parked or deactivated state, the error F2220231 "Invalid master axis data" is generated when the command value decoupling is deactivated. Thus, all slave axes are to be decoupled before deactivating the command value decoupling.

Displaying the Command Value Decoupling

The command value decoupling can be combined with the axis states "Activated", "Deactivated" and "Parked". The decoupling is symbolized by an open switch that superimposes the icons of the axis states mentioned before. In the "Axis Status" dialog, the following icons are used for the visualization:

Axis status	Without command value decoupling	Command value decoupling active
Active		
Parked		
Deactivated		

Fig.9-18: Icons to represent axis states

The command value decoupling is normally only activated for a short time by the control while the axes are synchronized. It can also be activated manually via the context menu of the axis.

SercosDrive For sercos-compliant drives (SercosDrive), the menu item "Command value decoupling active" is displayed in the context menu:

Motion Functionality

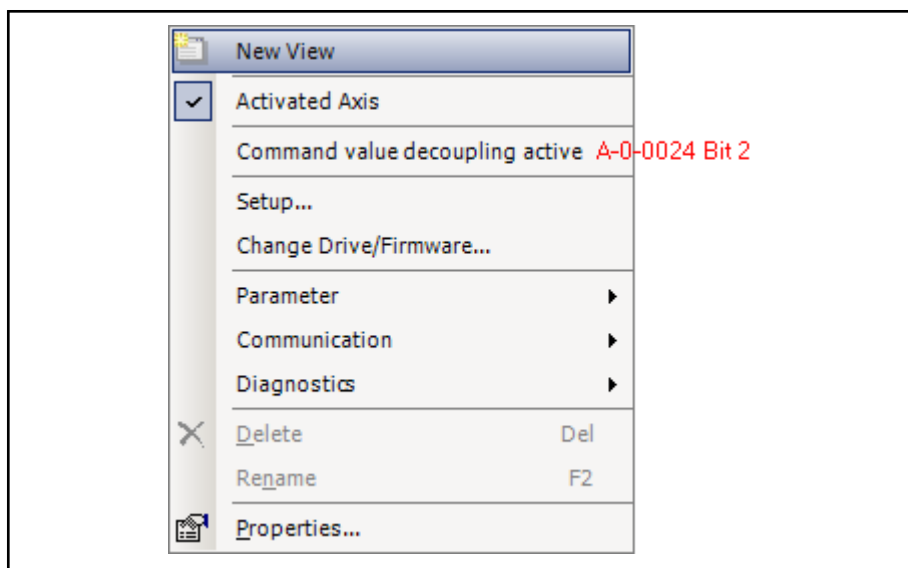


Fig.9-19: Context menu of a servos drive

IndraDrive

The command value decoupling can only be selected if the interpolation is executed in the control. In this case, the context menu looks as follows (command value decoupling selected):

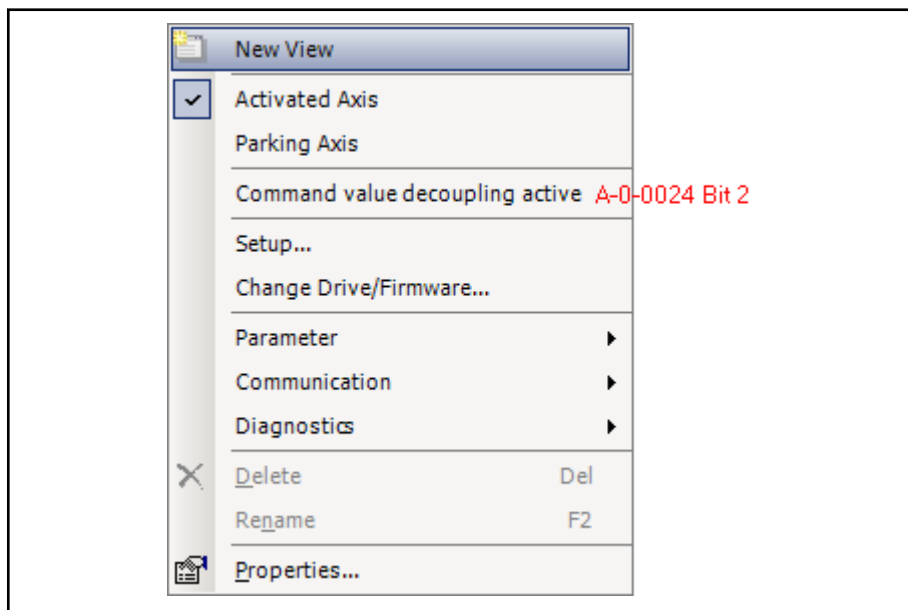


Fig.9-20: Context menu of a IndraDrive drive



If the interpolation is executed in the drive, the menu item "Command Value Decoupling Active" is visible but cannot be selected.

Axis State Dialogs for Real Axes

SercosDrive

For real axes created as sercos-compliant drive (SercosDrive), the interpolation is always executed in the control. The respective axis dialog looks like follows:

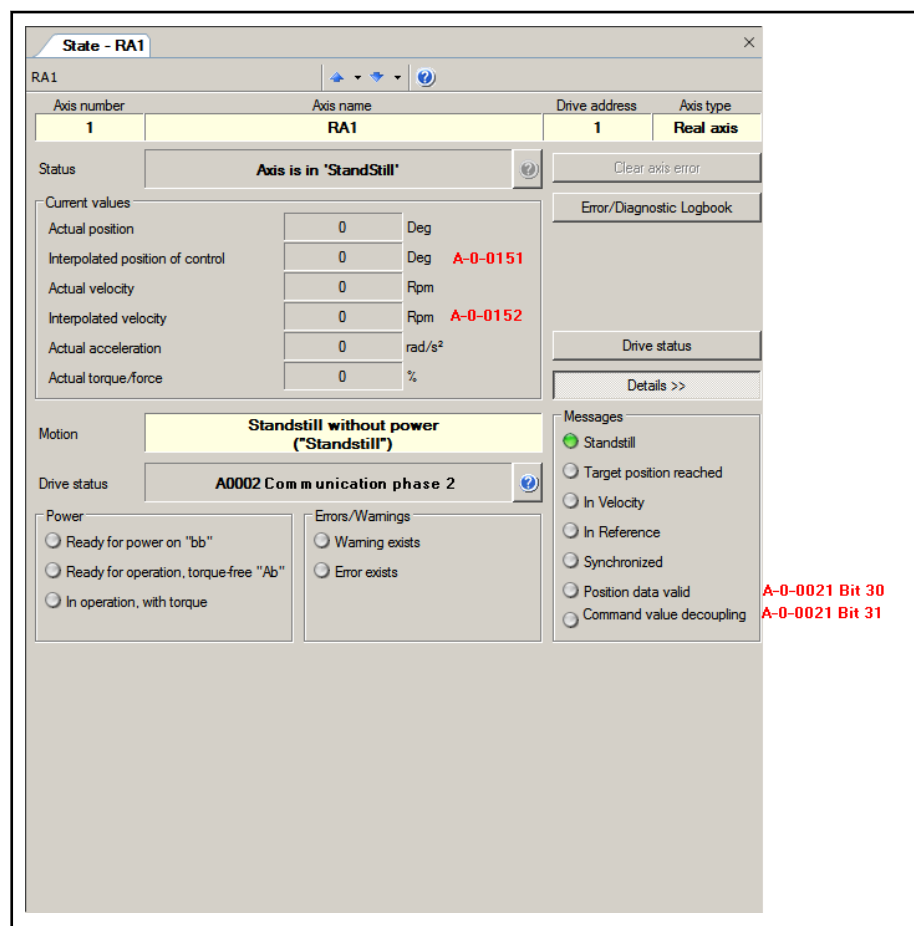


Fig.9-21: Axis state dialog for real axes (SercosDrive)

IndraDrive

For the command value decoupling, additional parameters are displayed in the axis state dialog. The dialog changes with respect to the location of interpolation. The following figure shows the axis state dialog for the IndraDrive drive if the interpolation is carried out in the control:

Motion Functionality

State - RA1

RA1

Axis number	Axis name	Drive address	Axis type
1	Axis_name	1	Real axis

Status: **Axis is in 'StandStill'**

Clear axis error

Error/Diagnostic Logbook

Current values

Actual position	0	Deg
Interpolated position of control	0	Deg A-0-0151
Actual velocity	0	Rpm
Interpolated velocity	0	Rpm A-0-0152
Actual acceleration	0	rad/s ²
Actual torque/force	0	%

Drive status

Details >>

Motion: **Standstill without power ("Standstill")**

Drive status: **A0013 Ready for power on**

Power

- ☒ Ready for power on "bb"
- ☐ Ready for operation, torque-free "Ab"
- ☐ In operation, with torque

Errors/Warnings

- ☐ Warning exists
- ☐ Error exists

Messages

- ☒ Standstill
- ☐ Target position reached
- ☒ In Velocity
- ☐ In Reference
- ☐ Synchronized
- ☒ Position data valid
- ☐ Command value decoupling **A-0-0021 Bit 30** **A-0-0021 Bit 31**

Fig.9-22: Axis state dialog for real axes (IndraDrive) with interpolation in the control

If the interpolation is executed in the drive, the parameters "Interpolated position in the control (A-0-0151)" and "Command value decoupling active (A-0-0021 Bit 31)" are not presented as shown below:

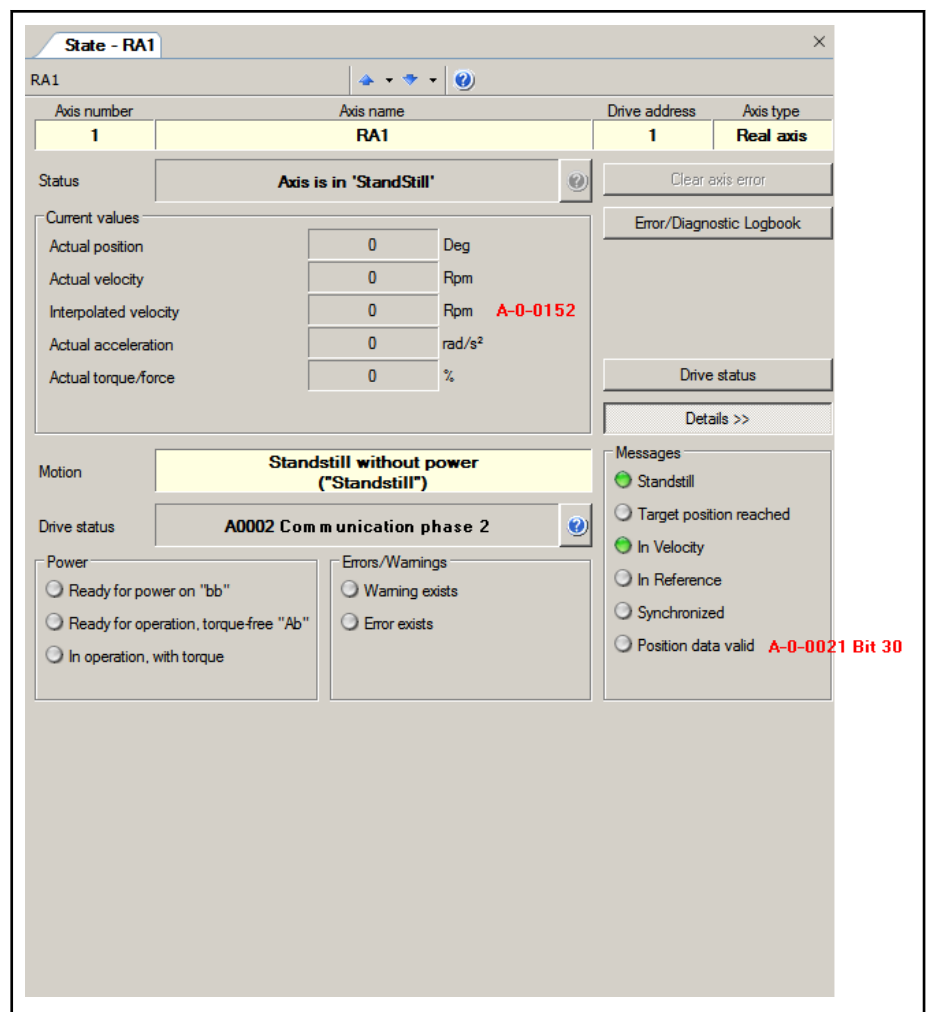


Fig. 9-23: Axis state dialog for real axes (IndraDrive) with interpolation in the drive

Parameters involved

- A-0-0021, Axis state (bits 30 and 31)
- A-0-0024, Axis state
- A-0-0151, Interpolated position of the control
- A-0-0152, Interpolated velocity

9.6 Single-Axis Modes

9.6.1 Overview

This chapter describes the various single-axis modes. A single-axis mode is always active if the axis does not follow another axis (that is if its motion only depends on instructions from the user or the PLC program).

The following operation modes are described in this chapter:

- [Operation Modes \(refer to page 366\)](#)
 - Positioning: Absolute, additive or relative
Motion of the axis to a specific position. The position value can be specified absolutely or relatively to the command or actual position
 - Velocity control

Motion Functionality

- Continuous axis motion with a defined velocity
 - Torque/force control
- Continuous axis motion with defined torque/force
- [Operation Modes for Synchronous Motions with Electronic Gear Function \(refer to page 372\)](#)
 - Velocity synchronization

In the "velocity synchronization" mode, the axis follows a specified master axis velocity velocity-synchronously
 - Phase synchronization

In the "phase synchronization" mode, the axis follows a specified master axis position either absolutely or relatively phase-synchronous
 - Electronic cam

In the "electronic cam" mode, there is a fixed relationship between the master axis position and the slave axis. This relationship is defined in a cam table by points
 - Electronic FlexProfile MotionProfile

The "FlexProfile" belongs to the group of electronic motion laws. These are described as analytic functions by mathematical motion laws

9.6.2 Operation Modes

Absolute positioning

This operation mode enables the time-optimum positioning of a single axis. In "absolute positioning" mode, a target position is directly specified for the axis. In the interpolator, a position command value characteristic is created as input variable for the position controller using the specified value for the target position including the specified positioning data (velocity, acceleration and jerk).

Typical use cases:

- Control of infeed axes
- Manually triggered positioning operations for setup tasks
- Positioning a virtual master axis ...

The following figure shows two examples of successive motions with absolute positioning using the MC_MoveAbsolute function block:

1. The left side of the time diagram shows the case in which "Instance2" is called after "Instance1". After Instance1 has been implemented the desired position 6,000 (velocity 0), its "Done" output activates "Instance2" with the target position 10,000
2. The right side of the time diagram shows how "Instance2" is already enabled while "Instance1" is still running. In this case, the motion of "Instance1" is stopped and aborted by the "Test" signal. "Instance2" directly approaches its target position 10,000, although the position 6,000 has not yet been reached.

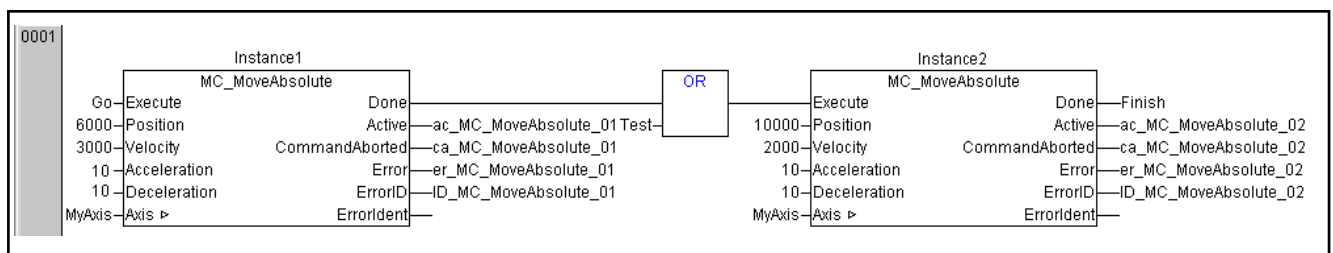


Fig.9-24: Two instances of MC_MoveAbsolute

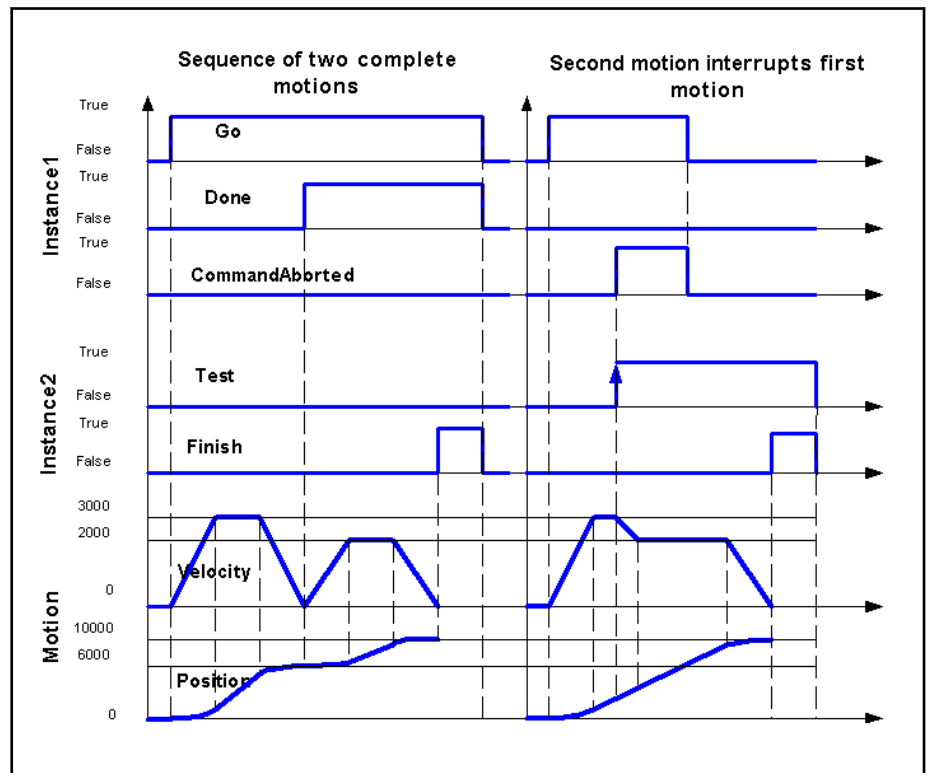


Fig.9-25: Time diagram: Two successive motions of "absolute positioning" mode

Additive positioning

This operation mode enables the time-optimum positioning of a single axis. A distance is specified in the "additive positioning" mode. The distance is added to the current **target position** and transmitted to the interpolator as new target position. The interpolator calculates a position command value sequence for the position controller considering the given positioning data (velocity, acceleration and jerk).

Typical use cases:

- Clocking a stacker

The following figure shows two examples of the combination of absolute and additive positioning:



In this instance, two different types of function block are used:

- Instance1: MC_MoveAbsolute
- Instance2: MC_MoveAdditive

1. The left part of the time diagram shows a case in which the second function block is started after the first one. The first function block rea-

Motion Functionality

ches the desired distance of 6,000 (at a velocity of 0). The "Done" output causes the second function block to travel a distance of up to 10,000.

- The right part of the time diagram shows a case in which the second function block is already started while the first function block is still active. In this case, the first motion is stopped and aborted at constant velocity by the "Test" signal. The second function block adds its distance of 4,000 to the originally commanded position of 6,000 and moves the axis to the resulting position 10,000 at a velocity specified at the second function block.

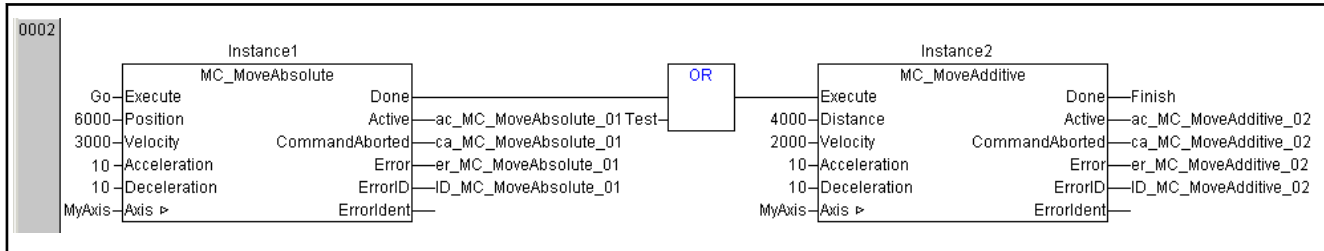


Fig.9-26: Instances of MC_MoveAbsolute and MC_MoveAdditive

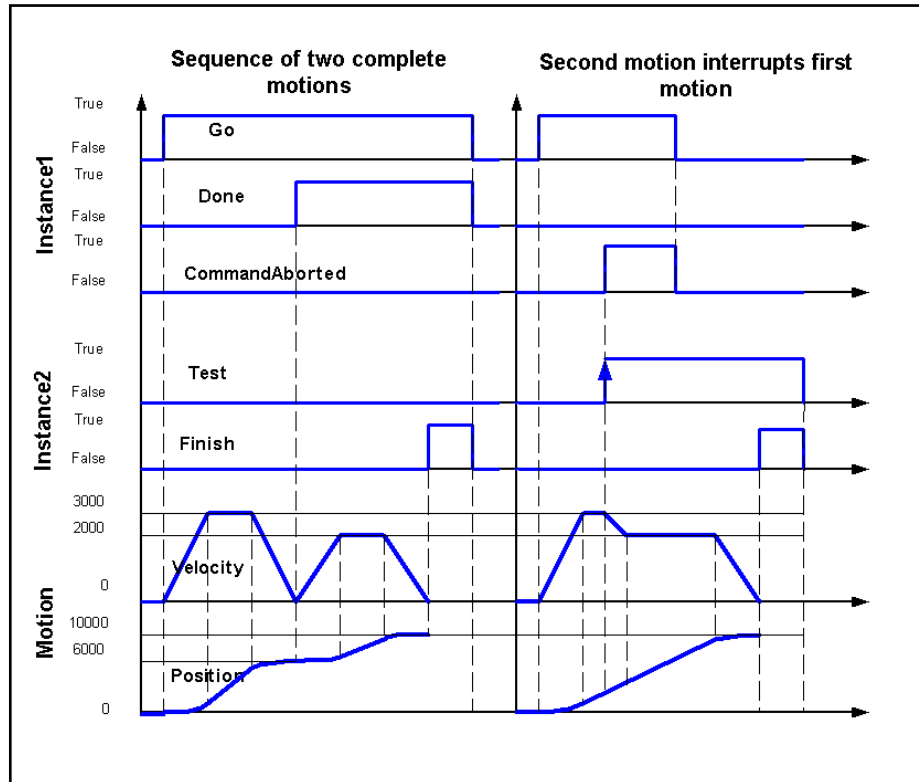


Fig.9-27: Time diagram: Two successive motions of "absolute" and "additive positioning" operation modes

Relative positioning

This operation mode enables the time-optimum positioning of a single axis. A distance is specified in the "relative positioning" mode. The distance is added to the current **actual position** and transferred to the interpolator as new target position. The interpolator calculates a position command value sequence for the position controller considering the given positioning data (velocity, acceleration and jerk).

Typical use case:

- Clocked feed

The following example shows the combination of two successive motions with "relative positioning" using the MC_MoveRelative function block:

1. The left part of the time diagram shows a case in which "Instance2" is called after "Instance1" has been processed. After "Instance1" has reached the desired distance 6,000 (velocity 0), its "Done" output uses "Instance2" to cause a further motion by 4,000 to reach the total distance of 10,000.
2. The right part shows how "Instance2" is already enabled by the "Test" signal while "Instance1" is still running. In this case, the first motion is aborted during the constant velocity of "Instance1". "Instance1" adds its distance of 4,000 to the current position (3,250). The axis approaches the resulting position of 7,250.

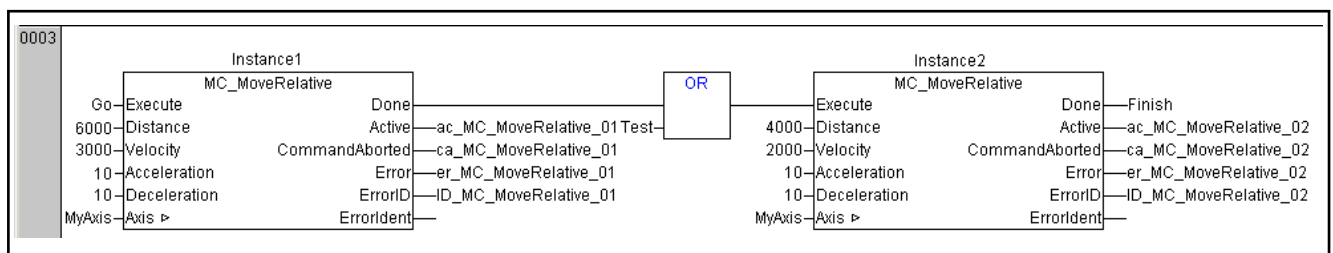


Fig.9-28: Two instances of MC_MoveRelative

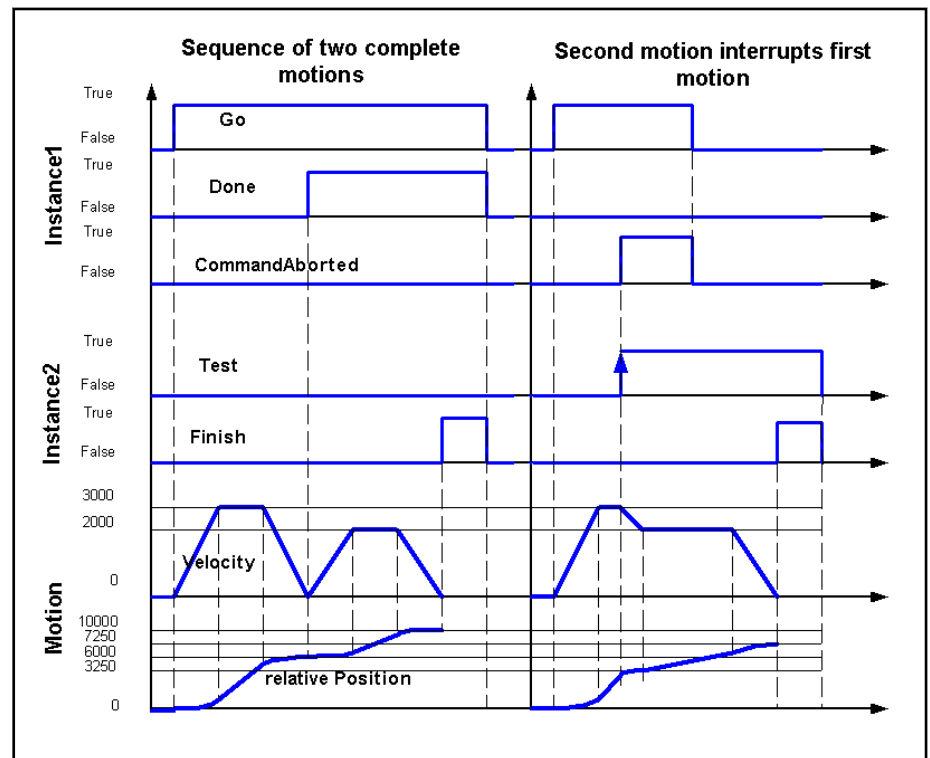


Fig.9-29: Time diagram: Two successive motions of the "relative positioning" mode

Velocity control

In the "velocity control" mode, a velocity command value for an "endless" motion to the next motion command is defined for the axis.

Motion Functionality

Typical use cases:

- Control of the virtual master axis speed
- Continued execution of paint application or paste rolling at machine standstill
- Continuous product feed...

For real axis in the "SercosDrive" mode, the subordinate drive mode "velocity control" is not available. It is not available, since the determination on the sercos application profile PackProfile does not intend a velocity control as secondary mode. In this case, the function block MC_MoveVelocity or the "velocity control" mode is represented in the drive via a "position control" by a control. The velocity command values are integrated into a position command value by the control and the function block "MC_MoveVelocity" is also available for this axis type.

Remark:

The use of the position control leads to a following distance (that might build up) normally corrected by the position controller. This is not the case for a "real" closed-loop velocity control. Whether this missing slippage is relevant, has to be determined application-specifically.

This restriction does not apply to real axes operated in the IndraDrive profile. A "real" velocity control takes place in the drive controller. The interpolation of the velocity command values is executed in the control or in the drive depending on the axis mode (central/decentral) used.



Up to 11VRS, a position control has also been used for the IndraDrive with interpolation in the control. When applying old projects, observe whether the change in the closed-loop control method has effects on the machine processing. Compatibility to the 11VRS behavior can be achieved by changing the axis type in a SercosDrive.

The following figure shows two examples of two combined instances of the MC_MoveVelocity function block:

1. The left part of the time diagram shows a case in which the second instance is called after the job of the first instance has been completed. Once the first function block has reached the required velocity of 3,000, the "Instance1.InVelocity" output **AND** the "Next" signal cause the second function block to continue the motion at a velocity of 2,000.
2. The right side of the time diagram shows a case where the first function block has not yet reached the desired velocity when the second function block is started. The following is shown: The first motion is restarted by "Go" at the "Instance1.Execute" input. While the first function block is still accelerating to reach a velocity of 3,000, it is interrupted by the "Test" signal which starts the second function block. Now, the second function block is running and decelerates to a velocity of 2,000.

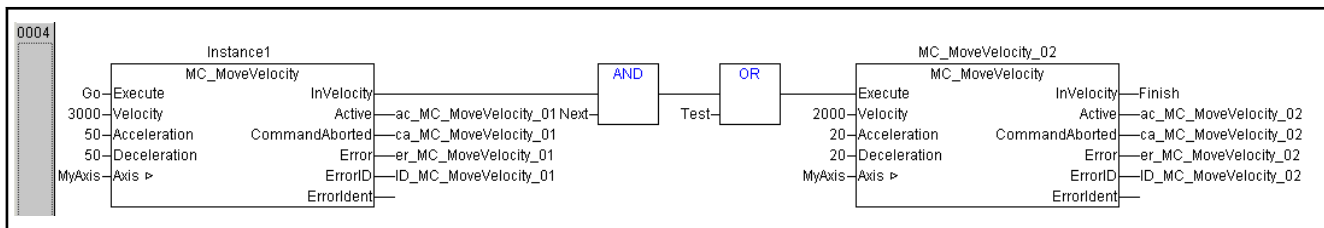


Fig.9-30: Two instances of MC_MoveVelocity

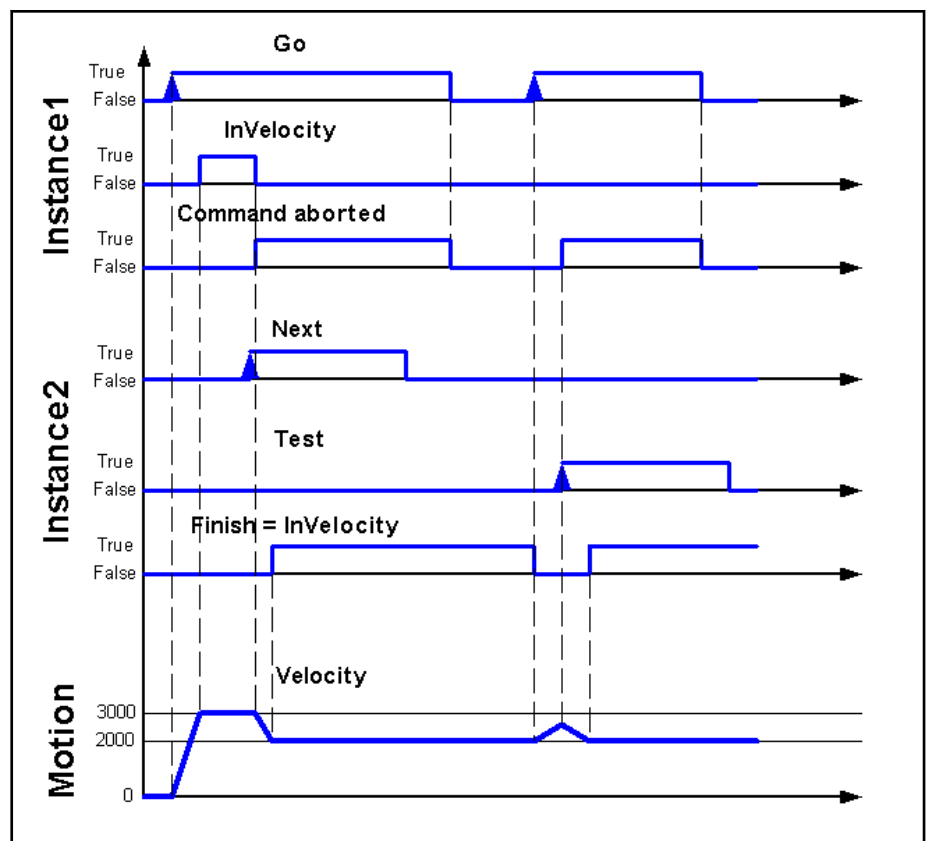


Fig.9-31: Time diagram: Two successive motions of the "velocity control" mode

Torque/force control

In the "torque/force control" mode, a torque/force command value is specified for the drive.

Typical use cases:

- Torque coupling for mechanically coupled ("parallelly switched") axes
- Hydraulic applications

The torque/force control support can also be disabled for a real axis with interpolation in the control. This is performed via the A-parameter A-0-0007 of the respective axis. As a consequence, the operation mode and the respective S-/P-parameters are not parameterized by the control anymore. The parameters and the operation mode are thus not reserved by the control and can be linked drive-internally for special functions such as coupling the torque command value to an analog input.

The "torque/force control" mode is only provided for real axes operated in the IndraDrive profile. For real axes in the SercosDrive profile, no torque/force control can be used by the control.

The figure below shows two examples of two combined instances of the MC_TorqueControl function block:

1. The left part of the time diagram shows a case in which the second instance is called after the job of the first instance has been completed. Once the first function block has reached the requested torque of 30, the "Instance1.InVTorque" output **AND** the "Next" signal cause the second function block to be reduced to a torque of "20".
2. The right part of the time diagram shows a case where the first function block has not yet reached the desired torque when the second function block is started. The following is shown: The first motion is restarted by

Motion Functionality

"Go" at the "Instance1.Execute" input. While the first function block is still operating to reach a torque of 30, it is interrupted by the "Test" signal which starts the second function block. Now, the second function block reduces the torque to 20.

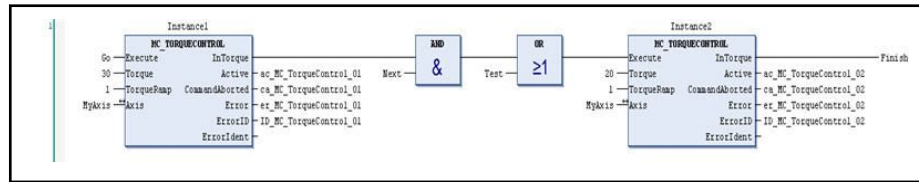


Fig.9-32: Two instances of MC_TorqueControl

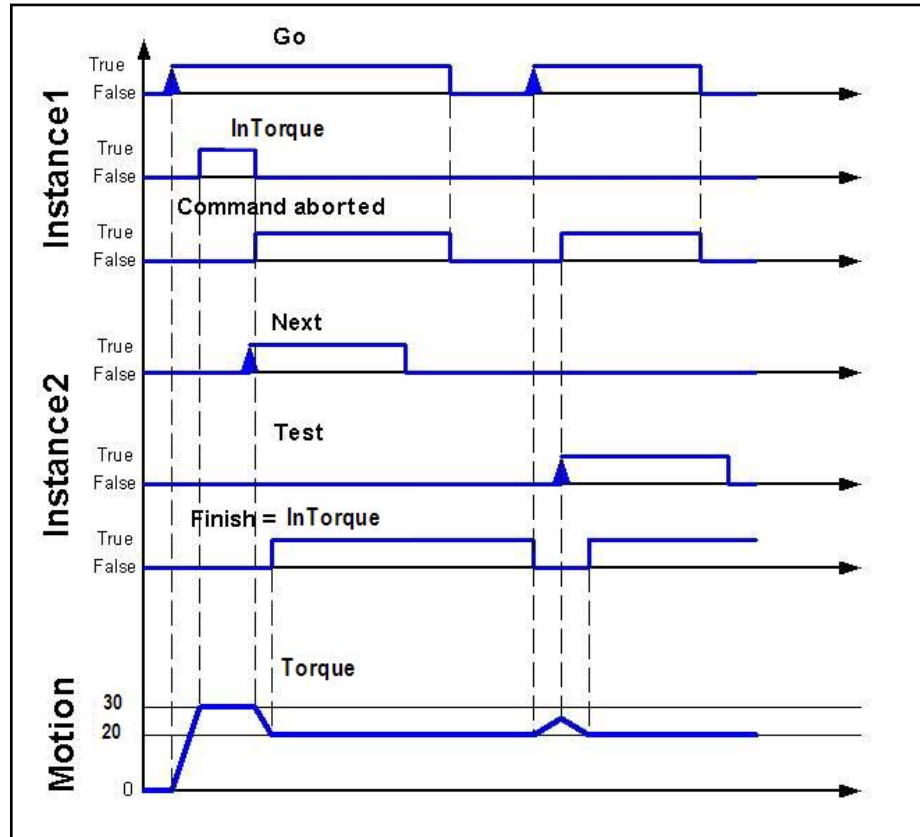


Fig.9-33: Time diagram: Two successive motions of the "torque/force control" mode

9.7 Operation Modes for Synchronous Motions with Electronic Gear Function

Velocity synchronization

In the "velocity synchronization" mode, the axis follows a specified master axis velocity velocity-synchronously. The master axis velocity can be specified by an encoder axis (real master axis), a virtual axis (virtual master axis) or a real axis.

The objective is a velocity-synchronous (drift-free) run between the master axis and the selected slave axis.

Typical use cases:

- Simple transport axes
- Draw rollers in winder applications

- Cooling rolls
- Impression cylinders
- Paint application rollers
- Film/paper drawing ...

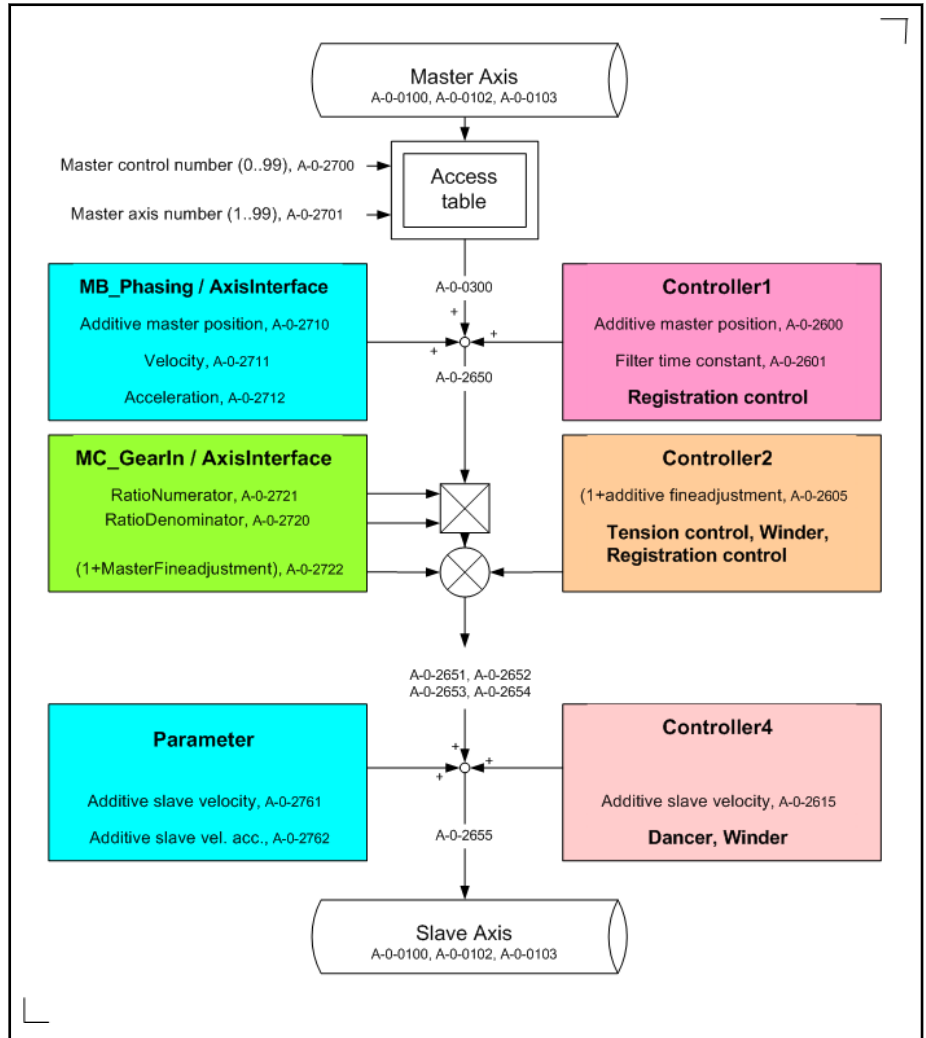


Fig. 9-34: MotionControl function chain: Velocity synchronization

The operation mode provides the following access points:

- The additive master axis position defined by "A-0-2710, Master axis, additive position command value", can be specified via MB_Phasing or the AxisInterface
- Electronic gear defined by the ratio of "A-0-2721, Master drive gear output revolutions" and "A-0-2720, Master drive gear input revolutions" and the "A-0-2722, Gear fine adjustment". The fine adjustment value is specified in percent and calculated as $(1 + \text{fine adjustment})$. These values are specified via MC_GearIn or the AxisInterface.
- The additive command velocity of the slave axis defined by "A-0-2761, Slave axis, additive command velocity"

For real axis in the "SercosDrive" mode, the subordinate drive modes "velocity control" or "velocity synchronization" are not available. It is not available, since determining on the sercos application profile PackProfile does not intend a velocity control/synchronization as secondary mode. In this case, the

Motion Functionality

function block MC_GearIn or the "velocity synchronization" mode are represented in the drive via a "position control" by a control. The velocity command values of the slave axis are integrated into a position command value by the control and the function block MC_GearIn is also available for this axis type.

Remark:

The use of the position control leads to a following distance (that might build up) normally corrected by the position controller. This is not the case for a "real" velocity control/synchronization. Whether this missing slippage is relevant, has to be determined application-specifically.

This restriction does not apply to real axes operated in the IndraDrive profile. A "real" velocity control takes place in the drive controller.

- At interpolation in the drive, it is synchronized via the "velocity synchronization" mode. The master axis velocity is transferred to the drive. The drive device calculates the synchronous velocity.
- In the mode with interpolation in the control, the "velocity synchronization" mode is not required in the drive. The control applies the calculation of the resulting synchronous velocity command values of the slave axis and transfers this value as cyclic velocity command value to the drive.

Axis mode	SercosDrive	IndraDrive (interpolation in the control)	IndraDrive (interpolation in the drive)
Drive type used	Position control	Velocity control	Velocity synchronization

Fig. 9-35: Drive type used in different axis modes

Phase synchronization

In the "phase synchronization" mode, the axis follows a specified master axis position either absolutely or relatively phase-synchronous. The master axis position can be specified either by an encoder axis (real master axis), a virtual axis (virtual master axis) or a real axis.

The objective is a phase-synchronous and velocity-synchronous (drift-free) run between the master axis and the selected slave axis.

Typical use cases:

- Machining units for stamping, embossing, perforating ...
- Printing cylinders
- Materials transportation ...

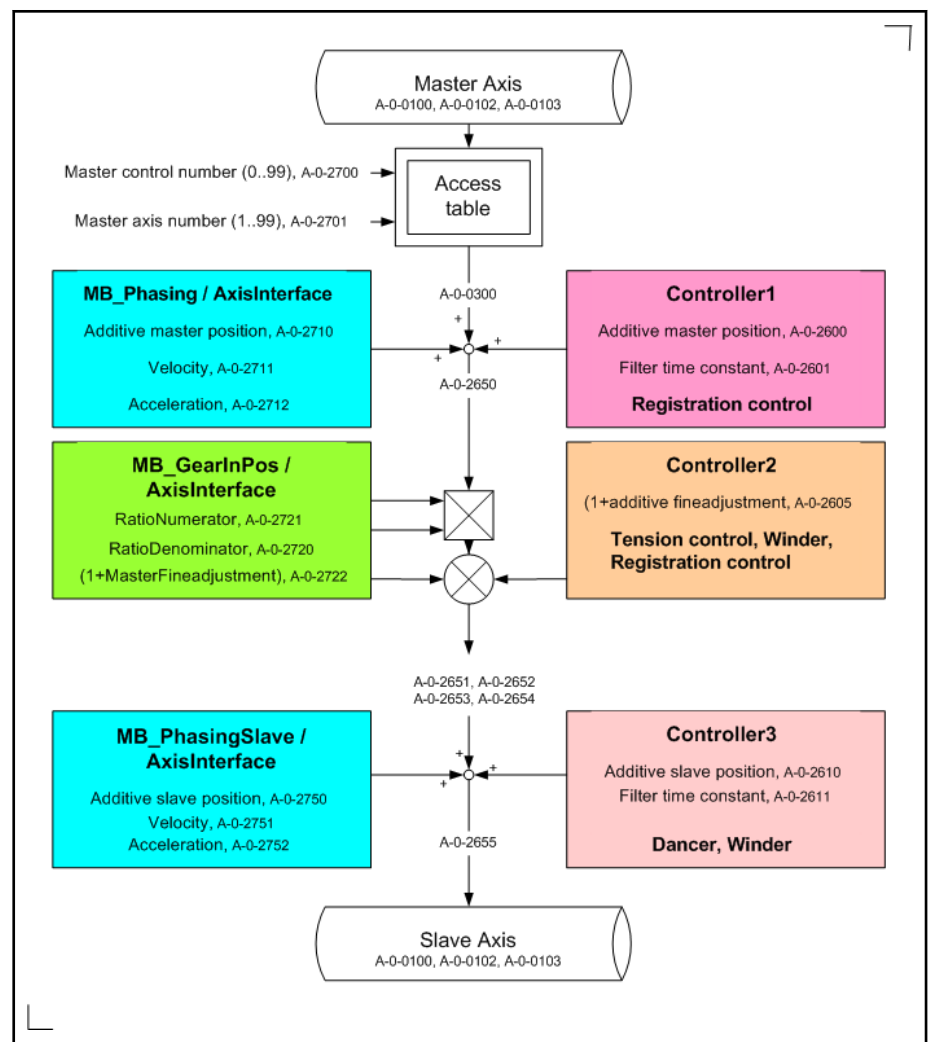


Fig. 9-36: MotionControl functional chain: Phase synchronization

The operation mode provides the following access points:

- The additive master axis position defined by "A-0-2710, Master axis, additive position command value", can be specified via MB_Phasing or the AxisInterface
- Electronic gear defined by the ratio of "A-0-2721, Master drive gear output revolutions" and "A-0-2720, Master drive gear input revolutions" and the "A-0-2722, Gear fine adjustment". The fine adjustment value is specified in percent and calculated as $(1 + \text{fine adjustment})$. These values are specified via MB_GearInPos or AxisInterface.
- The additive slave axis position defined by "A-0-2750, Slave axis, additive position command value" can be specified via MB_PhasingSlave or the AxisInterface

Electronic cam

In the "electronic cam" mode, the axis follows a specified master axis position defined by a fixed relationship between the master axis position and the slave axis. This relationship is defined by points in a cam table. The master axis position can be specified either by an encoder axis (real master axis), a virtual axis (virtual master axis) or a real axis.

The objective is to implement the dependency between the master axis and the selected slave axis. This dependency is stored in the cam table.

Motion Functionality

Typical use cases:

- CrossCutter
- FlyingShear
- Cycle-synchronous lock-on and lock-off ...

The cam tables can be generated and loaded using the "CamBuilder" integrated in IndraWorks. Moreover, PLC function blocks are provided by IndraMotion to generate cam tables for specific applications such as Cross-Cutter and CrossSealer.



The "electronic cam" for real axes is only available for interpolation in the drive.

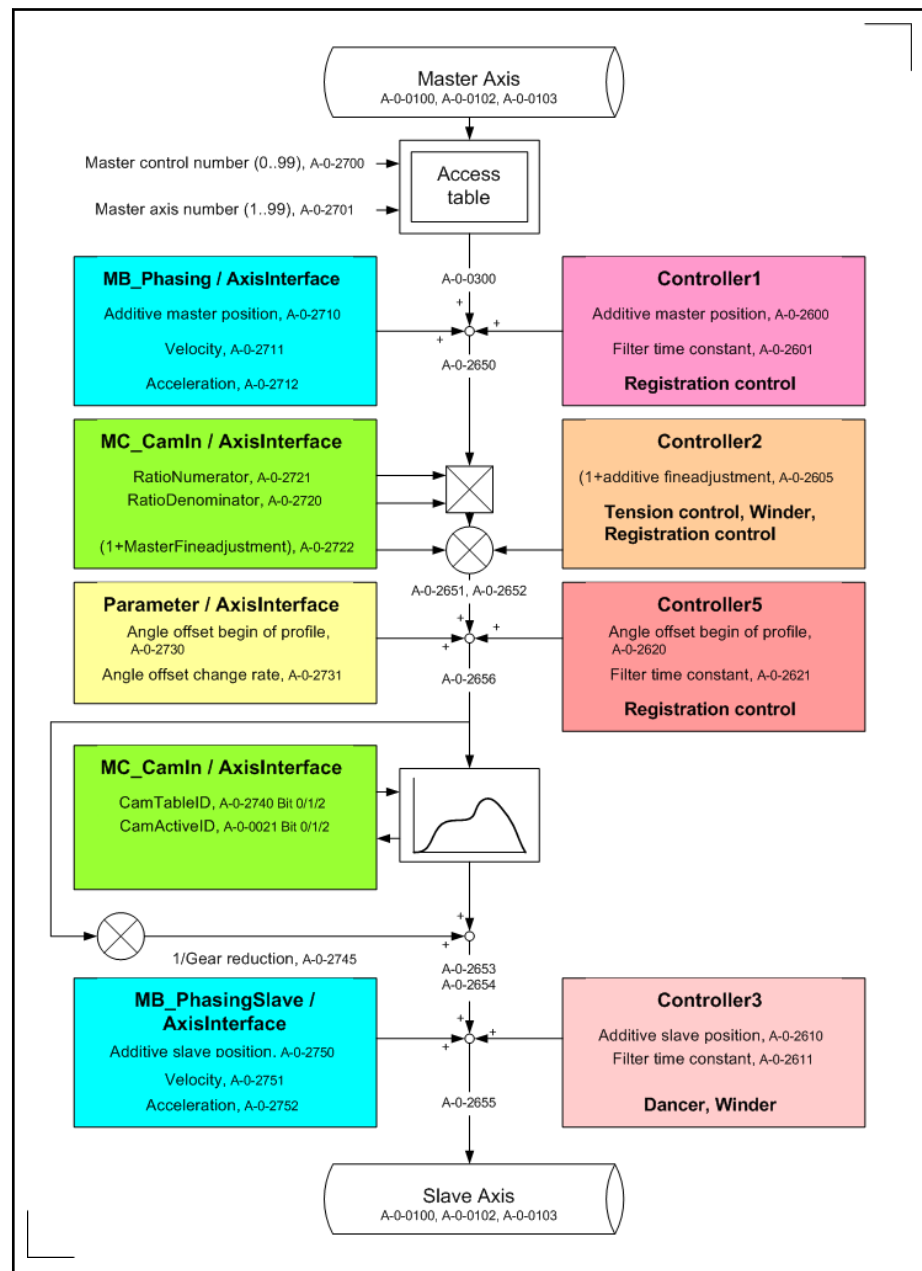


Fig.9-37: MotionControl functional chain: Electronic cam

The operation mode provides the following access points:

- The additive master axis position defined by "A-0-2710, Master axis, additive position command value", can be specified via MB_Phasing or the AxisInterface
- Electronic gear defined by the ratio of "A-0-2721, Master drive gear output revolutions" and "A-0-2720, Master drive gear input revolutions" and the "A-0-2722, Gear fine adjustment". The fine adjustment value is specified in percent and calculated as $(1 + \text{fine adjustment})$.
The selection of the two possible cam tables to be previously stored in the drive: "A-0-2740, Cam preselection"
These values are specified via MC_CamIn or the AxisInterface.
- The additive slave axis position defined by "A-0-2750, Slave axis, additive position command value" can be specified via MB_PhasingSlave or the AxisInterface

Electronic FlexProfile MotionProfile

In the "FlexProfile" mode, the axis follows a specified master axis position defined as analytic functions by mathematical motion laws. The master axis position can be specified either by an encoder axis (real master axis), a virtual axis (virtual master axis) or a real axis.

The objective is to implement a dependency between the master axis and the selected slave axis. This dependency is stored in the FlexProfile.

Typical use cases:

- Time-controlled profiles for thermoforming machines
- Sealing axis with bagging machines
- FlyingShear/bottle fillers, ...

FlexProfiles can be generated and loaded with the "CamBuilder" integrated in IndraWorks.



"FlexProfile" on the IndraMotion MLC are only provided for virtual axes and real axes with interpolation in the control.

Motion Functionality

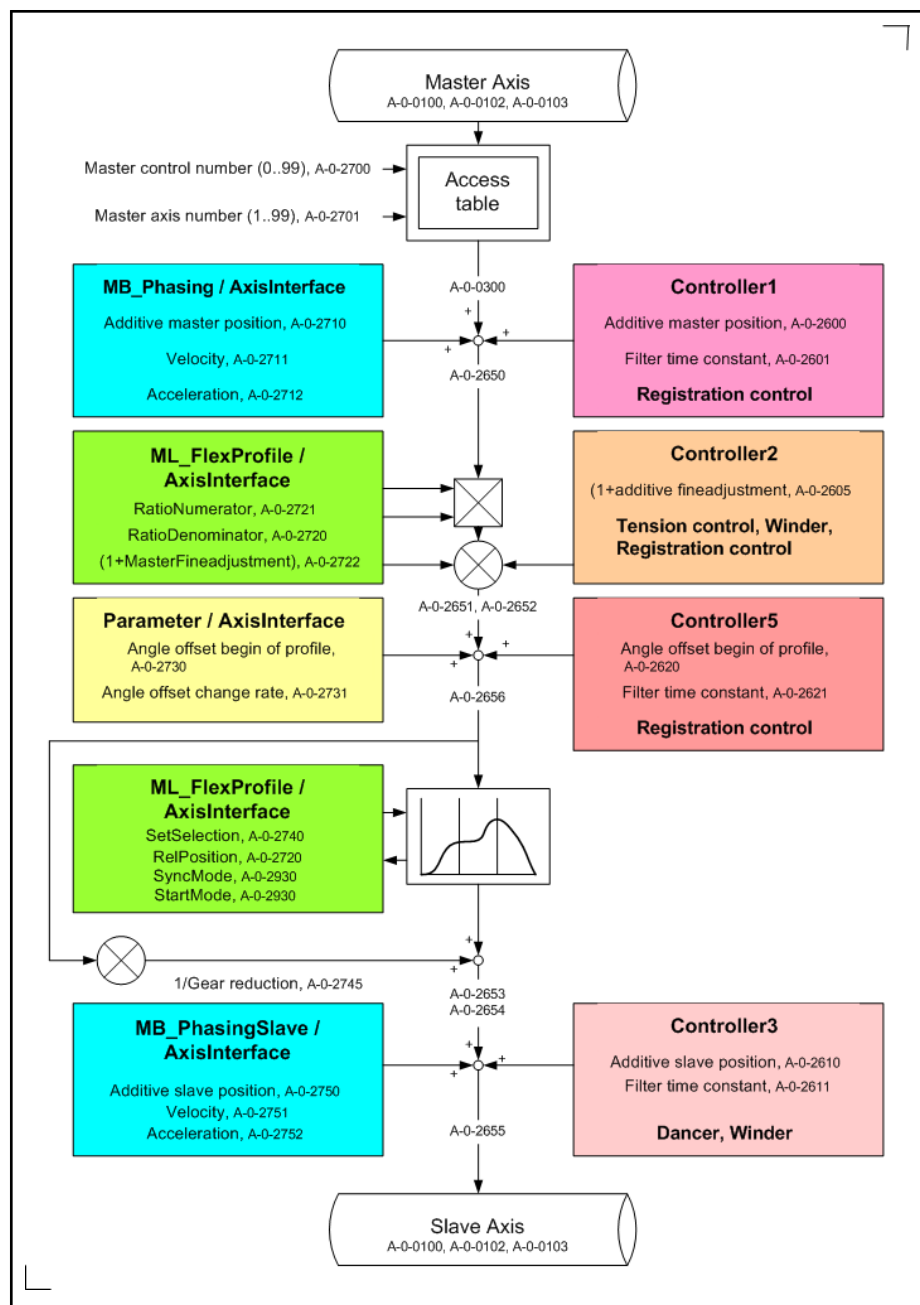


Fig.9-38: MotionControl functional chain: Electronic Motion Profile FlexProfile

The operation mode provides the following access points:

- The additive master axis position defined by "A-0-2710, Master axis, additive position command value", can be specified via MB_Phasing or the AxisInterface
- Electronic gear defined by the ratio of "A-0-2721, Master drive gear output revolutions" and "A-0-2720, Master drive gear input revolutions" and the "A-0-2722, Gear fine adjustment". The fine adjustment value is specified in percent and calculated as $(1 + \text{fine adjustment})$.

The selection of one of the two possible FlexProfile to be saved in the control before: "A-0-2740, Cam preselection".

These values are specified via ML_Flexprofile or the AxisInterface.

- The additive slave axis position defined by "A-0-2750, Slave axis, additive position command value" can be specified via MB_PhasingSlave or the AxisInterface
- The configuration of the "FlexProfile" mode is comprehensive and described in the documentation for ML_FlexProfile.

9.8 Validating Motion Configuration

9.8.1 Overview

When switching online or switching to the "offline parameterization mode", IndraWorks checks the entered Motion configuration automatically for projection errors. The result of this validation is displayed in the task list as error and warning messages. It is validated in several steps: First, it is validated offline. An error in this validation aborts the complete action. Otherwise, it is switched to "online" or "offline parameterization mode". Subsequently, it is validated online.

When switching online or switching to the "offline parameterization mode", IndraWorks checks the entered Motion configuration automatically for projection errors. The result of this validation is displayed in the task list as error and warning messages. It is validated in several steps: First, it is validated offline. An error in this validation aborts the complete action. Otherwise, it is switched to "online" or "offline parameterization mode". Subsequently, it is validated online (see also "Validating project data" in /5/, [Required and Supplementing Documentations, page 14](#)).

The offline validation checks the data configured on the PC and includes the following checks:

- Name checks for controls, axes, kinematics, etc.
- Consistency checks for axis and kinematic numbers as well as sercos addresses
- Check for maximum number of axes, kinematics, function modules, etc. allowed.
- Consistency checks for function modules

The online validation checks the data consistency in the control and includes the following checks:

- Check for the function packages of the connected drives
- Check for cross communication configuration of the control
- Check for axis parameter consistency



The online validation does not perform some checks of the cross communication configuration in the "offline parameterization mode".

Validation can be started manually at any time via the context menu of the control. The extent of validation depends on the connection state. The offline validation is always performed. If the control is switched online or the "offline parameterization mode" is enabled, it is also validated online.

Outputting of warnings while "validating" can be enabled or disabled via the XLC/MLC "Options" dialog under **Tools** ▶ **Options...** ▶ **XLC / MLC** "Switch Online / Offline" under "Message visibility" (see [fig. 9-11 "IndraWorks menu item Tools/Options" on page 356](#)).

Motion Functionality

For a better identification, each error and warning message in the task list is assigned to one unique designation. This designation can be used to search for the message in the online help for example. The designation format is.

MLs-<Type><Digit><Class><Number>.

The meanings of the individual fields is as follows:

- <Type>: Message type
 - E: It is an error message (Error)
 - W: It is a warning message
- <Digit>: This field is reserved and always contains the value 0
- <Class>: This field indicates the type of object affected. The following values are defined:
 - A: The message refers to an axis
 - C: The message refers to a control (controller)
 - F: The message refers to a function module
 - L: The message refers to the cross communication (link configuration)
 - P: The message refers to the complete project
 - K: The message refers to a kinematics
 - S: The message refers to a sercos device
 - X: The message has no special object reference
 - Y: The message refers to an object with safety engineering
- <Number>: A four-digit number

Example:

The message MLs-E0A0001 is an error message for an axis. The message MLs-W0S0013 is a warning for a sercos device.

9.8.2 Error Messages

General Information

Error messages refer to fatal input errors. Thus, the current Motion configuration cannot be used to switch online.

An example of a fatal error is the use of an axis name that is not IEC-compliant, since the axis can then not be accessed in the PLC program. An IEC-compliant name may only comprise letters, digits and the underscore. The first character of the name may not be a digit. Additionally, two underscores may not directly follow each other and the last character of the name may not be an underscore.

Examples of IEC-compliant names: ABCD, _ABCD, ABCD1, A_B_C_D, ABCD_1

Examples of non-IEC-compliant names: 1ABCD, __ABCD, AB__CD, ABCD_

Another example of a fatal error are names that do not comply with the RCL naming convention, but used in a kinematics. An RCL-compliant name may only comprise letters, digits and the underscore and the first character of the name may not be a letter. Additionally, the maximum length of such a name is limited to twelve characters. The name of an axis used in a kinematics has thus to comply with the RCL naming conventions as well.

Axis Error Messages

MLs-E0A0001: The axis name '<Name>' is not IEC-compliant

Cause The axis name specified does not comply with the naming conventions of the IEC-61131 standard. To reference an axis in a PLC program, IndraWorks automatically creates a global variable with the name of the axis. Thus, the axis name has to be IEC-compliant.

Solution

- Change the axis name to an IEC-compliant name

MLs-E0A0002: The axis name '<Name>' below the control is not unique

Cause The same name was specified for two (or more) axes below one control. To uniquely identify an axis, its name may only be specified once in one control.

Solution

- Rename one of the two axes
- Delete one of the two axes

MLs-E0A0003: The axis name <Name> is identical to an IndraLogic keyword

Cause An axis name has been specified that is identical to an IndraLogic keyword (e.g. "INT" or "ELSE"). To reference an axis in a PLC program, its name may not be identical to an IndraWorks keyword.

Solution

- Modify the axis name so that it is not identical to an IndraLogic keyword anymore
- Delete the axis

MLs-E0A0004: The axis <Name> and a kinematics have the same name

Cause The same name was specified for an axis and a kinematics. Since axes and kinematics are declared as global variables in the PLC program, their names have to differ.

Solution

- Rename the axis
- Rename the kinematics
- Delete the axis
- Delete the kinematics

MLs-E0A0005: The axis <Name> and a drive have the same name

Cause The same name was specified for an axis and a multiple-axis drive. Since this might cause misunderstandings, the names have to differ.

Solution

- Rename the axis
- Rename the drive
- Delete the axis
- Delete the drive

MLs-E0A0010: The number of projected axes (<Digit1>) is greater than the maximum permissible number of axes (<Digit2>)

Cause The number of axes projected exceeds the number the control can process. The maximum number of supported axes is written in <Digit2>.

Solution

- Delete the surplus axes
- Convert the control to a control type of higher performance

Motion Functionality

MLs-E0A0012: The axis number <Digit> is invalid

Cause An axis was specified with a number higher than supported by the control.

- Solution**
- Delete the axis
 - Convert the control to a control type of higher performance

MLs-E0A0013: The axis number <Digit> below the control is not unique

Cause The same number was specified for two (or more) axes below one control. To uniquely identify an axis, its number may only be assigned once in one control.

- Solution**
- Change the axis number of an axis to a unique number in the "Properties" dialog
 - Delete one of the two axes

MLs-E0A0014: The axis name <Name>' is longer than permitted (maximum <Digit> character

Cause The specified axis name is too long. The maximum length is written in <Digit>.

- Solution**
- Enter a shorter name for the axis

MLs-E0A0015: The axis identification <Unique ID> below the control is not unique

Cause The unique ID of an axis exists multiple times below a control. This indicates a corrupt project.

- Solution**
- Export all corresponding axes and import them again. When switching online again, the parameters of the axes can also be backed up

MLs-E0A0020: The encoder axis <Name1>' is assigned to the real axis <Name2>' This axis does not support encoder axes.

Cause The encoder axis <Name1> was assigned to the real axis <Name2>. This real axis does not support any encoder axes.

- Solution**
- Assign a real axis supporting encoder axes to the encoder axis
 - Delete the encoder axis

MLs-E0A0021: The encoder axis <Name> is not assigned to a real axis

Cause The encoder axis was not assigned to a real axis or the assigned real axis was deleted.

- Solution**
- Assign a real axis to the encoder axis
 - Delete the encoder axis

MLs-E0A0022: The axis <Name> does not match with the projected sercos protocol

Cause A drive was configured, but it does not match with the set sercos protocol, e.g. an EcoDrive was projected for sercos III even though it requires a sercos II master communication. The sercos protocol is indirectly set via the function module configuration, e.g. via the configuration of the sercos II function module.

- Solution**
- Adjust the function module configuration for the master communication
 - Delete the drive

MLs-E0A0023: The offline configuration of the drive <Name> does not match with the configured sercos protocol.

Cause Offline configuration data was created for a drive and the protocol of the master communication (e.g. by deleting or adding a "sercos II" function module) was changed. Thus, the configured control section does not match with the sercos master communication anymore and possibly existing offline parameters become invalid.

Solution

- Modify the master communication until it matches with the projected drives
- Open the properties of the drive and change the control section
- Delete the drive

MLs-E0A0030: For this axis, the control only supports the basic package "Closed Loop"

Cause The "Open Loop" basic package was configured for a real axis, but the control firmware does not support this basic package. In some cases, the cause can also be that the basic package is not supported by the drive.

Solution

- Upgrade the control firmware supporting the "Open Loop" basic package
- Replace the drive with a drive supporting the "Open Loop" basic package
- Replace the drive with a drive with the basic package "Closed Loop"

MLs-E0A0100: A conversion error was determined , thus causing missing internal axis data

Cause This project was transferred from an earlier IndraWorks (08VRS, 09VRS), but errors occurred during the conversion.

Solution

- Open the properties of the respective control and click **OK**. This corrects the internal data of all axes. Note that all parameters of the respective axes are reset to their default values when switching online next time. But the option to export the parameters before is provided

Control Error Messages**MLs-E0C0001: The control name <Name> is not IEC-compliant**

Cause The control name specified does not comply with the naming conventions of the IEC-61131 standard.

Solution

- Change the control name to an IEC-compliant name

MLs-E0C0002: The control name <Name> in the project is not unique

Cause The same name was specified for two (or more) controls in the project. To uniquely identify a control, its name may only be specified once in the project.

Solution

- Rename one of the two controls
- Delete one of the two controls

MLs-E0C0003: The hardware "VEP" is not supported as control by IndraWorks

Cause An older project was opened. This project contains a control of type "IndraMotion MLP VEP" or "IndraLogic XLC VEP". These controls are not supported by IndraWorks anymore for 12VRS or higher.

Solution

- Convert the control to a supported control type
- Delete the control

Motion Functionality

MLs-E0C0004: A control firmware of the version {VersionNumber} is not supported by this IndraWorks version

Cause An older project was opened, in which a firmware - not supported anymore by the IndraWorks version - is entered for the control.

- Solution**
- Change the firmware in the "Properties" dialog of the control to a supported version and load the firmware to the control
 - Open the project with a firmware suitable for the IndraWorks version

MLs-E0C0005: The device <Hardware> (<Version>) is not installed

Cause The project contains a control for which no device description is installed. This indicates that either the project was opened with a wrong IndraWorks version or the IndraWorks installation is faulty.

- Solution**
- Open the project with a suitable for the IndraWorks version or
 - repair the IndraWorks installation

MLs-E0C0006: The firmware release <Firmware> is not supported

Cause The project contains a control whose firmware release is not supported. This can be the case if the control was created with an IndraWorks test version and is now edited with an officially released IndraWorks version.

- Solution**
- Change the firmware release to a supported release in the "Properties" dialog of the control. Remember to update the firmware in the control (firmware download)

MLs-E0C0010: The IP address "<IP address>" of the control is not unique in the project

Cause The same IP address was assigned to two (or more) controls in the project. This can cause access conflicts while establishing a connection (logging in or switching online). Thus, one IP address may only be specified once in a project.

- Solution**
- Change the IP address of one of the two controls
 - Delete one of the two controls

MLs-E0C0014: The control name <Name> is longer than permitted (maximum <Digit> character)

Cause The specified control name is too long. The maximum length is written in <Digit>.

- Solution**
- Enter a shorter name for the control

Error Messages of the Function Modules**MLs-E0F0010: The number of projected function modules (<Digit1>) is greater than the maximum permissible number of function modules (<Digit2>)**

Cause The number of projected function modules exceeds the number the control can process. The maximum number of supported function modules is written in <Digit2>.

- Solution**
- Delete the surplus function modules
 - Convert the control to a control type of higher performance

MLs-E0F0011: More than the valid number (<Digit>) of function modules of type <Name> were projected

Cause More function modules of type "name" were created than allowed for the current control type. The maximum number allowed is written in <digit>.

- Solution**
- Delete the surplus function modules
 - Convert the control to a control type supporting more function modules of this type

MLs-E0F0012: The slot <Digit> is not valid for the function module <Name>

Cause A invalid slot was assigned to a function value.

- Solution**
- Move the function module to a valid slot. For more detailed information on the special features, refer to the description of the function module.

MLs-E0F0020: The function module <Name> is not valid for the current control type

Cause A function module not supporting by the current control type was projected.

- Solution**
- Delete the function module
 - Convert the control to a control type supporting the function module

Error Messages of the Kinematics

MLs-E0K0001: The kinematic name <Name> is not IEC-compliant

Cause The kinematic name specified does not comply with the naming conventions of the IEC-61131 standard. To reference an axis in a PLC program, IndraWorks automatically creates a global variable with the name of the kinematics. Thus, the kinematic name has to be IEC-compliant.

- Solution**
- Change the kinematic name to an IEC-compliant name.

MLs-E0K0002: The kinematic name <Name> below the control is not unique

Cause The same name was specified for two (or more) kinematics below one control. To uniquely identify a kinematics, its name may only be specified once in one control.

- Solution**
- Rename one of the two kinematics
 - Delete one of the two kinematics

MLs-E0K0003: The kinematic name <Name> is identical to an IndraLogic keyword

Cause A kinematic name identical to an IndraLogic keyword (e.g. INT or ELSE) was specified. To reference a kinematics in a PLC program, its name may not be identical to an IndraWorks keyword.

- Solution**
- Modify the kinematic name so that it is not identical to a keyword anymore
 - Delete the kinematics

MLs-E0K0005: The RCL program name <Name> below the control is not unique

Cause The same name was specified for two (or more) RCL programs below one control. To uniquely identify an RCL program, its name may only be specified once in one control.

- Solution**
- Rename one of the two RCL programs
 - Delete one of the two RCL programs

Motion Functionality

MLs-E0K0006: The axis name <Name> does not comply with the RCL naming conventions

Cause An axis not complying with the RCL naming conventions is referenced in one kinematics. Thus, the axis cannot be used by the kinematics.

Solution

- Rename the axis
- Use another axis in the kinematics

MLs-E0K0010: The number of projected kinematics (<Digit1>) is greater than the maximum permissible number of kinematics (<digit2>).

Cause The projected number of kinematics exceeds the number the control can process. The maximum number of supported kinematics is written in <Digit2>.

Solution

- Delete the surplus kinematics
- Convert the control to a control type of higher performance

MLs-E0K0012: The kinematic number <Digit> is invalid

Cause A kinematics was specified with a number higher than supported by the control.

Solution

- Delete the kinematics
- Convert the control to a control type of higher performance

MLs-E0K0013: The kinematic number <Digit> below the control is not unique

Cause The same number was specified for two (or more) kinematics below one control. To uniquely identify a kinematics, its number may only be specified once in one control.

Solution

- Delete one of the two kinematics

MLs-E0K0020: The axis assignment to position <Digit> is missing under the properties of the kinematics

Cause An axis that does not exist, since it has already been deleted, is referenced in a kinematics. <Digit> indicates the respective line in the axis assignment of the kinematics.

Solution

- Correct the axis reference in the kinematics
- Delete the kinematics

MLs-E0K0021: The first axis has to be configured as positioning axis

Cause The first axis of the kinematics was not configured as positioning axis.

Solution

- Configure the first axis of the kinematics as positioning axis
- Delete the kinematics

MLs-E0K0022: No kinematics was configured with the number <Digit>

Cause The kinematic numbers are not gaplessly ascending. There are gaps in between. It is for example not permitted to configure only the kinematic numbers 1 and 4, since the numbers 2 and 3 are then missing.

Solution

- Add a new kinematics with the missing number
- Delete all kinematics with a number higher than the digit if this one is not required

MLs-E0K0023: The axis <Name> is not unique in the kinematics

Cause The same address is referenced several times in the axis assignment of the kinematics. This is invalid.

- Solution**
- Modify the axis references in the kinematics until they are unique
 - Delete all ambiguous axis references

MLs-E0K0024: Belt axes <Digit1> were configured. Only a maximum of <Digit2> belt axes is permitted.

Cause More belt axes than allowed were configured in the kinematics.

- Solution**
- Delete the surplus belt axes or

MLs-E0K0025: Only <Digit1> positioning axes were configured. At least <Digit2> positioning axes are required for the kinematics

Cause Less positioning axes than required for this kinematic type were configured in one kinematics.

- Solution**
- Configure at least the number of positioning axes specified in <Digit2>
 - Change the kinematic type

MLs-E0K0026: Only <Digit1> axes were configured. At least <Digit2> axes are required for the kinematics

Cause Less axes than required for this kinematic type were configured in one kinematics.

- Solution**
- Configure at least the number of axes specified in <Digit2>
 - Change the kinematic type

MLs-E0K0027: The axis on position <Digit> cannot be configured after an empty axis

Cause A further axis was configured after an empty axis (axis type stands on "---") in a kinematics. This is invalid. <Digit> indicates the respective line in the axis assignment of the kinematics.

- Solution**
- Ensure that there are no empty elements between the individual axis references in the axis assignment of the kinematics

MLs-E0K0028: The positioning axis on position <Digit> cannot be configured after an orientation axis.

Cause A positioning axis was configured after an orientation axis in the axis assignment of the kinematics. This is invalid. <Digit> indicates the respective line in the axis assignment of the kinematics.

- Solution**
- Ensure that all orientation axes are configured after the positioning axes

MLs-E0K0029: The axis on position <Digit> is no belt axis and can thus not be configured after a belt axis.

Cause In the axis assignment of the kinematics, another axis, which is no belt axis, was configured after the first belt axis. This is invalid. <Digit> indicates the respective line in the axis assignment of the kinematics.

- Solution**
- Move the axis in front of the belt axes
 - Change the axis type to "belt axis"

MLs-E0K0030: 0 has to be entered as control number for the axis on position <Digit>, as it is no belt axis.

Cause In the axis assignments of the kinematics, a control number not equal to 0 was entered while assigning an axis, even though this axis was not a belt ax-

Motion Functionality

is. Axis assignments with a control number not equal to 0 refer to an axis in the cross link and are only allowed for belt axes. <Digit> indicates the respective line in the axis assignment of the kinematics.

- Solution**
- Change the axis type to "belt axis"
 - Enter 0 as control number and select the configured axis

MLs-E0K0031: The axis <Name> is impermissibly used in several kinematics (<kinematic list>)

- Cause** The same axis is referenced in several kinematics. This is only allowed for belt axes.

- Solution**
- Ensure that the axis is only referenced in one kinematics

MLs-E0K0032: The axis <Name> is already used as belt axis in the kinematics (<KinematicList>)

- Cause** In the axis assignment of the kinematics, an axis was referenced that is also referenced as belt axis by other kinematics. This is invalid. An axis referenced as belt axis by a kinematics may only be referenced as belt axis by other kinematics as well.

- Solution**
- Change the axis reference to an axis that has not been used yet
 - Change the type of axis assignment to "belt axis"

MLs-E0K0033: The number of coordinates does not match the number of axes

- Cause** A number of axes was configured in the kinematics, but not the same number of coordinates. To reasonably operate a kinematics, the number of coordinates and the number of axes have to be equal.

- Solution**
- Adjust the number of coordinates to the number of axes
 - Adjust the number of axes to the number of coordinates

MLs-E0K0034: The coordinate name <Name> below the control is not unique

- Cause** The same name was specified for several coordinates in one kinematics. This is invalid.

- Solution**
- Specify unique names for the coordinates
 - Delete all ambiguous coordinates

MLs-E0K0035: The coordinate name <Name> does not comply with the RCL naming conventions

- Cause** A name was specified for the coordinate of a kinematics. This name does not comply with the RCL naming conventions. Thus, the coordinate cannot be used by the kinematics.

- Solution**
- Rename the coordinate
 - Delete the coordinate

MLs-E0K0036: The length of the coordinate name <name> is invalid, as at least <Digit> characters are required.

- Cause** The name specified for the coordinates of a kinematics is too short. Thus, the coordinate cannot be used by the kinematics.

- Solution**
- Specify a longer name for the coordinate
 - Delete the coordinate

MLs-E0K0037: The coordinate name <Name> is identical to an RCL keyword

Cause A name was specified for the coordinate of a kinematics. This name corresponds to an RCL keyword (e.g. "BEGIN") . Thus, the coordinate cannot be used by the kinematics.

Solution

- Rename the coordinate
- Delete the coordinate

MLs-E0K0038: No axis type was configured for the axis on position <Digit>

Cause An axis has been referenced in the axis assignment of the kinematics, but no axis type was entered.

Solution

- Enter the axis type
- Delete the axis assignment

MLs-E0K0039: The kinematic name <Name> does not comply with the RCL naming conventions

Cause A name was specified for a kinematics. This name does not comply with the RCL naming conventions. Thus, this kinematics cannot be used.

Solution

- Rename the kinematics
- Delete the kinematics

MLs-E0K0040: The kinematic name <Name> is identical to an RCL keyword

Cause A name was specified for a kinematics. This name corresponds to an RCL keyword (e.g. "BEGIN") . Thus, this kinematics cannot be used.

Solution

- Rename the kinematics
- Delete the kinematics

MLs-E0K0041: The axis name <Name> is identical to an RCL keyword

Cause An axis is referenced in a kinematics. The axis name corresponds to an RCL keyword (e.g. "BEGIN") . Thus, the axis cannot be used by the kinematics.

Solution

- Rename the axis
- Use another axis in the kinematics

MLs-E0K0042: The transformation number <Digit1> is smaller than the maximum valid value <Digit2>

Cause The value specified for the transformation number of a user-defined kinematics is too low. This causes a conflict between the internally used transformation numbers. Thus, the kinematics cannot be used.

Solution

- Specify a value higher than or equal to <Digit2> for the transformation number
- Delete the kinematics

MLs-E0K0043: The axis type <Type2> is invalid for the axis on the position <Number> (kinematic axis type <Type1>)

Cause An axis - whose type does not fit to the kinematic axis type - is referenced in a kinematics. Thus, it is not reasonable to use an encoder axis as positioning axis, since an encoder axis cannot be directly traversed.

Solution

- Change the kinematic axis type so that it fits with the referenced axis

Motion Functionality

- Reference an axis whose type fits with the kinematic axis type
- Delete the axis assignment

Error Messages of Cross Communication

MLs-E0L0012: The CrossComm sercos III control address <Digit> is invalid

Cause A control address was entered into the sercos III C2C configuration. That control address is not in the valid range.

Solution

- Change the control address in the sercos III C2C configuration

MLs-E0L0020: The control does not simultaneously support a sercos II and a sercos III cross communication

Cause A function module "CrossComm sercos II" and a function module "CrossComm sercos III" was projected for a control. Such a configuration is technically impossible.

Solution

- Delete the function module that is not required

MLs-E0L0021: A sercos III cross communication is only possible if the master communication also takes place via sercos III

Cause A function module "CrossComm sercos III" was configured for a control via sercos II, even though the master communication takes place via sercos II. Such a configuration is technically impossible.

Solution

- Delete the function module "CrossComm sercos III"
- Change the master communication to sercos III by deleting the sercos II function module

Error Messages of sercos Devices

MLs-E0S0010: The number of projected sercos devices (<Digit1>) is greater than the maximum permissible number of sercos devices (<Digit2>)

Cause An impermissibly high number of sercos devices was configured. The maximum number of sercos devices is written in <Digit2>.

Solution

- Delete the surplus sercos devices

MLs-E0S0012: The sercos address <Digit> is invalid

Cause An invalid high sercos address was configured for a sercos device.

Solution

- If it is a sercos III I/O device, change the sercos address in the "Configuration sercos Devices" dialog
- If it is a drive, change the sercos address in the "Configuration of sercos Devices" dialog or in the "Properties" dialog of the corresponding axis
- Delete the sercos device

MLs-E0S0013: The sercos address <Digit> below the control is not unique

Cause The same address <Digit> was assigned to several sercos devices. Thus, the sercos bus cannot be switched to BB.

Solution

- If it is a sercos III I/O device, change the sercos addresses in the participating devices in the "Configuration sercos Devices" dialog
- If it is a drive, change the sercos addresses of the involved devices in the "Configuration of sercos Devices" dialog or in the "Properties" dialog of the corresponding axis

- Delete the surplus sercos devices

MLs-E0S0014: The topology address <Digit> below the control is not unique

Cause The same topology address was assigned to several sercos devices. In practice, such a configuration is not possible. It can result if several sercos devices are imported from another project.

Solution

- Open the "Configuration sercos Devices" dialog and correct the topology addresses

MLs-E0S0022: The I/O device <Name> does not match the projected sercos protocol

Cause A sercos III I/O device was projected, even though the master communication is set to sercos II.

Solution

- Delete the I/O device
- Delete the sercos II function module if a sercos III master communication is to be set

9.8.3 Information Messages

General Information

Information messages provide notes on the validation. They are used to point on special states or options and not on an error condition.

Non-Object-Related Information Messages

MLs-I0X0001: The following error <Identification> is temporarily only considered as warning.

Cause After contacting the support, a special setting was made in IndraWorks. Due to that setting, the following error is only considered as warning. This setting is only permitted to continue operating (e.g. switching online) if the validation incorrectly detected an error. This setting is to be undone if it is upgraded to a higher IndraWorks release.

Solution

- Upgrade to a higher IndraWorks release and undo the special setting

MLs-I0X0002: An internal error was detected during validation. Display details by double-clicking on the error.

Cause An internal error occurred during validation impeding some checks to be performed. This information indicates a potential problem with unknown effects. Error details are to be reported to the Bosch Rexroth Service.

Solution

- Report the error details to the Bosch Rexroth Service

9.8.4 Warning Messages

General Information

Warning messages indicate a potential problem in the Motion configuration. Still, it can be switched online.

Axis Warning Messages

MLs-W0A0020: The configured function package of the axis <Name> is not enabled in the corresponding drive

Cause An axis was configured so that a certain function package is required in the drive. This function package is not in the drive. The most frequent cause for this error is that the interpolation method "Drive" was selected for this axis or

Motion Functionality

the encoder axis was configured, but the function package "SNC Synchronization" is not enabled in the drive. Thus, the sercos bus cannot be switched to BB.

- Solution**
- Enable the required function package in the drive
 - Change the interpolation type of the axis in the "Properties" dialog or delete the possibly projected encoder axes

MLs-W0A0021: The projected axis name <Name1> does not match the axis name in the control <Name2>

- Cause** The axis name (parameter A-0-0002) set in the control does not correspond to the axis name projected in IndraWorks. Possible reasons for this difference are:

- The corresponding parameter in the drive (S-0-0142) is password-protected
- The parameter A-0-0002 was changed manually or by the PLC program after switching online

Normally, the axis name is updated in the control when switching online. To avoid comprehension problems, do not change the name in the control.

- Solution**
- Delete the drive password or allow write access on its parameter
 - Rename the axis so that its name differs from the one of the control
 - Ensure that the parameter A-0-0002 is not changed anymore

MLs-W0A0022: An encoder is configured in the drive <Name>, even though the function package "Open Loop" is set

- Cause** The "Open Loop" function package was set for an axis, but the corresponding drive configured an encoder. Thus, it is not possible to switch the sercos bus to phase 4.

- Solution**
- Parameterize the drive to ensure that the encoder is not used anymore
 - Replace the drive with a drive with the basic package "Closed Loop"

MLs-W0A1000: The controller type <Name> (ID: <Digit>) could not be loaded

- Cause** A project is open containing a controller axis, but whose type is not installed in IndraWorks. This can be the case if an IndraWorks project created on another computer is open.

- Solution**
- Install the controller axis type
 - Delete the controller axis

MLs-W0A1000: The controller type <Name> (ID: <Digit>) could not be loaded

- Cause** An IndraMotion MLC project is open containing a controller axis, but whose type is not installed in IndraWorks. This can be the case if an IndraMotion MLC project created on another computer is opened in IndraWorks.

- Solution**
- Install the controller axis type
 - Delete the controller axis

Warning Messages of the Control

MLs-W0C0020: A higher device version (<Version>) is available for the control

Cause A project is open that was created with an older, unofficial IndraWorks release and the device version of the control changed in the meantime. Thus, some new functions of the control cannot be used.

Solution

- Update the device in the properties of the control

Warning Messages of Function Modules

MLs-W0F0020: An SRAM function module was configured, even though the control is provided with an already integrated SRAM

Cause An SRAM function module was configured, even though the control is already provided with an integrated SRAM. Since always the internal SRAM is used in this case, this function module is not required.

Solution

- Delete the SRAM function module
- Use a control without integrated SRAM

MLs-W0F0021: If the control is switched off, kinematic data is lost as no SRAM function module is projected

Cause The kinematic data (RCL programs, workpiece coordinates, etc.), but no SRAM module were projected. Thus, the control is not able to save this data permanently. If the control is switched off and switched on again, the kinematic data is lost and has to be reloaded again to the control.

Solution

- Insert an SRAM function module
- Delete all kinematic data if this is not required anymore

MLs-W0F0022: An SRAM function module was projected, but no kinematic data

Cause An SRAM function module was projected, but no kinematic data (RCL-programs, workpiece coordinates, etc.). Since this function module is only used for the permanent storage of this data, it is not required.

Solution

- Delete the SRAM function module if it is not required
- Add kinematic data (RCL programs or workpiece coordinates) to the project

Warning Messages of the Kinematics

MLs-W0K0001: The axis <Name> is also used in user-defined kinematics (<KinematicList>)

Cause In the axis assignment of the kinematics, an axis has been referenced that is also referenced in one or several user-defined kinematics. This can cause problems.

Solution

- Change the axis reference to an axis that has not been used yet
- Delete the axis reference

Warning Messages of the Cross Communication

MLs-W0L0020: The produced axis with the number <Digit> is not projected

Cause A number of a produced axis was entered in the sercos III CrossComm configuration, This axis is not configured. This message is only for information

Motion Functionality

purposes. Phase switching the cross link and the master communication are still possible.



This check is only made by the offline validation if the control is configured as CrossComm master.

- Solution**
- Project the missing axis in IndraWorks
 - Delete the axis number from the list of the produced axes

MLs-W0L0022: The produced axis with the number <Digit> is not in the control

- Cause**
- A number of a produced axis was entered in the sercos III CrossComm configuration, This axis is not configured. This message is only for information purposes. Phase switching the cross link and the master communication are still possible.



This check is always made by the online validation irrespective of the control configuration as CrossComm master or slave.

- Solution**
- Project the missing axis in IndraWorks
 - Delete the axis number from the list of the produced axes

MLs-W0L0023: The Motion cycle time (<Digit1>ms), the sercos cycle time (<Digit2>ms) and the cross link cycle time (<Digit3>ms) differ

- Cause**
- A cross link was configured, but the cycle times for the Motion task, master communication and cross link differ. Thus, the sercos bus cannot be switched to BB.

- Solution**
- Set the same value for all cycle times. It can either be set via the parameter editor or the respective IndraWorks dialog
 - The cycle times for the Motion task (parameter C-0-0400) and the sercos bus (parameter C-0-0503) can be set in the "Cycle time" that can be opened via the context menu of the "Motion" node
 - The cycle time of the cross link (parameter C-0-0700) can be found in the configuration dialog of the respective function module

MLs-W0L0024: The link axis with the number <Digit> is not in the control

- Cause**
- A non-existent link axis was entered into the sercos II CrossComm configuration. This link axis is supposed to provide its axis position in the cross link. This message is only for information purposes. Phase switching the cross link and the master communication are still possible.

- Solution**
- Correct the axis specification in the settings of the function module CrossComm sercos II

MLs-W0L0026: The control address in the link (<Digit1>) is greater than the maximum permissible control address (<Digit2>).

- Cause**
- An invalid high control address was specified in the sercos II CrossComm configuration. The maximum valid control address <Digit2> depends on the desired cycle time of the cross link and is:

- 16, if the cycle time ≤ 2ms
- 32, if the cycle time > 2ms and if the cycle time ≤ 4ms
- 64, if the cycle time > 4ms

- Solution**
- Correct the control address in the settings of the function module "CrossComm sercos II"

- Reduce the cycle time of the cross link

MLs-W0L0027: Link axes were configured for the control. However, no function module of type "CrossComm sercos II" was configured

Cause Link axes were projected, but no function module "CrossComm sercos II". Thus, the link axes cannot be accessed.

- Solution**
- Add a "CrossComm sercos II" function module to the control
 - Delete link axes if they are not required anymore

Warning Messages of sercos Devices

MLs-W0S0013: Several devices with the sercos address <Digit> are connected to the control

Cause Several devices were found at the sercos bus. These devices are provided with the same sercos address. It cannot be switched to BB with this configuration.

Caution: This warning refers to the actually connected devices and **not** to the devices configured in IndraWorks.

- Solution**
- Set the sercos addresses of the devices unambiguously. For drives, the sercos address can be set directly at the device. For I/O devices, the address has to be set in the "Configuration sercos Devices" dialog

MLs-W0S0020: The device with the sercos address <Digit> is disabled even though it is connected to the control

Cause A sercos III device has been disabled in the control, even though it is actually connected to the control. Thus, the sercos bus cannot be switched to BB.

- Solution**
- Enable the device
 - Remove the device from the control

MLs-W0S0021: The device with the sercos address <Digit> is connected to the control, but it is not configured

Cause A device has been connected to the sercos bus, but it is not configured in IndraWorks. Thus, the sercos bus cannot be switched to BB.

- Solution**
- Project the device in IndraWorks and switch online again
 - Remove the device from the control

MLs-W0S0022: No sercos device was configured with the topology address <Digit>

Cause The projected topology addresses of the sercos devices are not gaplessly ascending. There are gaps in between. Such a configuration is impossible in practical applications. The error message refers to a detected gap.

- Solution**
- Open the "Configuration sercos Devices" dialog and correct the topology addresses

MLs-W0S0023: The Motion cycle time (<Digit1>ms) is no multiple of the sercos cycle time (<Digit2>ms)

Cause The cycle time for the Motion cycle has been set, but no multiple of the sercos cycle time. Thus, the sercos bus cannot be switched to BB.

- Solution**
- Correct the value for the cycle time of the Motion tasks (parameter C-0-0400) and the sercos bus (parameter C-0-0503) so that they fit. That means that the Motion cycle time is an integer multiple of the sercos cycle time. This can be performed via the parameter editor or the

Motion Functionality

"Cycle time" dialog, which can be opened via the context menu of the "Motion" node

9.9 Electronic Cams

9.9.1 General Information

Cams are used wherever complex, nonlinear motion takes place or where rotary and translatable slave motion sequences have to be generated depending on a given master axis.

In its basic functions, a cam gear generates the reference between a master axis and a slave axis using a freely designable transfer function. In this case, the reference signal is usually an angle of rotation " ϕ ". Using a cam, this is compiled into a new slave axis motion with the angle of rotation " ψ ". The figure below shows a simplified representation.

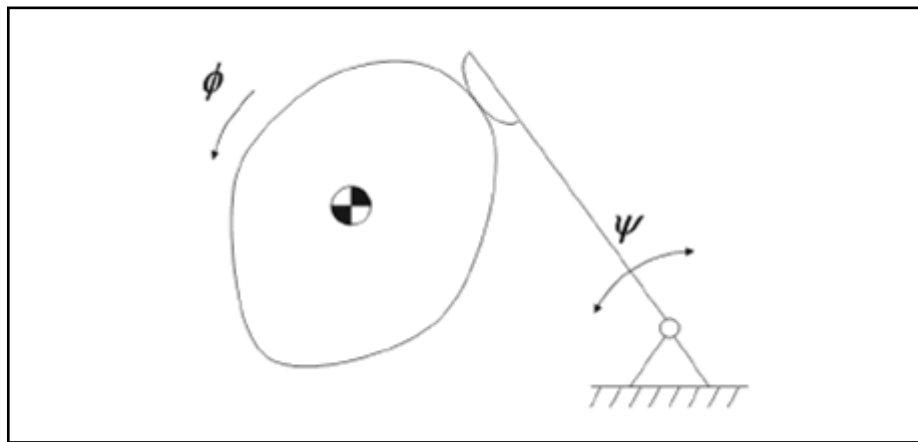


Fig.9-39: Block diagram of a general cam

In general, two cam solutions can be defined. These are the mechanical cam and the electronic cam solution. In the following, the purely mechanical cams are not discussed in detail. An electronic cam can be constructed either as a cam table or analytically as polynomial profiles (FlexProfile, MotionProfile).

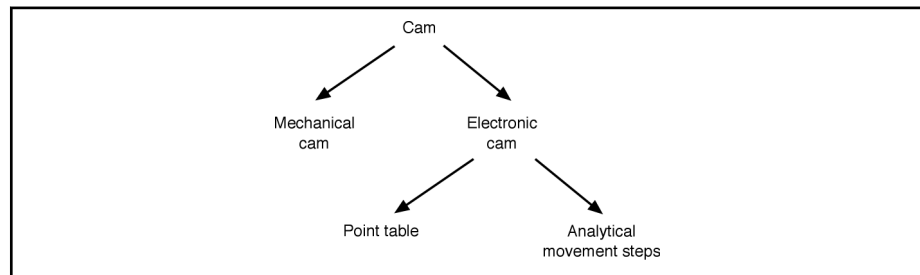


Fig.9-40: Basic cam solutions

The FlexProfile parameterization is executed via the dialog of the respective axis (see e.g. [FlexProfile](#), page 307).

An exemplary use case for the FlexProfile is described in /25/, [Required and Supplementing Documentations](#), page 14.

9.9.2 Cam Solutions

If existing electronic cam solutions are studied in terms of their functions, two basic implementation types can be distinguished:

Cam table In its simplest form, the curve profile which represents the coupling of master axis and slave axis typical for cams exists as curve profile defined by data

points. The slave axis position is determined by interpolation between the corresponding data points of the current interval. This is defined by the master axis position. Various interpolation procedures can be used, depending on the application.

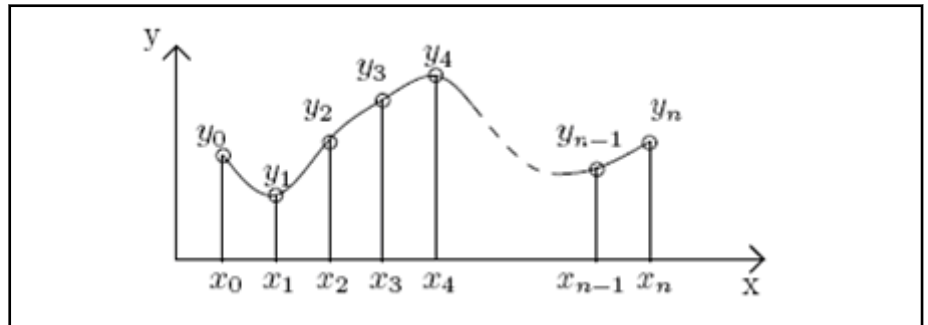


Fig. 9-41: Cam table with cubic compensation spline

Apart from the simple linear interpolation, spline functions are often used. If precise data points are available that should be traversed later, interpolating polynomial splines have to be used. The figure above shows an example of how to use a cubic spline to interpolate between the given data points. In order to be able to dynamically generate better curves, Hermite spline functions can be used while the cubic interpolation splines mentioned above are based on polynomials 5th order. These execute a curve approximation resulting in a smoother curve. Irrespective of the interpolation procedure used, this way of describing the cam is used very often in terms of processing in a servo controller.

The cam tables are provided with the following characteristics:

- Simple and easy mode of operation
- Low computing effort
 - All the coefficients required for spline interpolation are determined in a calculation made before the time of execution
- The slave axis position can be determined very easily at runtime
- Suitable for motion whose shape is not changed online
- All polynomial coefficients have to be calculated again after a data point was changed
- Large data volume

The cam tables can be configured either via the PLC function block **MB_ChangeCamData** or the IndraWorks interface.

Analytical motion laws

Apart from describing the cam exclusively with a cam table, it is possible to describe it using mathematical transfer functions. Here, these are called motion laws. Motion laws describe the relative motion of two gear members as analytical functions, in this case the output motion depending on the input motion. The entire cam profile is described in steps using various motion laws. Each step with the respective motion sequence, is called a "motion step". The figure below shows a cam profile that consists of three motion steps.

Motion Functionality

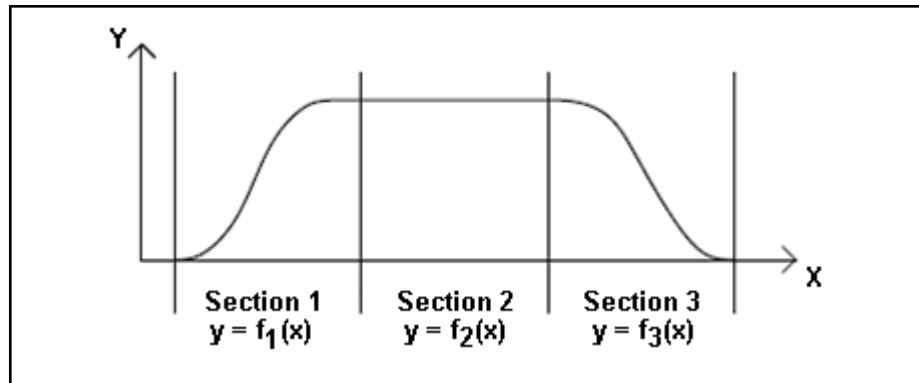


Fig.9-42: Cam profile consisting of three motion steps

Mainly polynomial functions of the 5th order or higher are used since these provide the corresponding degrees of freedom to specify velocity, acceleration and (if necessary) jerk at the start/end of the motion step. The individual motion steps can be put into sequence without impact or jerks by selecting identical limit values. Trigonometric motion laws such as different sine curves can be used as motion laws. Several of such motion laws are defined in the VDI 2143 standard. This solution has the following advantages:

- The cam profiles are described with little data volume. The required information is limited to only the specification of the boundary values of the individual motion steps
- Individual motion steps can be changed more easily and irrespective of the other motion steps. However, the price paid for these advantages is increased calculations at runtime. After all, the corresponding motion law has to be calculated for every point in time for which the slave axis position is to be determined

The analytical MotionProfiles can be configured either using PLC function blocks **MB_ChangeProfileSet** (for MotionProfile), **MB_ChangeFlexProfileSet** (for FlexProfile) or with the IndraWorks interface.

9.9.3 Electronic Cam

In the control, the typical coupling between the master axis and slave axis also results for an electronic cam (see figure below). In this context, often named master axis or slave axis or master and slave only. However, as opposed to mechanical cams, the coupling is not permanent. Virtually, each axis whose current actual position can be added to the cam table can be used as cam profile. It can also be a virtual master axis. This is an axis calculated on the control without physical counterpart. Its position can for example be used as a reference signal for a cam table, and thus indirectly control the motion of a slave axis.

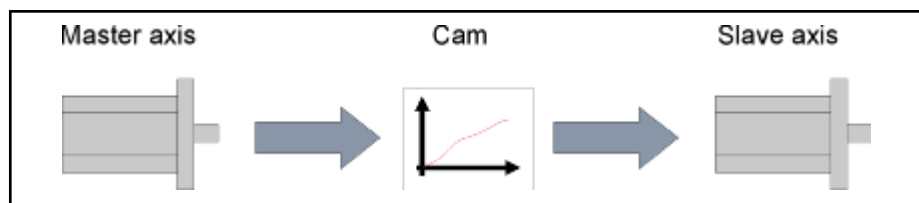


Fig.9-43: Signal flow and cam localization

9.9.4 Fields of Application

Typical fields of application of electronic cams include:

- Transforming a rotary motion into straight-line translational motions with a controllable course
- Transforming a rotary motion into an oscillating rotary or translational motion with a controllable course
- Generating master motions or mechanically synchronized relative motions of components or tools and controlling the resulting acceleration and velocity characteristics
- Linearization of physical or mechanical relationships or transformations

However, it cannot generally be specified to what degree gears that translate unequally represent a solution for a given application problem. As early as the planning stage of the application, this requires a close examination of the task as well as the necessary knowledge and mechatronic know-how of the designer. Due to its flexible ease of design and potential for optimization, the electronic cam represents a tool for a number of motion tasks

9.9.5 Centralized and Decentralized Cam Concept

The cam processing and the resulting command value generation can take place either centrally in a superordinate control or decentrally in the drives. Bosch Rexroth drive units allow the storage of several cam table profiles in the memory and the direct processing from a superordinate control or a PLC after the command was issued.

Apart from this decentral execution of the cam table operation in the IndraDrive family, the IndraMotion MLC control system also provide the option to execute the calculation/interpolation of the cam profile in the control and commanding the drives cyclically using a position command value interface.

9.9.6 FlexProfile – Basic Functionalities

General Information

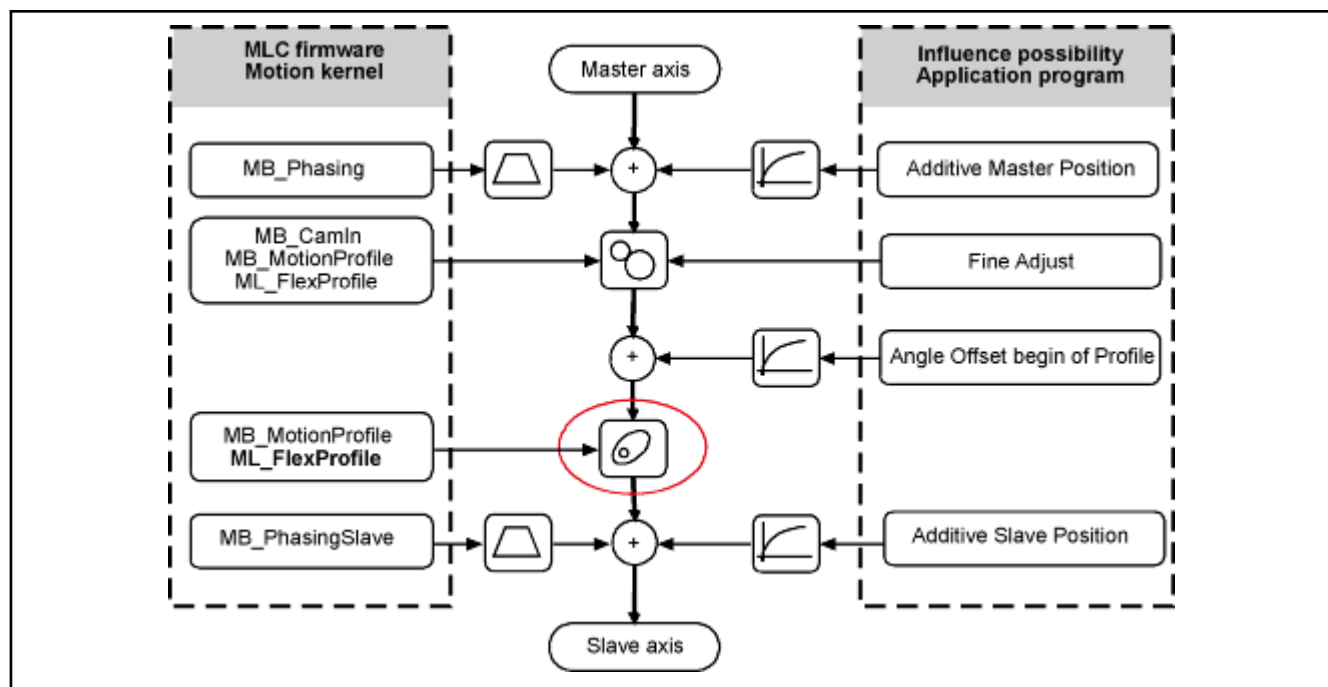
FlexProfile belongs to the group of electronic motion laws. These are described by mathematical motion laws as analytical functions. In addition to the functions described in [chapter 9.9.4 "Fields of Application" on page 398](#), the following properties can be selected in the FlexProfile:

- Time control
- Event Control
- Wide selection of motion laws, etc.

Basic Structure of Synchronous Operating States"FlexProfile"

In the "FlexProfile" operating state, , there is a firm relationship between the defined master axis position and the slave axis. The main procedural chain of this synchronization type also contains the gear and the FlexProfile that influence the relationship between the master axis and the slave axis. All axis types can be used as the master axis. In addition, there are several methods to affect the signal flow.

Motion Functionality

Fig.9-44: Synchronous operating states **ML_FlexProfile**

The FlexProfile can be started using the PLC FB **ML_FlexProfile**.

Profile Configuration

When creating a FlexProfile, note that the individual motion steps are defined using a relative master axis range and a relative distance (see the figure below). The motion sections defined in this manner are arranged in sequence. A profile created using this method has no absolute reference anymore. This can be regenerated by using a FlexStep.

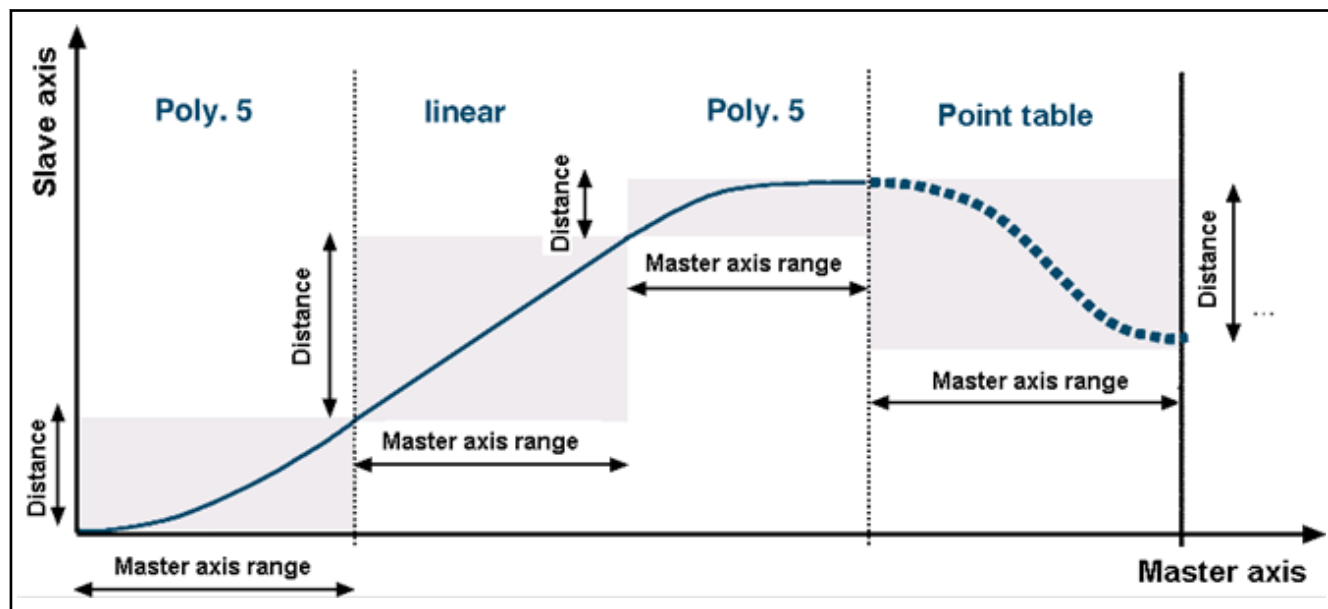


Fig.9-45: Configuring and creating the FlexProfile

The master axis range can be either axis or time (time control). The duration for processing a motion step or an entire profile can be entered directly. The scaling of the master axis and slave axis can be freely selected.

Time Master

In order to allow time-controlled sequences in a MotionProfile, this section shows how different "master sources" can be assigned to the individual motion steps of a MotionProfile. Now, not only one single master axis exists for all the motion steps for a cam profile, but a separate master source can already be specified during the generation of the MotionProfile/motion plan. If time as such is seen as a master source, time-controlled motions can also be implemented in this manner. For steps with such a time reference signal, the duration of the execution does not depend on the master axis velocity.

A Flex step ([Flex step, page 403](#)) has to be used to restore an absolute master axis range or to synchronize different master axis encoders.

Motion Laws

Default Motion Laws

The relationship between the master axis motion and the slave axis motion in the individual motion steps is specified by motion laws based on different standardized transfer functions.

According to VDI2143, the following four motion tasks can be specified in the boundaries of a motion step, depending on the velocity and acceleration:

Motion task	Abbreviation	Velocity	Acceleration
Rest	R	$v=0$	$a=0$
Constant velocity	G	$v \neq 0$	$a=0$
Reverse	U	$v=0$	$a \neq 0$
Motion	B	$v < > 0$	$a < > 0$

Fig.9-46: Motion tasks according to VDI 2143

Various motion laws can be classified accordingly in terms of their function courses possible boundary value adaptations. The following overview shows the most common standardized motion laws:

Rest→Rest

Standstill

Simple sine (simple sine curve)

Besthorn sine (inclined sine curve)

Acceleration-optimized inclined sine curve

Torque-optimized inclined sine curve

Gutman sinusoid

Modified sinusoid

Modified acceleration trapezoid

Polynomial 5th order

Polynomial 7th order

Velocity→Velocity

Linear interpolation

Polynomial 5th order

Polynomial 7th order

Modified sinusoid

Motion Functionality

Rest→Velocity

Polynomial 5th order

Polynomial 7th order

Velocity→Rest

Polynomial 5th order

Polynomial 7th order

General motion

Polynomial 5th order

Polynomial 7th order



A precise description of the motion laws can be found in the data type description of the enumeration data type MC_STEP_MODE.

Extended Motion Laws

Additional special motion laws (interpolation) for use in electronic MotionProfiles are now described. These improve the adaptation of the MotionProfile to the motion problem with different degrees of freedom.

In general, the boundary conditions such as:

- Distance
- Velocity
- Acceleration
- Jerk

are specified and the master axis range results (see the figure below).

The following extended motion laws are available:

- Acceleration-limited motion (trapezoid profile)
- Acceleration-limited sinusoid
- Jerk-limited motion
- Polynomial 5th order, limited velocity
- Overshooting-free polynomial 5th order
- Velocity-limited trapezoid



A precise description of the motion laws can be found in the data type description of the enumeration data type MC_STEP_MODE.

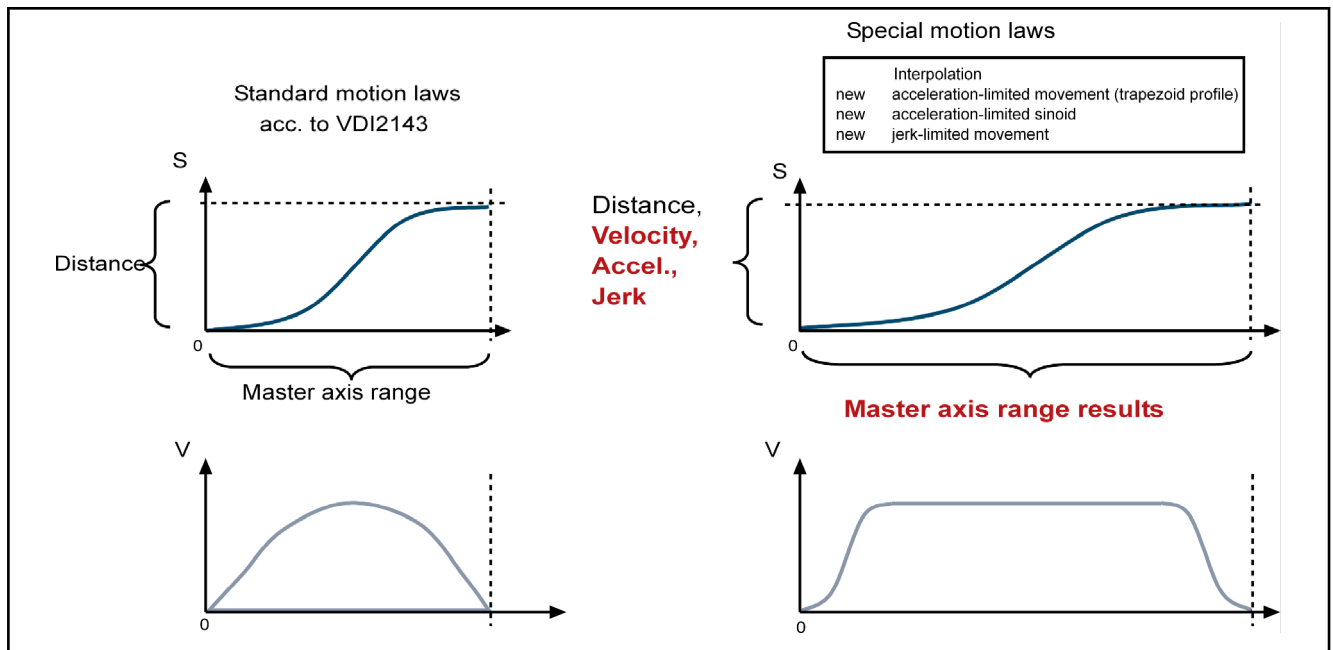


Fig.9-47: Properties of the special motion laws

Flex step

If a motion step is marked as a FlexStep, it can be considered as active motion step. Due to the boundary values specified previously, its motion path is corrected and adapted at runtime. Flex steps are used, among other things, to synchronize motion steps with different master axes (time axis and position axis).

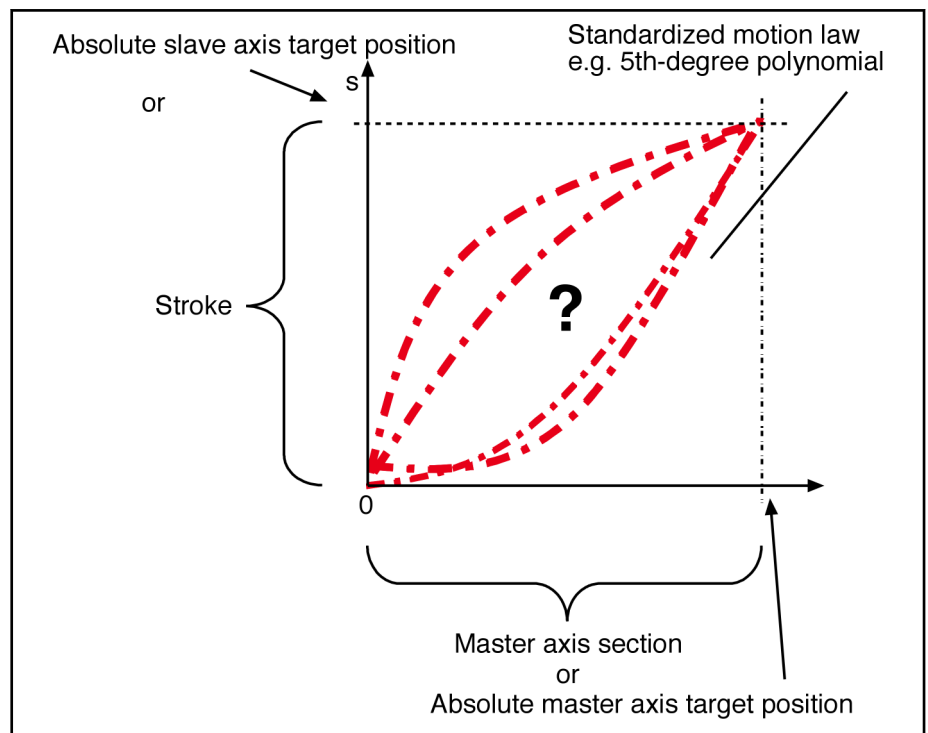


Fig.9-48: Configuration possibilities of the FlexStep

In a FlexStep, also specify - instead of the distance - an absolute slave axis target position to be reached and - instead of the master axis section - an absolute master axis end position to be reached (the distance and the reference

Motion Functionality

encoder section are calculated at runtime if necessary). The boundary values are adapted in such a way that velocity jumps are avoided.

In general, the FlexStep can be used to implement the following configurations:

- FlexStep with a relative distance and a relative master axis range
- FlexStep with an absolute distance and a relative master axis range
- FlexStep with a relative distance and an absolute master axis range
- FlexStep with an absolute distance and an absolute master axis range

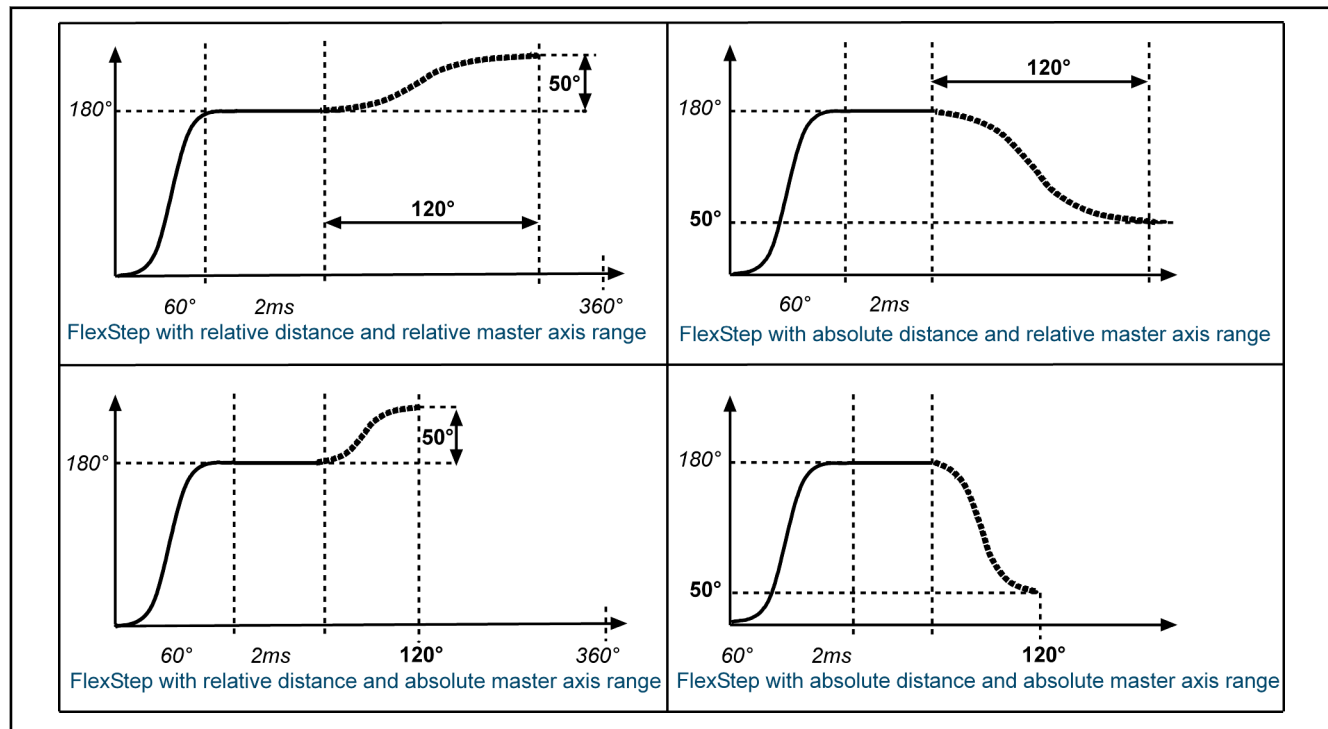


Fig. 9-49: Configuration of the FlexStep



The FlexStep can only be configured with the following motion laws:

- General motion, polynomial 5th order
- General motion, polynomial 7th order

A FlexStep has to be used after a time step.

Motion Law Boundary Conditions

	Definition of the motion law at the input "MotionLaw" (ProfileStepAdr)		Travel range		Boundary values left			Boundary value right			Max (not considered everywhere)			TPS possible	Flex-Step possible	Operation modes
			Distance	Range	StVel	StAcc	StJerk	EndVel	EndAcc	EndJerk	MaxVel	MaxAcc	MaxJerk			
Rest-in-rest																
Standstill	REST_IN_REST_LINEAR	2048	0	*	0	0	0	0	0	0	=	=	=	x		CT, FP
Sine curve	REST_IN_REST_SINE	2560	*	*	0	=	=	0	=	=	=	=	=	x		CT, FP
Inclined sine curve	REST_IN_REST_INCLINEDSINE	0	*	*	0	0	=	0	0	=	=	=	=	x		CT, MP, FP
Acc.-opt. inclined sine curve	REST_IN_REST_SINEACC	3072	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Torque-opt. inclined sine curve	REST_IN_REST_SINETORQUE	3328	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Gutman sinoide	REST_IN_REST_GUTMANSINE	2816	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Modified sine curve	REST_IN_REST_MOD_SINE	3840	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Modified acceleration trapezoid	REST_IN_REST_MOD_TRAPEZE	3584	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Polynomial 5th order	REST_IN_REST_POLY5	256	*	*	0	=	=	0	=	=	=	=	=	x		CT, FP
Polynomial 7th order	REST_IN_REST_POLY7	2304	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Polynomial 8th order	REST_IN_REST_POLY8	5120	*	*	0	0	0	0	0	0	=	=	=	x		CT, MP, FP
Quadratic parabola	REST_IN_REST_PARABOLA	4864	*	*	0	0	0	0	0	0	=	=	=	x		CT
Rest-in-velocity																
Polynomial 5th order	REST_IN_VELOCITY_POLY5	512	*	*	0	0	=	*	0	=	=	=	=			CT, MP, FP
Polynomial 7th order	REST_IN_VELOCITY_POLY7	768	*	*	0	0	0	*	0	0	=	=	=			CT, FP
Velocity-in-velocity																
Constant velocity	CONSTANT_VELOCITY	1536	*	*	=	0	0	=	0	0	=	=	=			CT, MP, FP

Motion Functionality

	Definition of the motion law at the input "MotionLaw" (ProfileStepAdr)		Travel range		Boundary values left			Boundary value right			Max (not considered everywhere)			TPS possible	Flex-Step possible	Operation modes
			Distance	Range	StVel	StAcc	StJerk	EndVel	EndAcc	EndJerk	MaxVel	MaxAcc	MaxJerk			
Polynomial 5th order	VELOCITY_IN_VELOCITY_POLY5	1792	*	*	*	0	=	*	0	=	=	=	=			CT, MP, FP
Polynomial 7th order	VELOCITY_IN_VELOCITY_POLY7	4096	*	*	*	0	0	*	0	0	=	=	=			CT, FP
Modified sine curve	VELOCITY_IN_VELOCITY_MOD_SINE	4352	*	*	*	0	=	*	0	=	=	=	=			CT, FP
Velocity-in-rest																
Polynomial 5th order	VELOCITY_IN_REST_POLY5	1024	*	*	*	0	=	0	0	=	=	=	=			CT, MP, FP
Polynomial 7th order	VELOCITY_IN_REST_POLY7	1280	*	*	*	0	0	0	0	0	=	=	=			CT, FP
General motion																
Polynomial 2nd order	COMMON_POLY2	16896	*	*	*	=	=	=	=	=	=	=	=			CT
Polynomial 3rd order	COMMON_POLY3	17152	*	*	*	=	=	*	=	=	=	=	=			CT
Polynomial 4th order	COMMON_POLY4	17408	*	*	*	=	=	*	=	=	=	=	=	x		CT
Polynomial 5th order	COMMON_POLY5	16384	*	*	*	*	=	*	*	=	=	=	=		x	CT, FP
Polynomial 7th order	COMMON_POLY7	16640	*	*	*	*	*	*	*	*	=	=	=		x	CT, FP
Polynomial 8th order	COMMON_POLY8	17664	*	*	*	*	*	*	*	*	=	=	=			CT
Extended motion																
- Resulting distance																
Velocity 2nd Order (StartAcc zero)	X_VEL_STARTACC	33280	=	*	*	0	=	*	=	=	=	=	=			CT
Velocity 2nd Order (EndAcc zero)	X_VEL_ENDACC	33280	=	*	*	=	=	*	0	=	=	=	=			CT
Linear velocity	X_LINEAR_VEL	33024	=	*	*	=	=	*	=	=	=	=	=			CT
Linear acceleration	X_LINEAR_ACC	32768	=	*	*	*	=	=	*	=	=	=	=			CT
- Resulting master axis range																
Acceleration-lim. motion (trapezoid profile)	X_FIT_VEL_TRAPEZE_ALIM	7000	*	=	*	=	=	*	=	=	*	*	=			CT, FP

Motion Functionality

	Definition of the motion law at the input "MotionLaw" (ProfileStepAdr)		Travel range		Boundary values left			Boundary value right			Max (not considered everywhere)			TPS possible	Flex-Step possible	Operation modes
			Distance	Range	StVel	StAcc	StJerk	EndVel	EndAcc	EndJerk	MaxVel	MaxAcc	MaxJerk			
Acceleration-lim. Sine curve	X_FIT_SINE_TRAPEZE_ALIM	28928	*	=	*	=	=	*	=	=	*	*	=			CT, FP
Jerk-lim. motion (trapezoid profile)	X_FIT_ACC_TRAPEZE_JLIM	29184	*	=	*	=	=	*	=	=	*	*	*			CT, FP
- Other																
Velocity-lim. Polynomial 5th order	X_MOTION_IN_MOTION_POLY5_VLIM	24576	*	*	*	*	=	*	*	=	*	=	=			CT, FP
Overshooting-free polynomial 5th order	X_MOTION_IN_MOTION_POLY5_SLIM	24832	*	*	*	0	=	*	0	=	=	=	=			CT, FP
Acceleration-lim. (trapezoid profile)	X_VELOCITY_IN_VELOCITY_TRAPEZE_ALIM	20480	*	*	*	0	=	*	0	=	*	*	=			CT, FP
Point table																
Cam	CAMTABLE_1	1...99	*	*	=	=	=	=	=	=	=	=	=			CT, MP, FP

0 Explicitly 0
 * User-defined
 = Calculated automatically
 CT Cam table
 MP MotionProfile
 FP FlexProfile

Fig.9-50: Boundary conditions of the FlexProfile motion laws

Switching and Synchronizing the FlexProfile

Selecting the Coordinate System

The classification of the different profiles in the coordinate system can be relative or absolute with regard to the respective master axis and slave axis. This allows the combination of the permitted setting options (see the figure below).

The red line symbolizes the current profile and the blue line symbolizes the profile to which it should be switched.

Motion Functionality

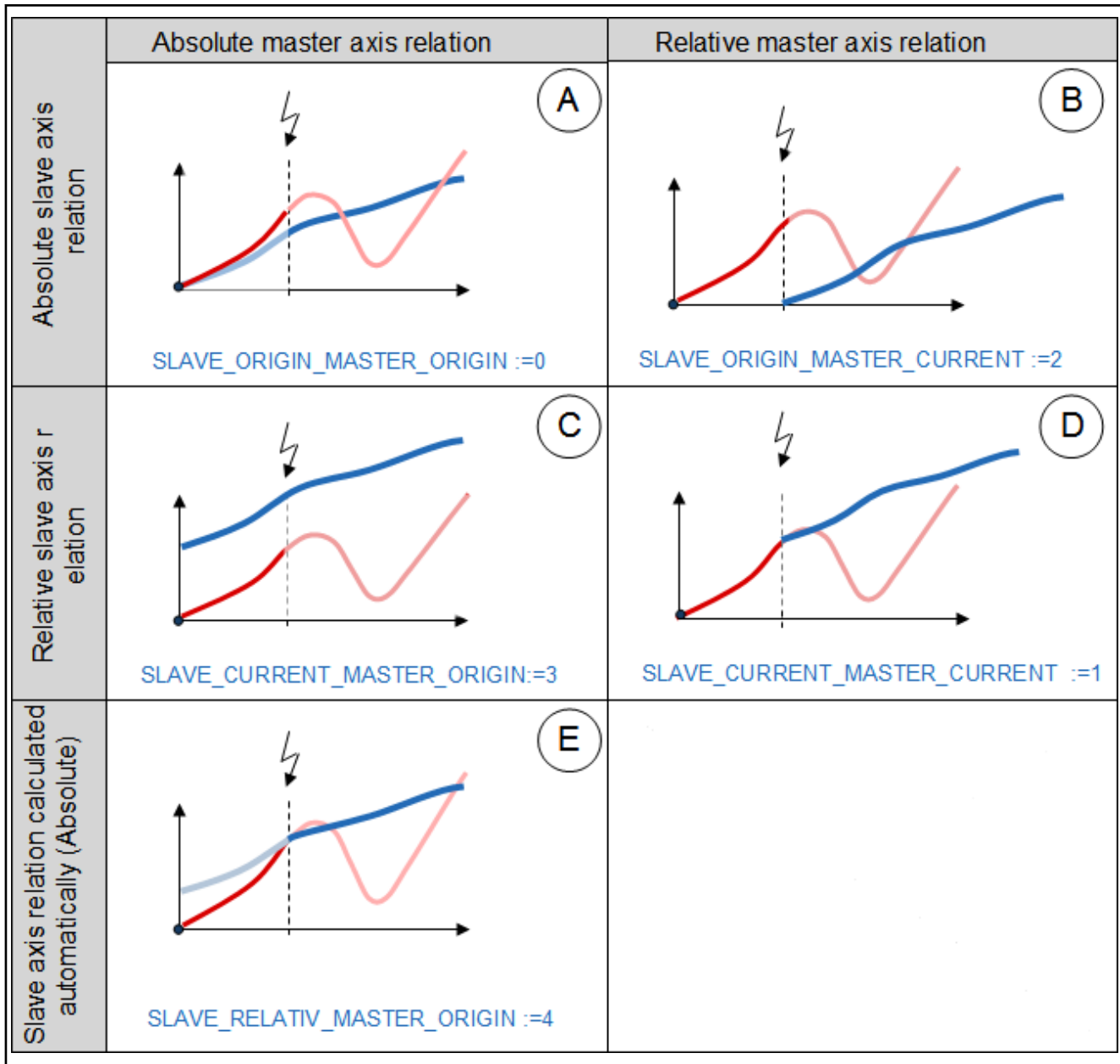


Fig.9-51: Selecting the coordinate system

Switching and Synchronizing

After the assignment of profiles has been defined, both the switching character (Switching at angle or immediately) and the synchronization type (with or without synchronization) can be selected. When switching to phase, wait after the enable signal till the specified phase is reached and at exactly this point, switching to another state is executed. In case of a decoupled state, the new profile is started instead. Immediate switching is executed simultaneously with the input of a positive edge. When using this functionality, note the congruence of the FlexProfiles. If this is ignored, command value jumps can occur in case of some configurations.

The following figures present some possible combinations for switching and synchronizing together with arbitrary coordinate systems.

The following figure shows the switching to a different MotionProfile. The origin of the first and second MotionProfile is congruent and is located in the

Motion Functionality

zero point of the coordinate system (absolute master axis position and absolute slave axis position - case A). The MotionProfiles are not moved.

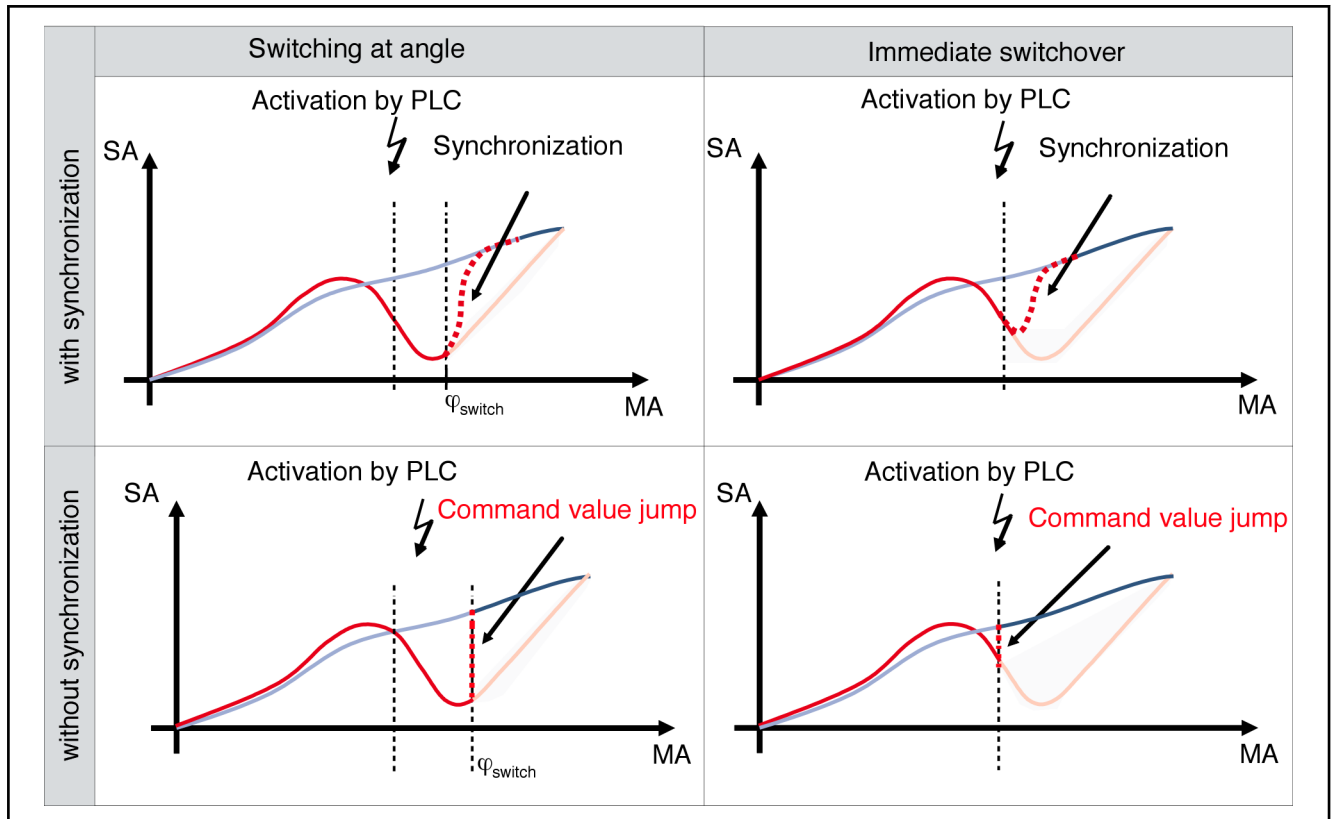


Fig.9-52: Case A – Switching and synchronizing with absolute master axis position and absolute slave axis position

The following figure shows the switching to a different MotionProfile. The origin of the second MotionProfile is moved to the current master axis position of the first MotionProfile (relative master axis position and absolute slave axis position – case B). The MotionProfile is not moved towards the slave axis.

Motion Functionality

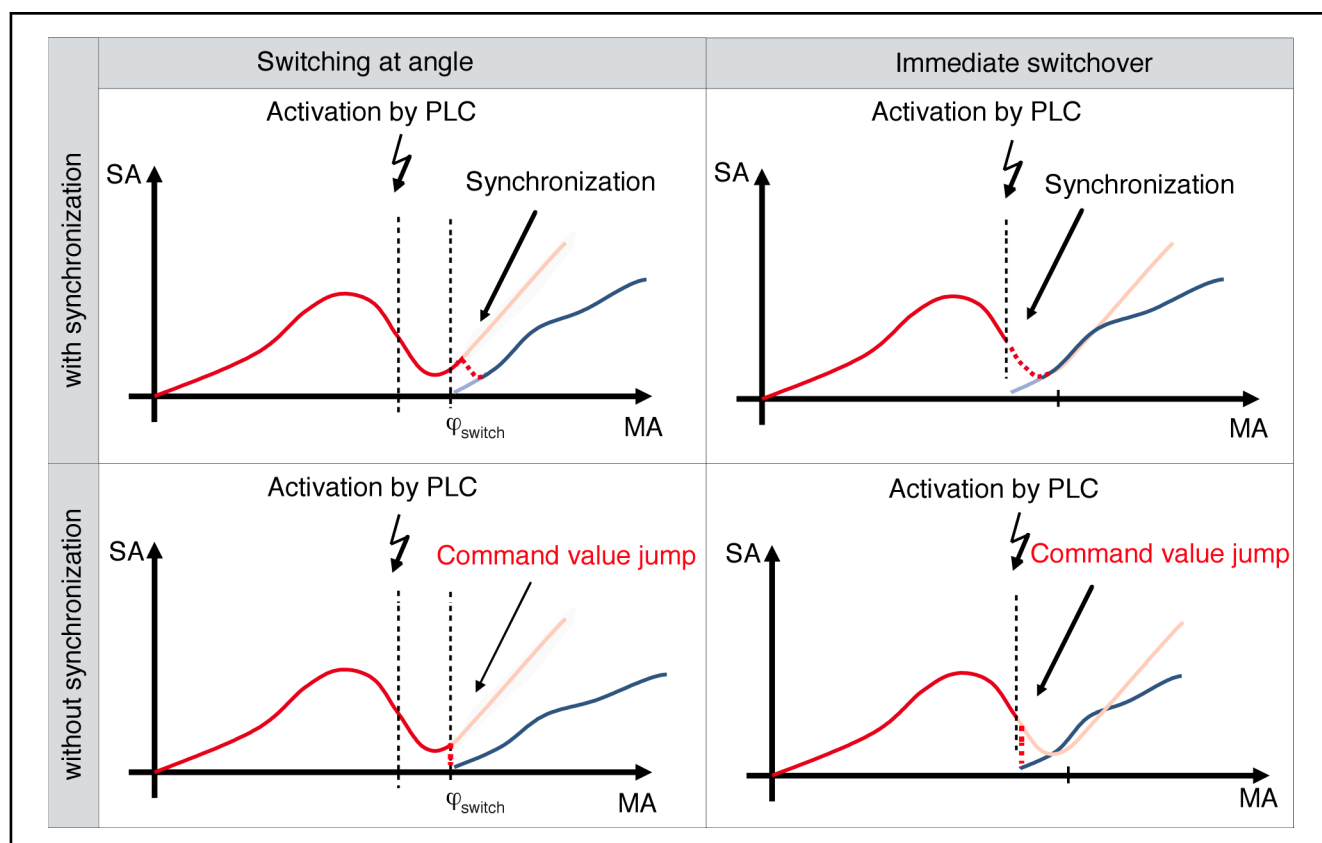


Fig.9-53: Case B – Switching and synchronizing with relative master axis position and absolute slave axis position

The following figure shows the switching to a different MotionProfile. The origin of the second MotionProfile is displaced to the current slave axis position of the first MotionProfile (absolute master axis position and relative slave axis position - case C). The MotionProfile is not moved towards the master axis.

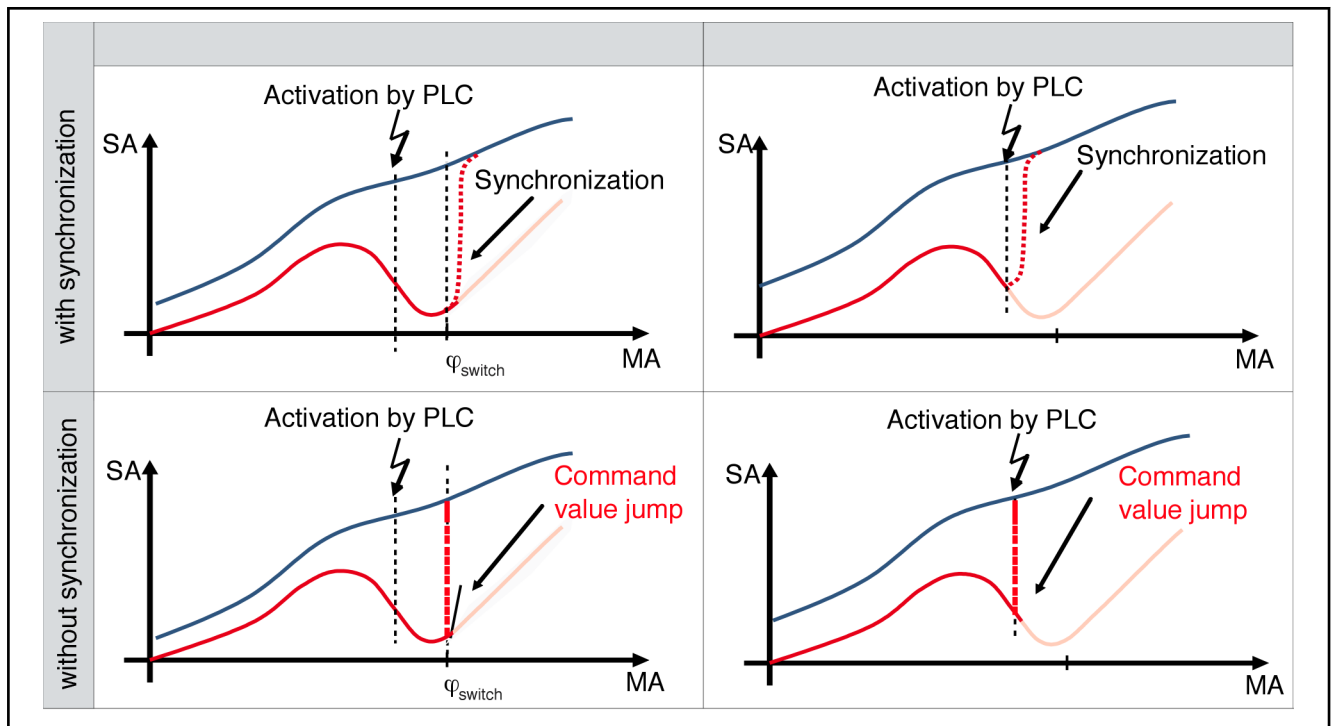


Fig.9-54: Case C – Switching and synchronizing with absolute master axis position and relative slave axis position

The following figure shows the switching to a different MotionProfile. The origin of the second MotionProfile is moved to the current master and slave axis position of the first MotionProfile (relative master axis position and relative slave axis position - case D).

Motion Functionality

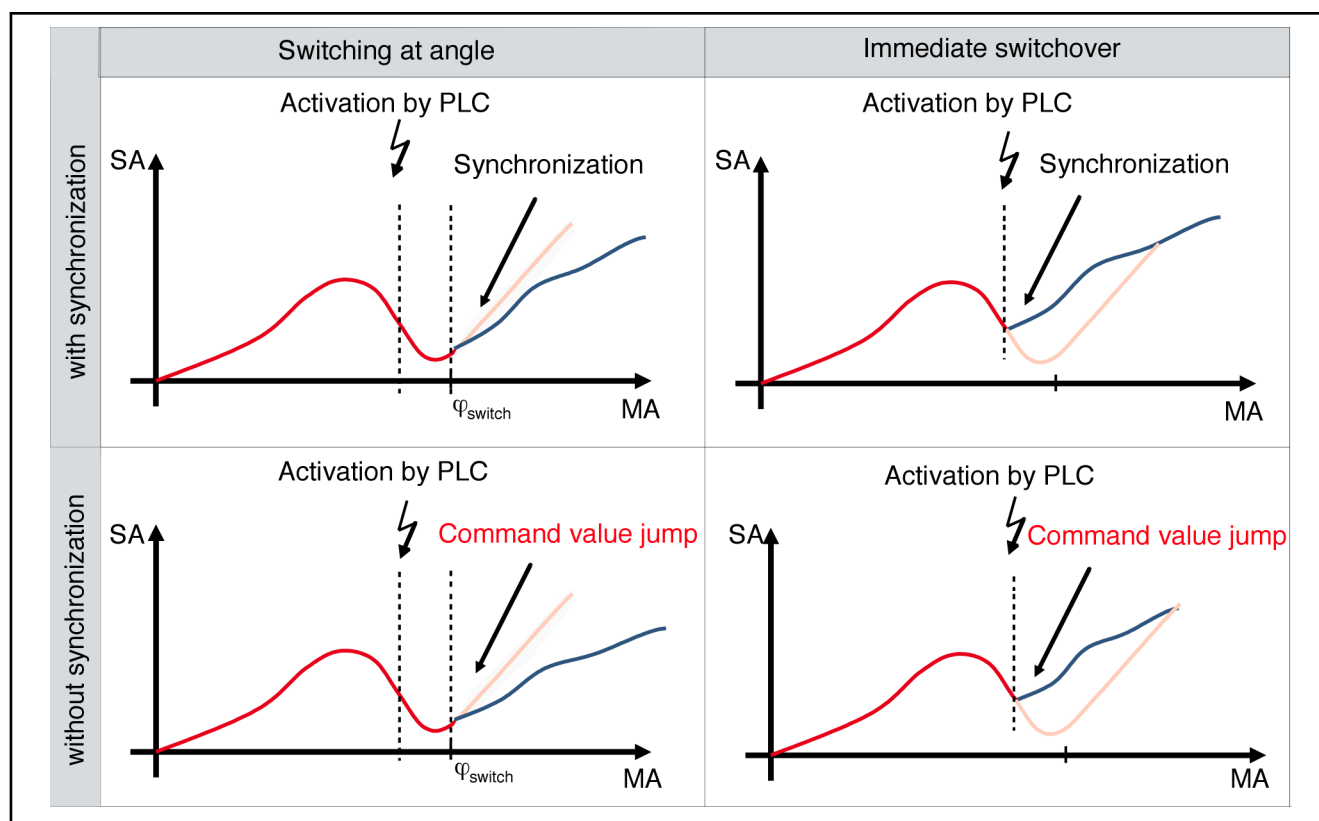


Fig. 9-55: Case D – Switching and synchronizing with relative master axis position and relative slave axis position

The following figure shows the switching to a different MotionProfile. The second MotionProfile is displaced to the current slave axis position of the first MotionProfile and started (absolute master axis position and automatically determined slave axis position (relative) - case E). The MotionProfile is not moved towards the master axis.

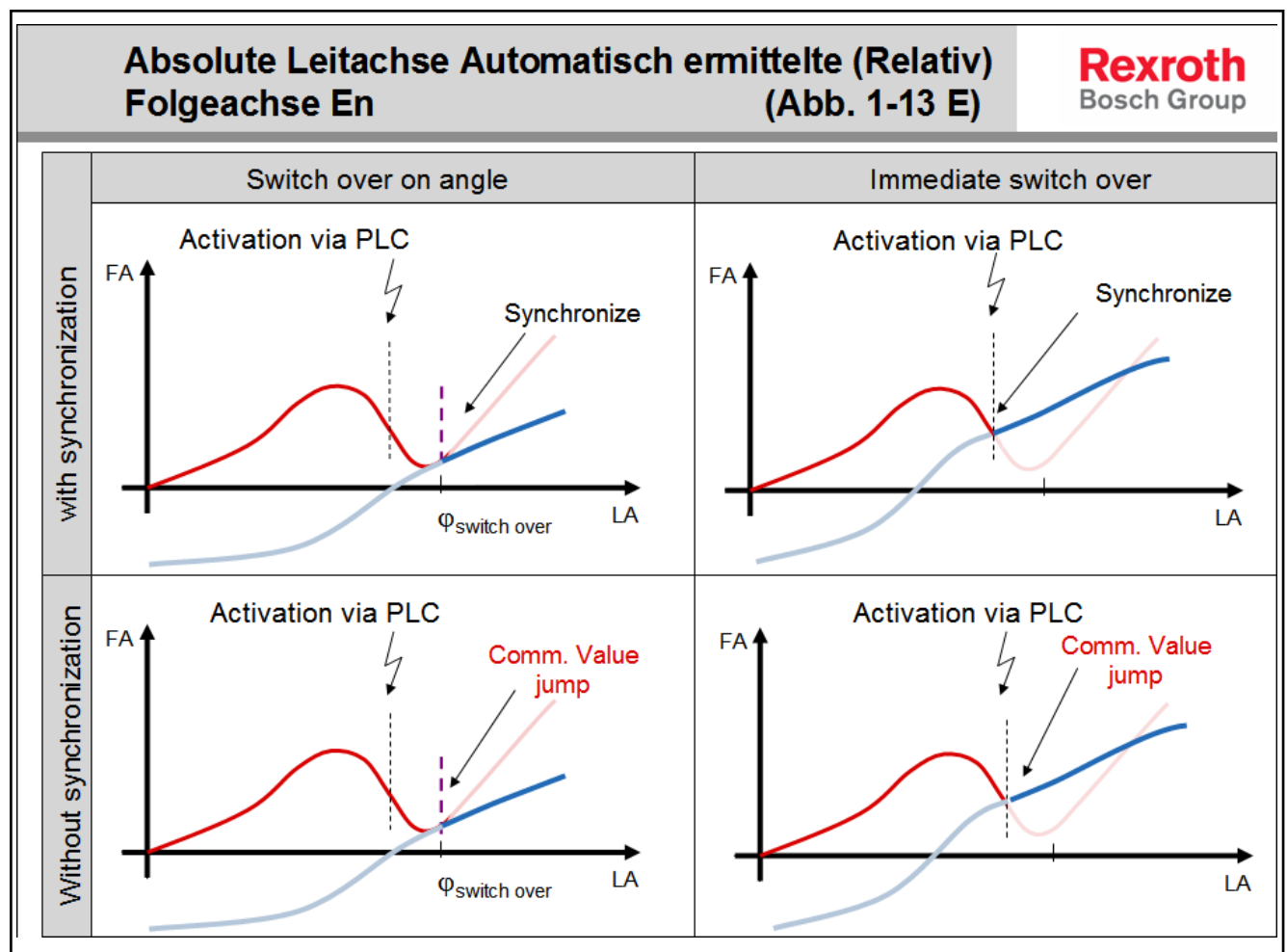


Fig.9-56: Case E – Switching and synchronizing with absolute master axis position and automatically determined slave axis position (relative)

Processing type

There are two processing types (see the following figure):

- Continuous processing
- One-time processing (one shot)

During continuous processing, the active FlexProfile is executed cyclically. Ensure that the final profile velocity of the last motion step is identical to the initial velocity of the entire profile. If necessary, an error message is output. With the one-time processing, the selected profile is only executed once. The same profile can be started several times with a new call for the "ML_Flex-Profile" FB.

Motion Functionality

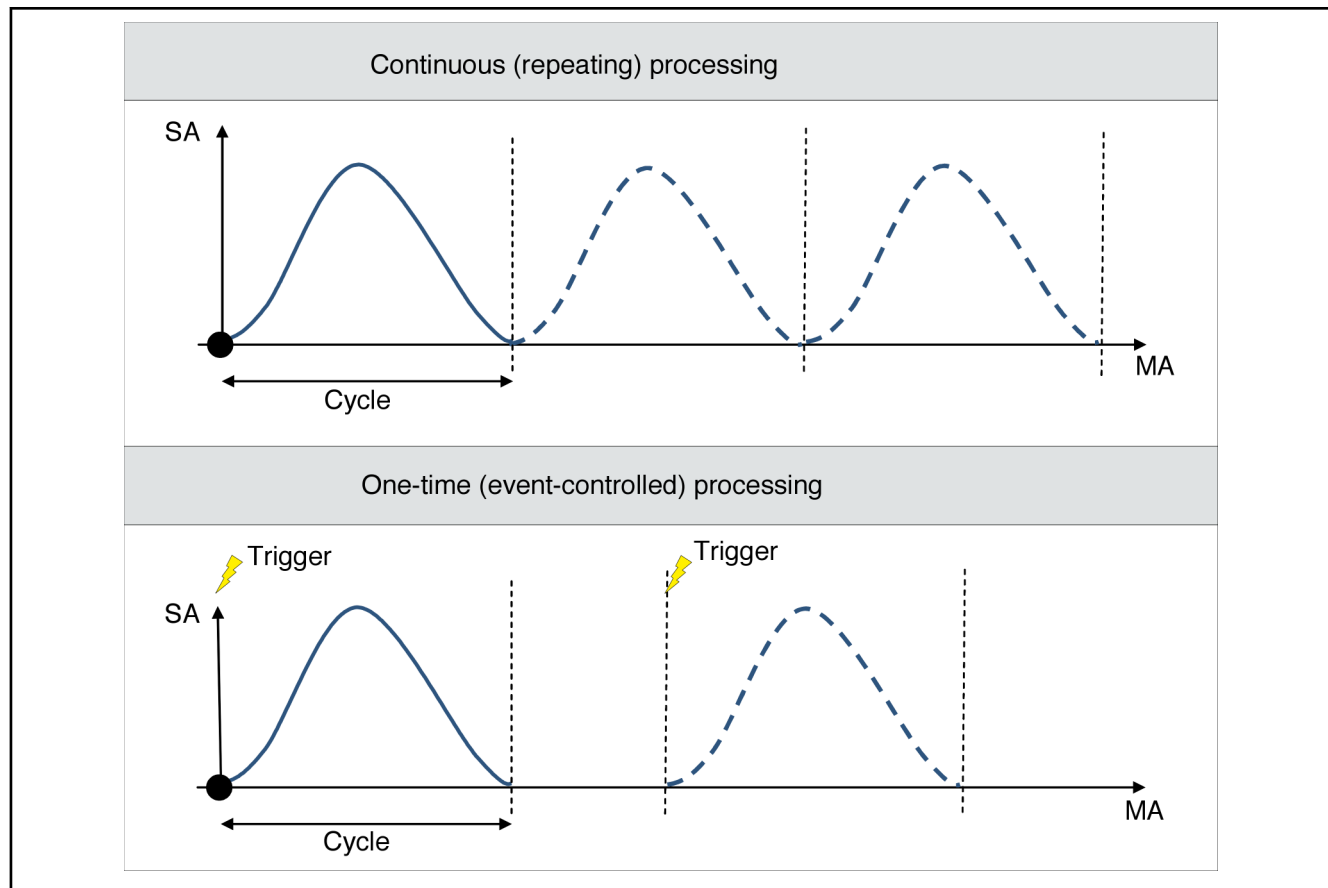


Fig.9-57: Continuous processing and one-time processing

Event Control

Characteristics of the Event Control

In almost all the processing machines, several axes participate in the process. The motion tasks to be carried out by the individual axes mostly depend on one another and result from the application task.

Therefore, the motion of one axis often has to start at exactly the place where another axis has reached a certain position or where another axis remained in a motion step for a certain time. This is shown in the following picture.

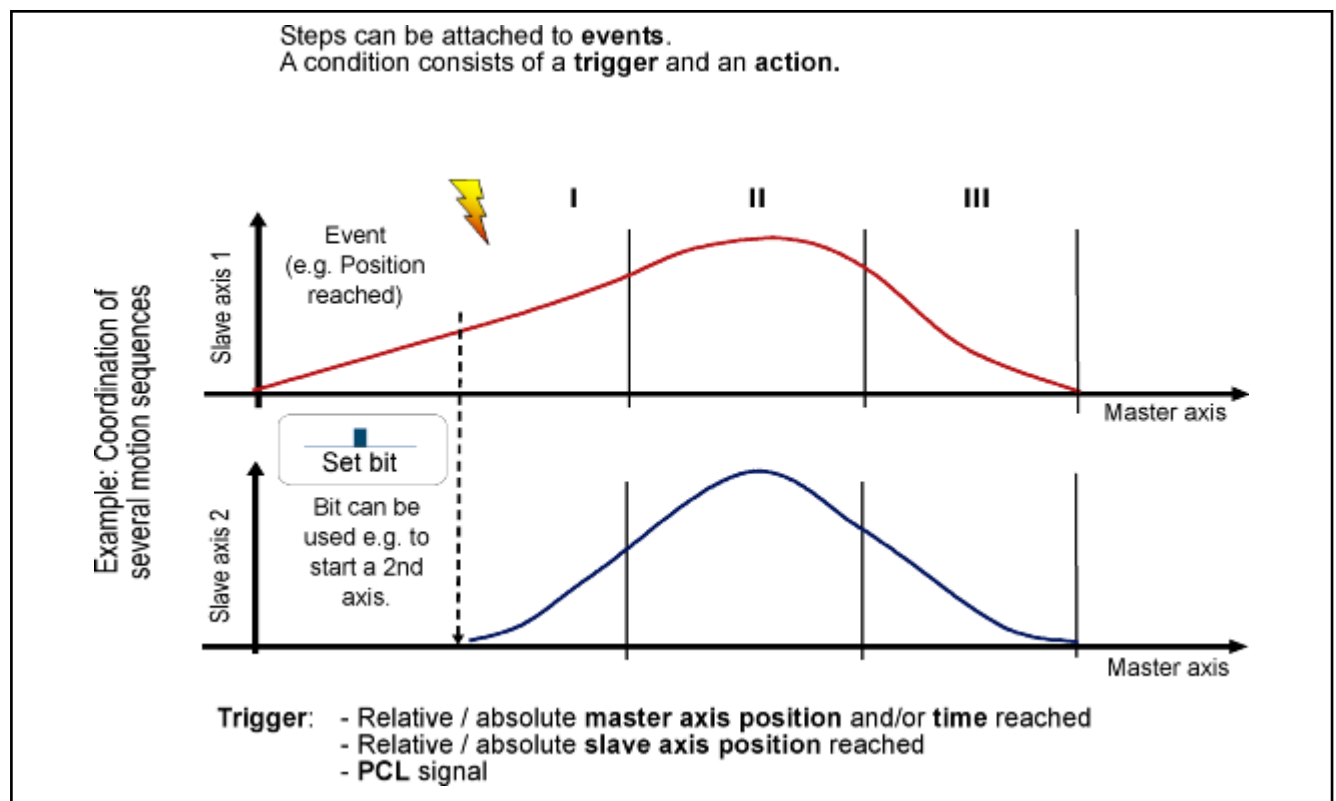


Fig.9-58: Example of an event control

Events are very suitable to mark these points in the MotionProfile. It is very easy to react in a PLC via the AxisData structure. It is possible to start follow-up motions of other axes or to decide on the further processing. A resulting dynamic "motion chain" is realized in a PLC reflecting the machine dependencies.

Events can be attached to the motion steps. An event consists of a trigger and an action.

The trigger identifies the control variable states. These are:

- Relative/absolute master axis position reached
Describes the reaching of a relative/absolute position of the master axis. Similar to a programmable limit switch, the current position of a master source can be monitored. If it crosses a specified position in the current motion step, the event occurs
- Relative/absolute slave axis position reached
Describes the reaching of a relative/absolute position of the slave axis. The same concept can also be applied to the slave axis position
- PLC signal
Actions can be triggered by a PLC. Events are reported to the PLC or directly start an action
- Time-controlled abort
The step is not time-controlled. That means that it has no time master, but a master axis to which traveling should be synchronous. Connect an event which monitors the travel time since the beginning of the motion step with an action triggering an abort of the current motion step. The resulting MotionProfile can be used generically for different master axis velocities

Motion Functionality

- Monitoring for time steps

With regard to the time-master functionality, the events can also be used for the monitoring of different master sources. Depending on the motion task, an event can be set in such a way that the travel distance of the master axis is monitored during a motion step coupled to the time master. Thus, execute error reactions if the master axis traveled too far during this time. Different process states can be easily and comfortably buffered by the user with the help of the events. It is possible to react on this in a PLC

The action is an active element responsible for the activation of certain variables.

Possible actions are:

- Set bit immediately
One of the available 16 bits is immediately set in the AxisData.
- Set bit at the end of the motion step
One of the available 16 bits is set in the AxisData at the end of the motion step
- Set bit at the end of the profile
One of the available 16 bits is set in the AxisData at the end of the profile
- Set bit immediately and continue with the next motion step
Causes the immediate abortion of the current motion step. This step is not processed completely, instead, the next motion step begins directly from the ongoing motion. If this is a Flex motion step, this change can take place smooth and jerk-free.

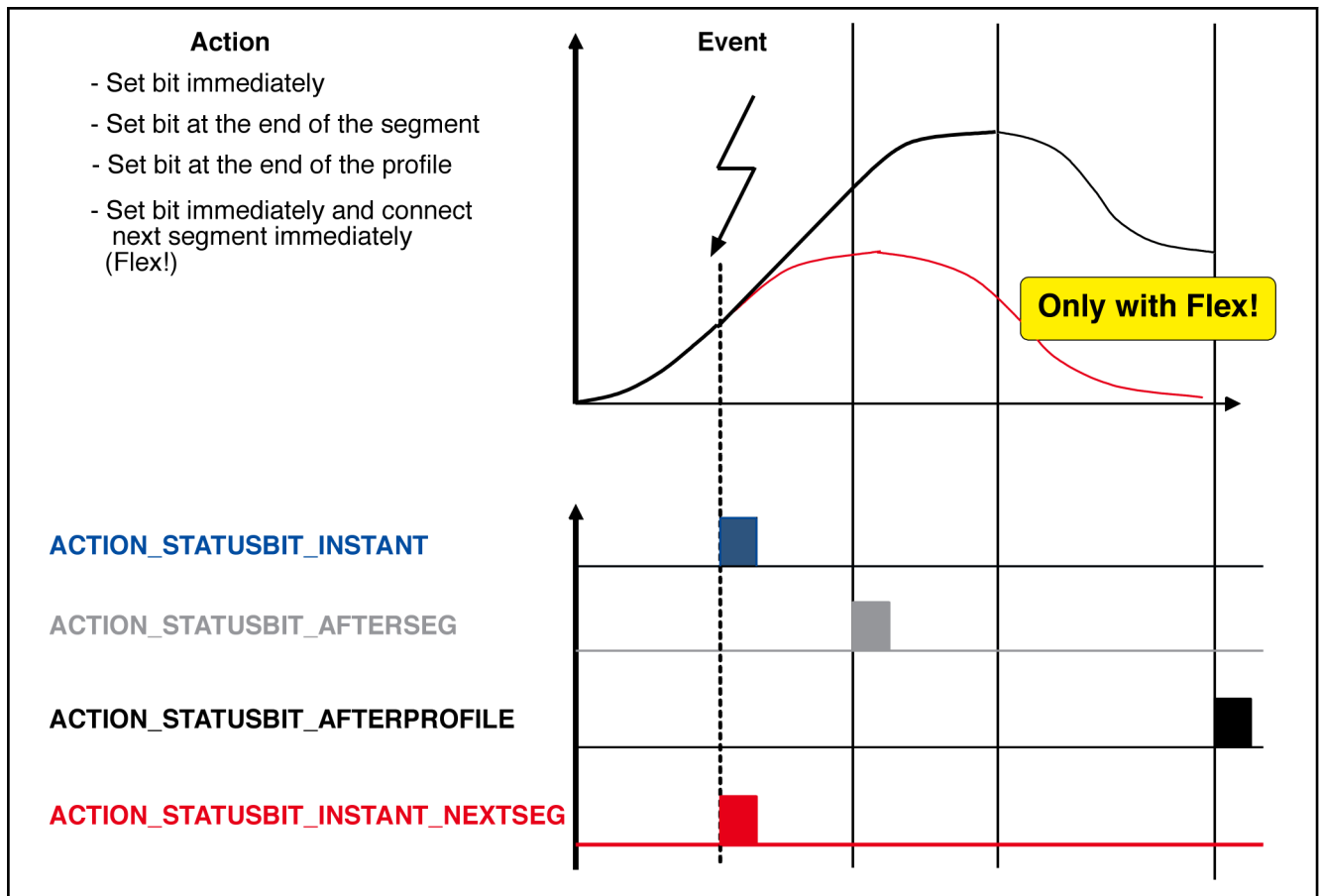


Fig.9-59: Possible actions

The events can be configured either via the PLC function block "MB_ChangeFlexEventSet" or the IndraWorks interface.

Assigning MotionProfile and Axis

Up to four profiles can be created for each axis; in turn, each can consist of a maximum of 16 motion steps. The individual motion steps can either be defined with certain motion laws (VDI 2143 and special motion laws) or with the cam tables referenced to those. The cam tables are created as C-parameters (C-0-2001 to C-0-2099). The cam tables can either be configured using the PLC function block MB_ChangeCamData or the IndraWorks interface.

Motion Functionality

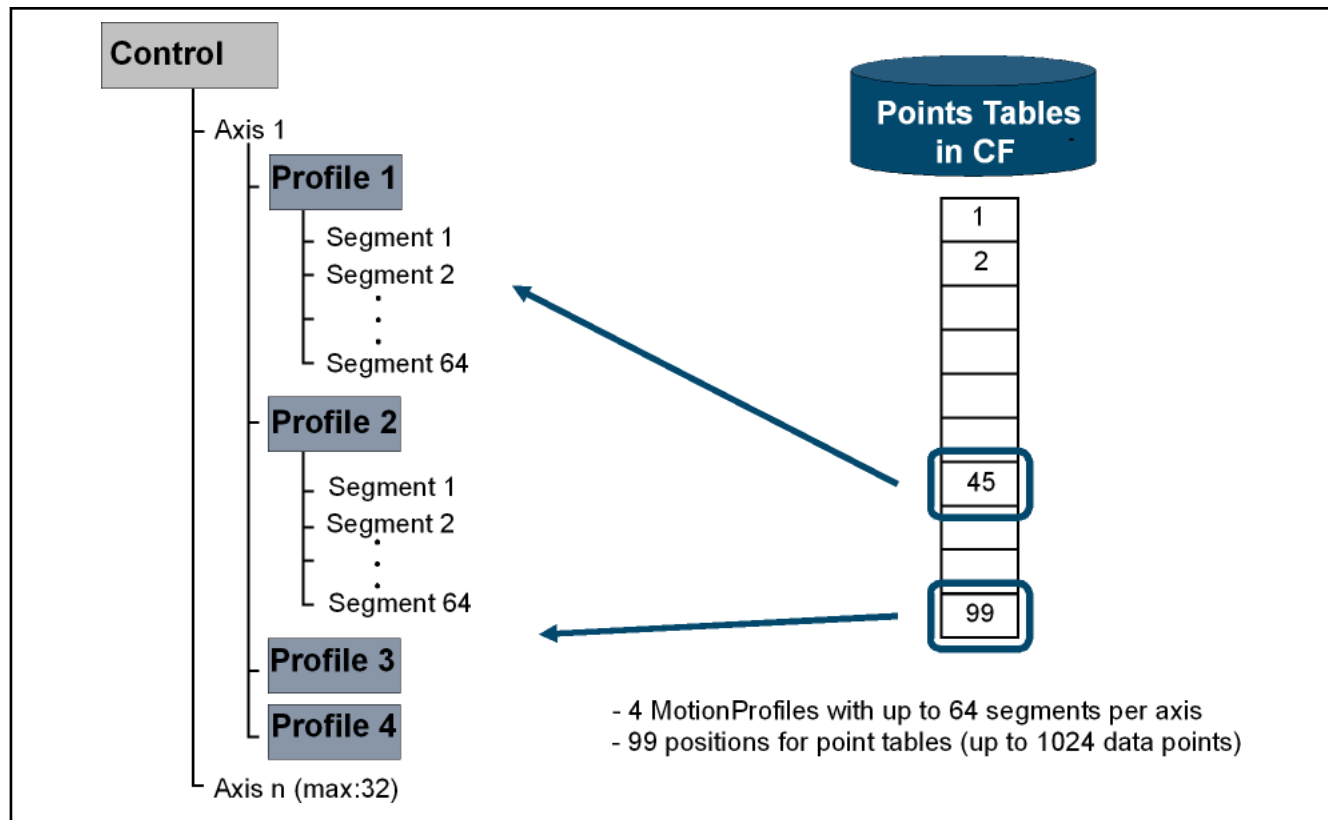


Fig.9-60: Assignment of MotionProfile and axis

FlexProfile – Configuration

Overview

FlexProfiles can either be configured using the FlexProfile Editor of the IndraWorks interface or the available function blocks of the PLC (see figure below).



Fig.9-61: FlexProfile configuration possibilities

The FlexProfile Editor (IndraWorks) is a tool supporting the user to clearly and precisely create and modify MotionProfiles for the corresponding coupling requirements. Several options are available for this procedure.

Configuration Using PLC Function Blocks

The PLC provides four function blocks that allows the FlexProfiles to be used. In general, these can be divided into two groups.

The first group consists of the function blocks "MB_ChangeFlexProfileSet", "MB_ChangeFlexEventSet" and "MB_ChangeCamData" permitting FlexProfiles and cam tables to be written and configured.

The second group consists of the "ML_FlexProfile" FB permitting a FlexProfile to be started.

The "MB_ChangeFlexProfileSet" FB is used to configure a FlexProfile. The individual motion steps are defined by structures (see the description of the

FB). Note that not all structural elements need to be defined during the motion step configuration - these have default values.

Analog to the "MB_ChangeFlexProfileSet" FB is the "MB_ChangeFlexEventSet" FB with which events can be written to the respective motion steps of the FlexProfile. In this case, configuration is also executed with structures.

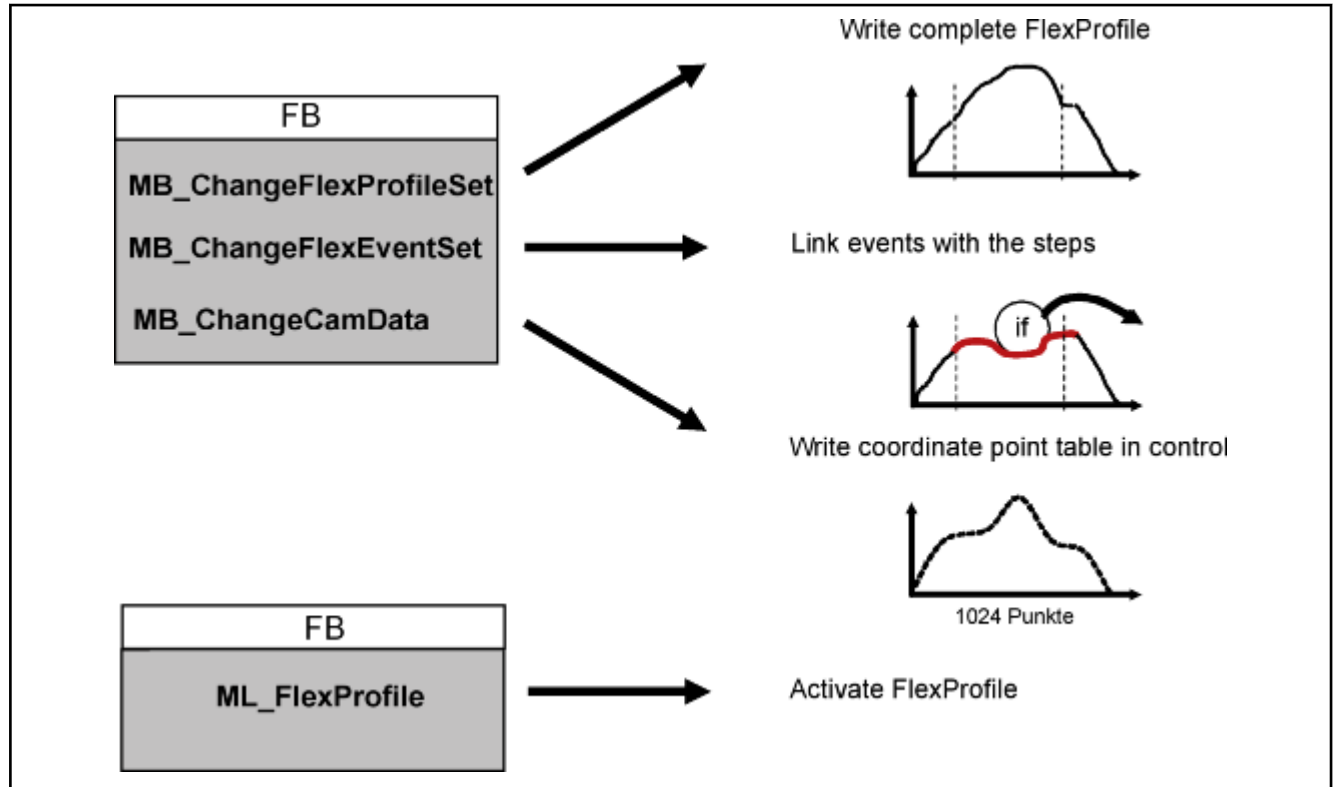


Fig. 9-62: Overview of the PLC function blocks

Finally, the corresponding motion steps can be linked to cam tables. These are defined among the C-parameters ("C-0-2001, Cam table 1" to "C-0-2099, Cam table 99") and can be written using function block "MB_ChangeCamData".

Configuration Using IndraWorks

By clicking on the "FlexProfile" entry in the axis tree of the desired project and axis, a dialog opens in which the general profile configuration can be set.

Motion Functionality

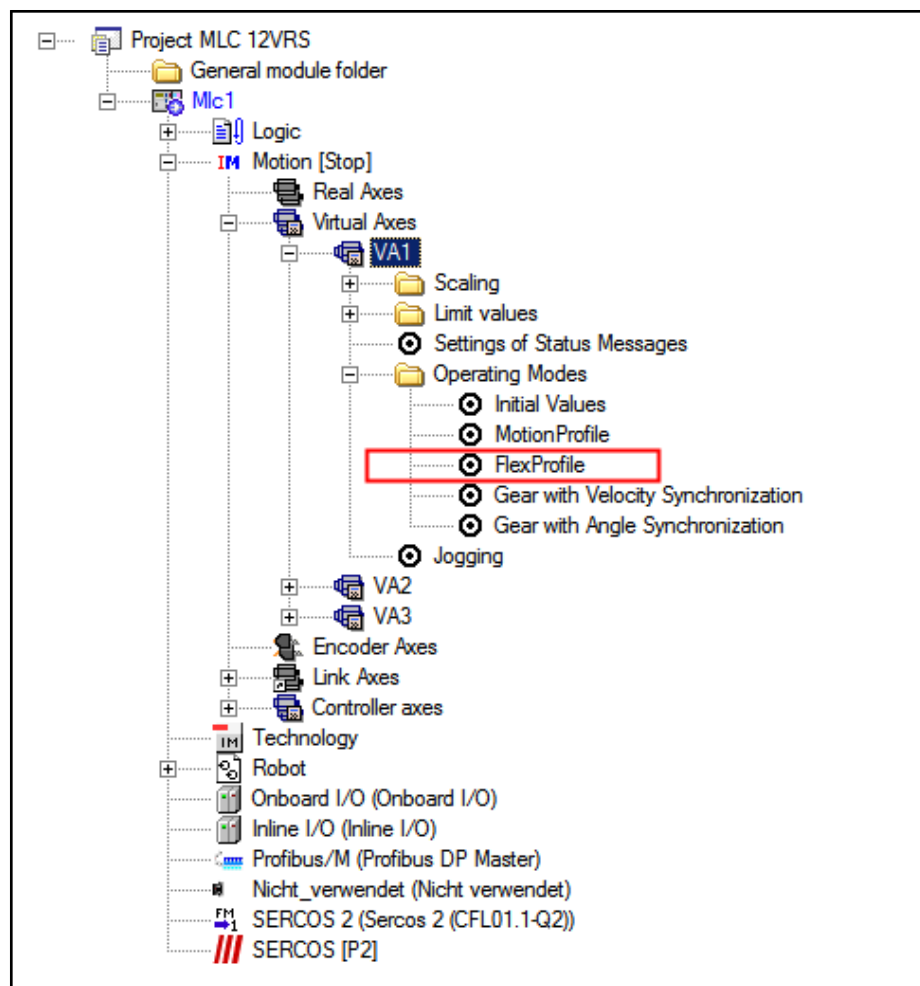


Fig.9-63: Locating the FlexProfile in the axis structure

FlexProfile MotionProfiles can be created/modified using the respective setting options. Furthermore, various parameters are visualized relating to the FlexProfile operating state, e.g. the master axis position, offsets, phase offset, electronic gears and more...

The action chain between master axis and slave axis and their current status is shown first.

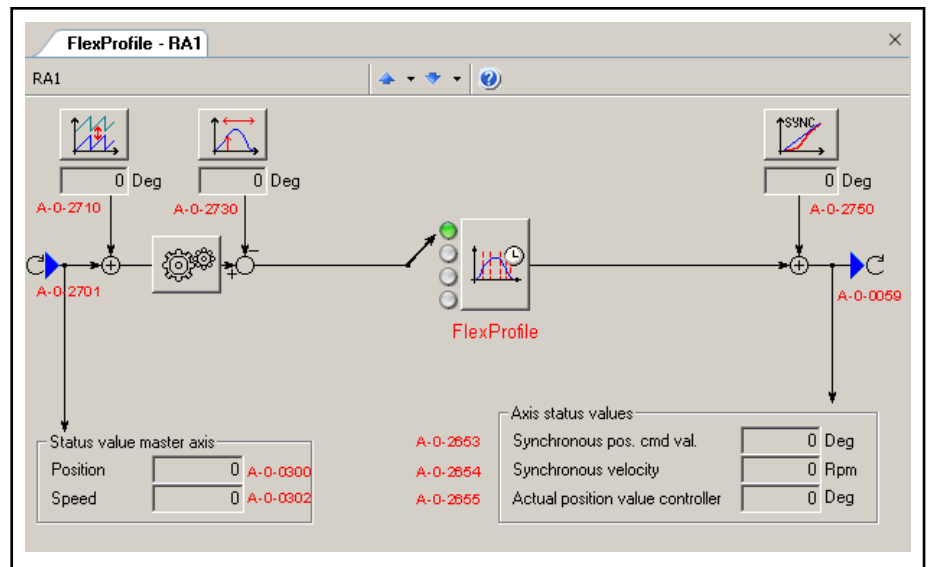


Fig.9-64: FlexProfiles: Action chain between master axis and slave axis

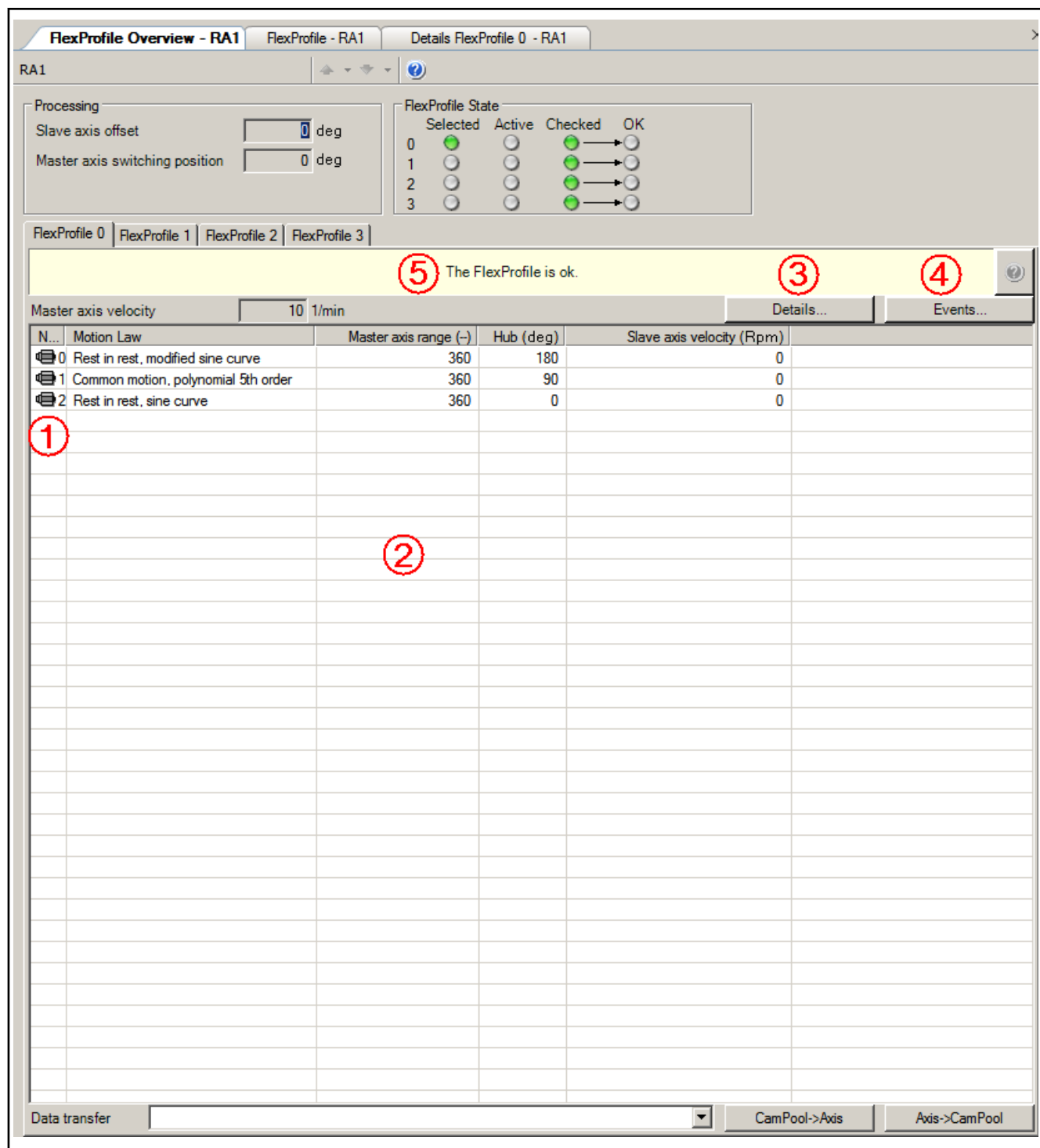
The MotionProfiles of the FlexProfile can be reached by clicking on the "Flex-Profile Overview" icon in the middle of the dialog (see [fig. 9-64 "FlexProfiles: Action chain between master axis and slave axis" on page 421](#)). The current FlexProfile configuration is shown. This information is available for all MotionProfiles of the axis.



Four FlexProfile MotionProfiles can be stored for the axis.

Furthermore, the MotionProfile states and their processing status are shown in the status window. Errors in the profile definition are reported in the status window above it. A new MotionProfile can be created - or an existing one can be modified - using the "Details" button (see [③ fig. 9-65 "FlexProfile overview window" on page 422](#)).

Motion Functionality



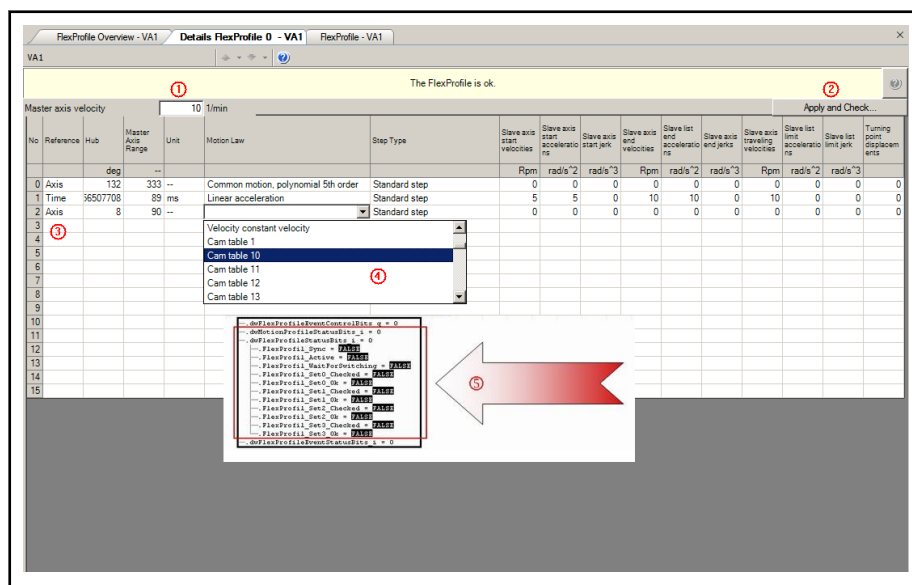
- ① Symbolic representation of the segment type (axis or time)
- ② FlexProfile settings
- ③ FlexProfile configurator
- ④ Event configurator
- ⑤ Error display

Fig.9-65: FlexProfile overview window

After the action is completed, the FlexProfile Editor opens automatically. First, an empty configuration table is shown. Every line defines a profile step. The master axis sources, motion laws and motion step types required can be selected from the drop down menu. After the FlexProfile configuration has

been completed, the changes have to be applied to the control. This is executed using the "Apply and check" button (see ② [fig. 9-66 "FlexProfile configuration" on page 423](#)).

It is to be considered that a MotionProfile can only be started if it is checked and marked as valid. This check consists of several plausibility tests, such as invalid boundary value combinations.



- ① Master axis velocity
- ② Applying and checking the configuration
- ③ Master axis source selection
- ④ Motion laws and motion types can be selected from a list
- ⑤ FlexProfile information in the AxisData

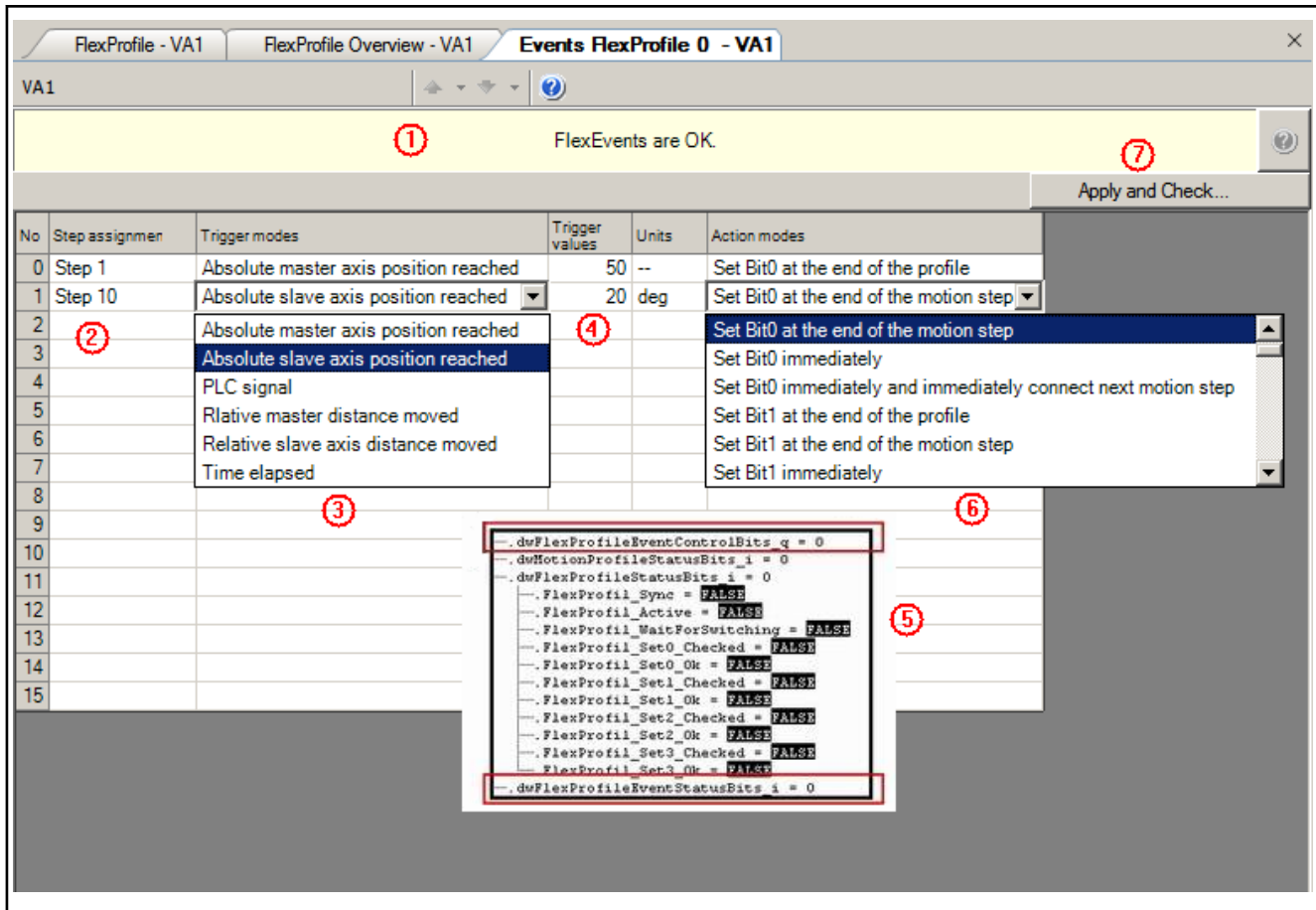
Fig. 9-66: FlexProfile configuration



If the profile is written using the PLC function block, the data is not permanently stored in the control or on the Compact Flash card.

The events of the respective motion steps of the profile can be assigned using the "Events" command button (see ④ [fig. 9-65 "FlexProfile overview window" on page 422](#)). An empty configuration table is shown. Every line defines a new event. The actions and triggers required can be selected from a drop-down menu. Apply the events to the control with "Apply and Check" (see ② [fig. 9-66 "FlexProfile configuration" on page 423](#)).

Motion Functionality



- ① Diagnostic window
- ② Motion steps can be selected from a list
- ③ Trigger can be selected from a list
- ④ Trigger value specification
- ⑤ FlexProfile information in the AxisData
- ⑥ Action can be selected from a list
- ⑦ Applying and checking the configuration

Fig.9-67: Event configuration window

Configuration using the CamBuilder

The CamBuilder is described in /7/, [Required and Supplementing Documentations](#), page 14.

9.10 Touch Probe

9.10.1 Introduction

With a binary sensor (touch probe), the touch probe function allows a highly accurate position determination in the running production process. Based on determined values, the required position corrections can be made for the best adjustment between product and production process. Use cases: Printed mark detection, register controller in printing and packaging industry as well as metal sheet processing.

9.10.2 Features

A touch probe has the following properties:

- Up to 72 touch probe inputs, depending on the hardware

- 100 touch probe evaluation units
- Measuring signals can be actual position values of all configured axes
- Free assignment of touch probe input to evaluation unit and axis
- Measurement of actual position value differences
- Determination of time intervals between measuring signals
- Triggering measurement by positive or negative touch probe signal edges
- Single shot measurement or continuous measurement can be selected
- Counting measurement events during continuous measurement
- Determining a position value range ("expectation window") per touch probe in which the measurements should take place (activation of a "failure counter" when expectation window does not return measurement result)

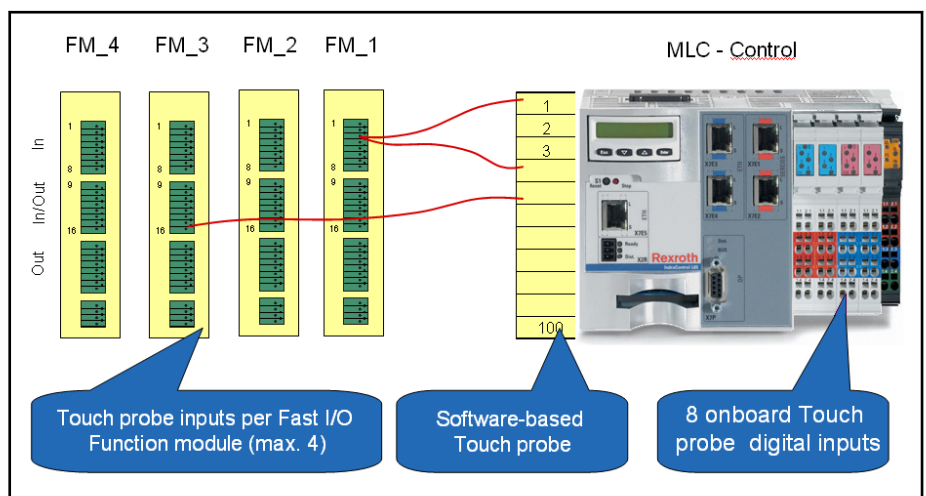


Fig.9-68: Touch probe hardware architecture

9.10.3 Functional Description

The touch probe input digitally evaluates the voltage level of the touch probe signal, i.e. only the signal states "high" (1) or "low" (0) can be detected. When actuating the touch probe, the signal state changes. The touch probe reports a rising (positive) or a falling (negative) switching edge.

The signal level ranges for "high" (1) and "low" (0) of the "fast" digital inputs are described in /24/, [Required and Supplementing Documentations](#), page 14.

Motion Functionality

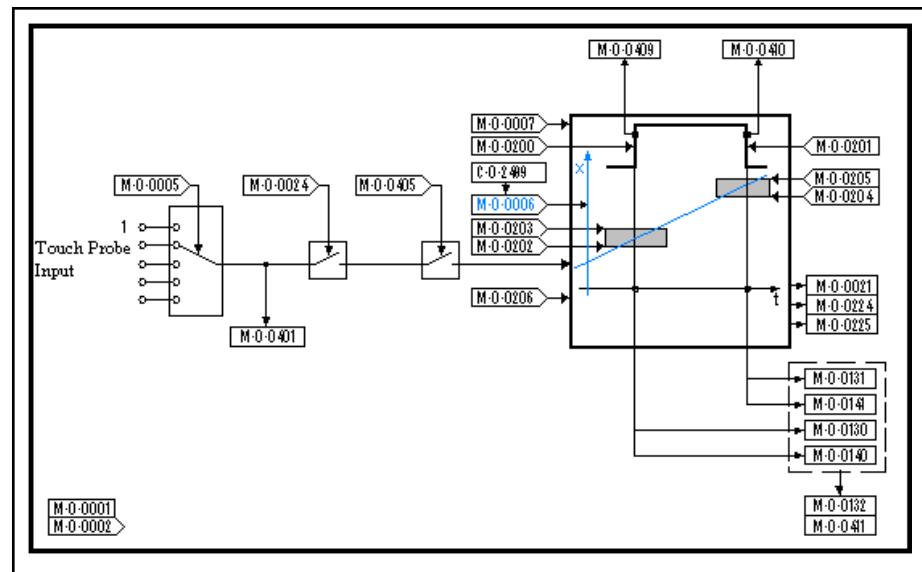


Fig. 9-69: Overview and mode of operation of the touch probe-related parameters

Touch probe-related parameters

- M-0-0001, Logic touch probe number
- M-0-0002, Touch probe name
- M-0-0005, Input selection
- M-0-0006, Signal selection
- M-0-0007, control word
- M-0-0021, Status word
- M-0-0024, Command: Activation
- M-0-0130, Measured value, positive edge
- M-0-0132, Difference value
- M-0-0140, Time stamp, positive edge
- M-0-0141, Timestamp, negative edge
- M-0-0200, Dead time compensation, positive edge
- M-0-0201, Dead time compensation negative edge
- M-0-0202, Starting position of the expectation window, positive edge
- M-0-0203, End position of the expectation window, positive edge
- M-0-0204, Starting position of the expectation window, negative edge
- M-0-0205, End position of the expectation window, negative edge
- M-0-0206, Max. number of marker failures
- M-0-0224, Number of marker failures, positive edge
- M-0-0225, Number of marker failures, negative edge
- M-0-0401, Input status
- M-0-0405, Enabled
- M-0-0409, Counter of measured values, positive edge
- M-0-0410, Counter of measured values, negative edge
- M-0-0411, Counter of difference values
- C-0-2499, Touch probe, IDN list signal selection

The currently detected signal state at the respective touch probe input is displayed in parameter M-0-0401. If the input is positioned at a function module, this module has to be activated (status LED flashes in green).

9.10.4 Accuracy

The actual position values are received in the raster of the set sercos cycle time by the control or generated by the drive. The actual position value belonging to the point of time of the signal edge is determined by means of linear interpolation between the last and the next generated actual position value.

9.10.5 Data Acquisition of the Touch Probe

Activation of Measured Value Acquisition

A switching edge at the touch probe input can "trigger" the acquisition of a measured value. Switching edges only result in the acquisition of a measured value if the following conditions are fulfilled:

- The corresponding hardware input was selected in parameter "M-0-0005, Selection input".
- The presetting for the measured value acquisition in case of positive or/and negative switching edge at the respective touch probe input was activated in "M-0-0007, Control word".
- The measured value acquisition was activated via the "M-0-0024 Command: Activation" enabled
- The respective touch probe input for the measured value acquisition was enabled in "M-0-0405, Enabling"

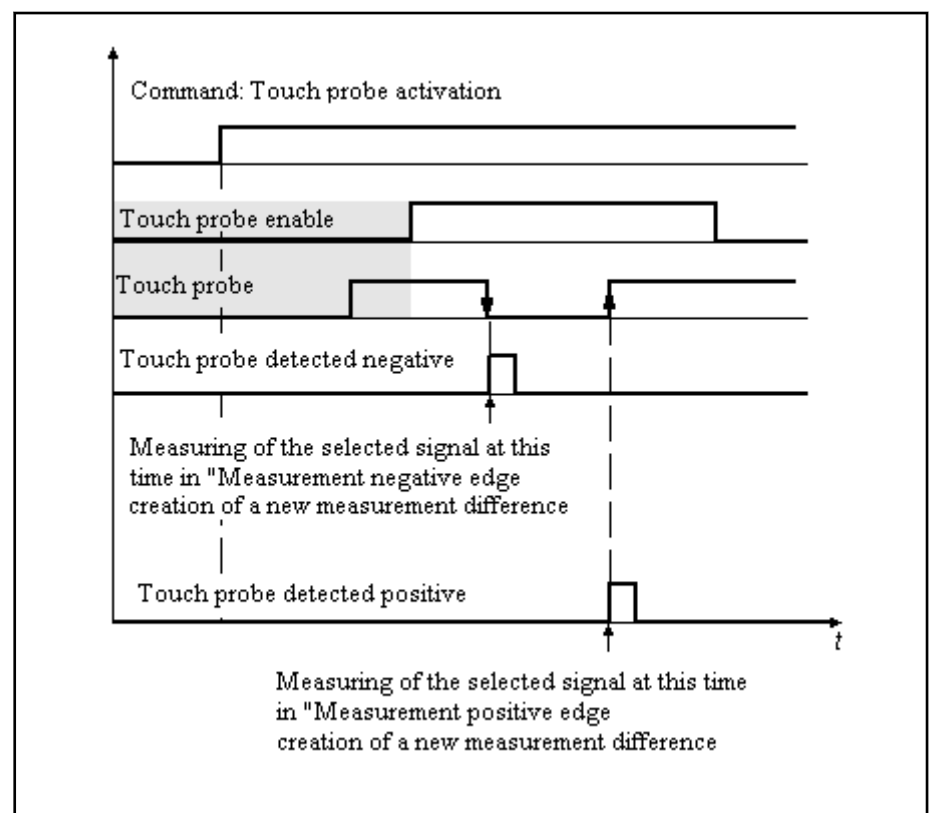


Fig.9-70: Evaluation of touch probe edges (in case of switched on evaluation of positive and negative edges in M-0-0007)

Motion Functionality

Bit 0 and 1 are applied for exactly one cycle in parameter "M-0-0021, Status word". The bits are reset to 0 in the next cycle, unless another touch probe event occurred.

Measuring Signal Selection

The signal selection, the value of which is measured during the respective measured value trigger, takes place by entering the respective IDN in the parameter "M-0-0006, Signal selection".

The IDN of parameters assigned to the measuring signals which can be selected are listed in the "C-0-2499, Touch probe, IDN list, signal selection".

Measured value acquisition mode

The measurement mode can be set for each touch probe in the control parameter M-0-0007. The measured value acquisition has to be activated.

The following enable modes are possible:

- **Single shot measurement** - After the touch probe (M-0-0405) was enabled, a measured value is only applied during the first measured value trigger. Prior to every further measurement, a new enabling of the touch probe is necessary!
- **Continuous measurement** - After the touch probe (M-0-0405) was enabled, a new measured value is only applied during the first measured value trigger.

Moreover, further configurations of the measured value acquisition can be preset in parameter M-0-0007:

- Limiting a range for actual position values ("expectation window") in which measurements can take place.
- No trigger failure monitoring which registers a traveling without measuring event can be activated for the "expectation window".

Saving Measured Values

Depending on the switching edge of the touch probe signal (setting in M-0-0007) activated by the measured value trigger, the measured value of the signal selected from the list parameter C-0-02499 is saved in one of the following parameters:

- M-0-0130, Measured value, positive edge
- M-0-0131, Measured value, negative edge

In case of continuous measurement and single shot measurement, the difference from the last two measured values of the same touch probe are calculated which were measured during opposed (positive/negative/positive...) triggering of the measured values (setting in M-0-0007). This difference is saved in parameter "M-0-0132 Difference value". The difference measurement type can also be changed via the parameter M-0-0007: Measurement of

- the difference between negative and negative edge
- the difference between positive and negative edge
- the difference between negative and positive edge
- the difference between positive and positive edge

Dead time compensation

Via the parameters

- M-0-0200, Dead time compensation, positive edge
- M-0-0201, Dead time compensation negative edge

the signal runtime of the actual touch probe event can be compensated until the detection in the control. The dead time is optimally set if the determined position remains constant irrespective of the axis velocity. This can be used to determine the system dead time.

The determined touch probe position is calculated from:

$$P_{Erm} = P_{Det} + v_{Act} \cdot T_{Tot}$$

Fig. 9-71: Determining the master position

With

P_{Det} : Determined touch probe position

P_{act} : Axis position when touch probe event occurs

v_{Act} : Actual axis velocity

T_{Dead} : Dead time of touch probe input

If the touch probe position is now measured at two different velocities, the dead time is calculated as follows (assumption: constant dead time):

$$T_{Tot} = \frac{P_{Erm2} - P_{Erm1}}{v_{Act2} - v_{Act1}}$$

Fig. 9-72: Dead time calculation

P_{Det1} , P_{Det2} : Determined touch probe position for measurement 1 and measurement 2

v_{Act1} , v_{Act2} : Actual axis velocity for measurement 1 and measurement 2

T_{Dead} : Dead time of touch probe input



The dead time can have a value of up to 128 cycles.

Example A touch probe is connected to an axis. The dead time should be determined for the positive and negative edge.

For the measurement, the parameters "M-0-0200 Dead time compensation, positive edge" and "M-0-0201 Dead time compensation, negative edge" are set to 0!

Subsequently, the respective axis is traveled with 10 rpm and the "M-0-0130 Measured value, positive edge" and the "M-0-0131 Measured value, negative edge" are recorded.

To reduce encoder noise as well as inaccuracies occurring due to the touch probe sensor scanning for example the measured values were recorded during several measurements. This resulted in an average value of 100° of M-0-0130 and an average value of 110° of M-0-0131.

The same measurement is repeated with 3000 rpm. This resulted in an average value of 101° of M-0-0130 and an average value of 112° of M-0-0131.

The dead time for the positive edge is calculated as follows:

Motion Functionality

$$T_{TotPos} = \frac{101^{\circ} - 100^{\circ}}{3000 \left[\frac{U}{\min} \right] - 10 \left[\frac{U}{\min} \right]} = \frac{1^{\circ}}{2990 \cdot \frac{360^{\circ}}{60s}} = 55.74ns$$

Fig. 9-73: Dead time calculation for positive edge

The dead time for the negative edge is calculated as follows:

$$T_{TotNeg} = \frac{110^{\circ} - 112^{\circ}}{3000 \left[\frac{U}{\min} \right] - 10 \left[\frac{U}{\min} \right]} = \frac{1^{\circ}}{2990 \cdot \frac{360^{\circ}}{60s}} = 111.48ns$$

Fig. 9-74: Dead time calculation for negative edge

Restarting or Disabling the Measured Value Acquisition

A single shot measurement or a continuous measurement restart is triggered when resetting the parameter for the touch probe enabling (1 $\Rightarrow$ 0) and setting it again (0 $\Rightarrow$ 1):

- M-0-0405, Enabled

Deleted are:

- M-0-0021, Status word
- M-0-0130, Measured value, positive edge
- M-0-0131, Measured value, negative edge
- M-0-0132, Difference value
- M-0-0140, Time stamp, positive edge
- M-0-0141, Time stamp, negative edge
- M-0-0224, Number of marker failures, positive edge
- M-0-0225, Number of marker failures, negative edge
- M-0-0409, Counter of measured values, positive edge
- M-0-0410, Counter of measured values, negative edge
- M-0-0411, Counter of difference values

Measured Value Status

Each measurement event generates a status information and increments a counter for each touch probe depending on the polarity of the respective measured value trigger. This is especially advantageous for the continuous measurement to identify new measurement events. This status information is displayed in the respective parameter:

- M-0-0021, Status word
- M-0-0409, Counter of measured values, positive edge
- M-0-0410, Counter of measured values, negative edge

9.10.6 Touch Probe Expectation Window

Use of the Expectation Window

The positioning range of an axis or a shaft, in which touch probe signal edges lead to a measured value acquisition, can be limited. In case of any limitation, measured value trigger signals are only expected in positioning values that can be set. Thus, this range is called "expectation window". The activation of

the measured value acquisition limited to the "expectation window" is carried out separately for the positive and negative edge in the parameter "M-0-0007 Control word".

Recording of marker failures with the activated "expectation window":

If the actual position value is outside the range the "expectation window", touch probe signal edges do not lead to a measured value acquisition!

If no "marker" was detected after a complete travel (exceeding of both position limits) which triggered a measured value, this state can be read in parameter "M-0-0224, Number of marker failures, positive edge" or "M-0-0225, Number of marker failures, negative edge". Therefore, the "marker failure monitoring" has to be enabled in parameter "M-0-0007, Control word"! If repeated, the value of M-0-0224 or M-0-0225 is incremented. If the value of M-0-0224 or M-0-0225 reaches a user-defined settable threshold (M-0-0206, Maximum number of marker failures"), a bit is set in the parameter "M-0-0021, Status word".

When registering a measured value trigger in the "expectation window", the value in the parameter M-0-0224 or M-0-0225 is deleted.

Expectation Window Setting

The limit values for the "expectation window" are set in the following parameters.

- M-0-0202, Starting position of the expectation window, positive edge
- M-0-0203, End position of the expectation window, positive edge
- M-0-0204, Starting position of the expectation window, negative edge
- M-0-0205, End position of the expectation window, negative edge

Marker Detection

The request to detect a "marker" is determined by activating a positive or negative signal edge for the touch probe. A "marker" is detected if the following event occurs during a complete measured signal processing via the "expectation window" with regard to the settings in the parameter "M-0-0007, Control word".

- A negative signal edge is detected in case of "activation of negative edge, touch probe"
- A positive signal edge is detected in case of "activation of positive edge, touch probe"
- A positive signal edge and a negative signal edge are detected in case of "activation of positive and negative edge, touch probe"

9.10.7 Overview on M-Parameters

M-0-0000, Dummy M-parameter

Function	A parameter serves as a placeholder; it does not have a function. A parameter can be described and read cyclically.
Use	Measure down time for writing and reading.

M-0-0001, Logical probe number

Function	The logical probe number can be read from this parameter. The number can deviate from the number of the input that is connected to the probe and can have values ranging from 1 to 100.

Motion Functionality

M-0-0002, Touch probe name

- Function** The descriptive text for the instance of the touch probe function can be saved in this parameter.
- The text does not provide any functional meaning.

M-0-0005, Input selection

- Function** By means of this probe, the digital input is selected that is to deliver the trigger signal for the probe function.

It is possible to assign the same input to various instances of the probe function so that different parameters can be simultaneously registered by means of one trigger signal.

IndraMotion MLC L25 / IndraLogic XLC L25

- Input 110..117 (digital inputs), 120..127(free configurable digital inputs) - fast I/O 1
- Input 210..217 (digital inputs), 220..227(free configurable digital inputs) - fast I/O 2

IndraMotion MLC L45/L65 / IndraLogic XLC L45/L65

- Input 1..8 - onboard module
- Input 110..117 (digital inputs), 120..127(free configurable digital inputs) - fast I/O 1
- Input 210..217 (digital inputs), 220..227(free configurable digital inputs) - fast I/O 2
- Input 310..317 (digital inputs), 320..327(free configurable digital inputs) - fast I/O 3
- Input 410..417 (digital inputs), 420..427(free configurable digital inputs) - fast I/O 4



1xx always represents the fast I/O module first found, not the PSI slot, 2xx represents the second fast I/O module, etc.

M-0-0006, Signal selection

- Function** Here, the selection of the parameter is carried out, whose current value for a signal edge is to be stored at the assigned input.

When selecting a parameter, the parameters for the measured values

- M-0-0130, Measured value, positive edge,
- M-0-0131, Measured value, negative edge and
- M-0-0132, Difference value

as well as the parameters for the expectation window limits

- M-0-0202, Starting position of the expectation window, positive edge,
- M-0-0203, End position of the expectation window, positive edge,
- M-0-0204, Starting position of the expectation window, negative edge and
- M-0-0205, End position of the expectation window, negative edge

automatically adopt the selected parameter's data type and unit.

If the signal selection is changed on applied enable, the enable signal is automatically reset.



The IDN of the selectable parameters are listed in "C-2499-2499, Probe, IDN list signal selection".

Axis parameters can only be selected for axes that are actually provided.

See also Functional Description "Touch Probe Function".

Use The readability of the selected parameters is only checked when the control unit is in operation mode. IDNs of non-readable parameters are not accepted. If the control unit is in P0/P2, the IDN is checked only. It cannot be determined, whether the axis the parameter belongs to is actually existing. The reading check is carried out

- with an activated touch probe function if the operation mode is switched or,
- if the control unit is in operation mode - as soon as the touch probe function is activated.

If the selected parameter cannot be read, the signal source is reset.

M-0-0007, Control word

Function By means of this parameter the probe function can be configured. The settings affect

- the activation of the signal edges,
- switching the measured value acquisition mode,
- the activation of the time stamp recording,
- the activation of the expectation window,
- the activation of the marker failure monitoring
- and the difference measurement mode.

Structure

Bit	Values/function
0	Activation of positive edge 0: positive edge not activated 1: positive edge activated
1	Activation of negative edge 0: negative edge not activated 1: negative edge activated
2	Enable mode 0: single measurement → after every measurement, the enable signal for the probe must be deleted and reset in order to enable another measurement. The measuring results are maintained until the next enable signal is applied. 1: continuous measurement → the measurements are continuously carried out for as long as the enable signal for the probe is set.
3	Enable time stamp 0: the time stamp is not registered 1: the time stamp is registered

Motion Functionality

Bit	Values/function
4	Activation of the probe window, positive edge 0: unrestricted range of measurement 1: range of measurement restricted to probe window (reserved)
5	Activation of the probe window, negative edge 0: unrestricted range of measurement 1: range of measurement restricted to probe window
6	Marker monitoring 0: not activated 1: activated
7	Extended difference measurement 0: not activated 1: activated
9, 8	Edge determination for extended difference measurement 00: difference from negative to negative edge 01: difference from positive to negative edge 10: difference from negative to positive edge 11: difference from positive to positive edge
10 - 11	Interpolation of the measured values 00: interpolation is not carried out The measured value corresponds to the current value at the point of time when the edge is at the input. 01: The measured value is interpolated in linear form from the current value and the subsequent value. 10: The measured value is determined from 3 values by means of cubic spline interpolation. 11: The measured value is determined from 4 values by means of cubic spline interpolation.
31 - 12	(reserved)



"Positive edge" means that the level at the probe input moves from 0 V to +24 V.

"Negative edge" means that the level at the probe input moves from +24 V to 0 V.

Even with the expectation window activated, the restriction of the measurement range is only effective if the start and end position of the expectation window are not identical.

See also Functional Description "Touch Probe Function".

M-0-0009, Osci: list of all M-parameters that can be oscilloscoped

Function This parameter contains the list of all M parameters which can be oscilloscoped.

M-0-0010, List of all M parameters

Function This list includes all M parameters. The entry is made in ID number format.

The entries are sorted according to identity number in ascending order.

Structure Following is the identity number format:

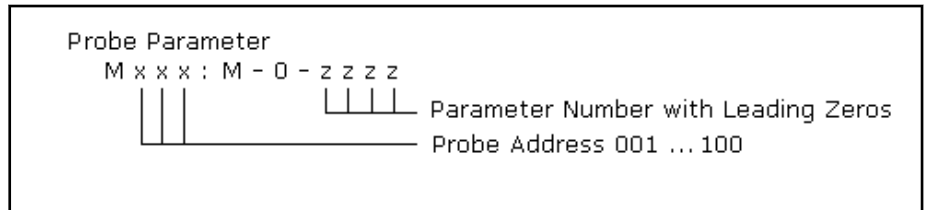


Fig.9-75: Identity number format

Use M001: M-0-0002 - Probe 1; Parameter M-0-0002.

M-0-0016, Project identification number

Function IndraWorks entered a series of unique characters in this parameter to identify worldwide the project. This series of characters is one to one for each project. This parameter has no influence on the functions of the Motion Logic Control.



This parameter is for internal purposes only.

M-0-0021, Status word

Function This parameter indicates the status of the measured value acquisition in the current cycle.

The status parameter includes information on the

- the availability of newly measured values,
- the exceedance of the marker failure threshold,
- and the overflow of touch probe events.

The parameter is reset with each touch probe enabling.

Structure

Bit	Name/function
0	Measured value at positive edge 0: no new measured value available 1: a new value was determined
1	Measured value at negative edge 0: no new measured value available 1: a new value was determined
2	Difference value calculation 0: no new difference value available 1: a new difference value was determined
3	Status of the marker failure monitoring 0: marker failure threshold not exceeded 1: marker failure threshold exceeded
4	Event overflow at positive edges 0: all positive edges could be detected 1: Various positive edges were detected in one sercos cycle, but only the first was processed.

Motion Functionality

Bit	Name/function
5	Event overflow at negative edges 0: all negative edges were detected 1: Various negative edges were detected in one sercos cycle, but only the first was processed.
15 - 6	(reserved)



The status of the marker failure monitoring is set in relation to the parameterized limit value for the marker failure "M-0-0224, Number of marker failures, positive edge, touch probe". It is reset as soon as the expectation window has been passed without marker failure.

If the maximum number of the marker failures is parameterized with "0", exceeding the limit value is not monitored.

Also refer to the Functional Description "Touch Probe Function".

M-0-0024, Command: Activation

Function The touch probe function is enabled using this input.



The touch probe function can only be enabled if the touch probe function is activated.

The parameter's data type and unit are only adapted for the measured values and the expectation window limits if the touch probe function is activated.

The touch probe function can only be activated if the values of parameter C-0-0400 and C-0-0503 are identical.

M-0-0130, Measured value, positive edge

Function In this parameter, the value of the selected parameter ("M-0-0006, signal selection touch probe") is saved at the time of the positive edge at the entrance of the touch probe. When the release is granted, the measurement value is deleted.

- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the positive edge by setting bit 0 in parameter "M-0-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".



The counter for the measured values (parameter "M-0-0409, Counter of measured values, positive edge") is incremented in the same cycle in which the measured value is output.

M-0-0131, Measured value, negative edge

Function In this parameter, the value of the selected parameter ("M-0-0006, Signal selection") is saved at the time of the negative edge at the entrance of the touch probe. When the release is granted, the measurement value is deleted.

- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",

- Activating the negative edge by setting bit 1 in parameter "M1-0007, Control word",
- Apply enable signal by setting "M-0-0405, Enabling".



The counter for the measured values (parameter "M-0-0410, Counter of measured values, negative edge") is incremented in the same cycle in which the measured value is output.

M-0-0132, Difference value

- Function** In this parameter, the difference value of two measured values is saved.
- For a normal difference measurement, the difference value is the difference between the last measured value with a positive edge and the last measured value with a negative edge.
- If an extended difference measurement is configured, various combinations are possible:
- last measured value with a positive edge - next to last measured value with a negative edge,
 - last measured value with a positive edge - next to last measured value with a positive edge,
 - last measured value with a negative edge - next to last measured value with a positive edge,
 - last measured value with a negative edge - next to last measured value with a negative edge.
- Since a difference value is not formed with each new measured value, there is a separate counter "M-0-0411, Counter of difference values" for measuring differences.
- The difference value is deleted with each release.
- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating at least one negative edge by setting bit 1 to bit 0 in parameter "M1-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".

M-0-0140, Time stamp, positive edge

- Function** The time stamp for the measured value positive edge is saved in this parameter.
- The datum has the same format as parameter "C-0-0050, System time".
- The value is deleted with each release.
- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the positive edge by setting bit 0 in parameter "M-0-0007, Control word",
 - Activating the time stamp recording by setting bit 3 in parameter "M1-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".

M-0-0141, Time stamp, negative edge

- Function** The time stamp for the measured value negative edge is saved in this parameter.
- The datum has the same format as parameter "C-0-0050, System time".

Motion Functionality

The value is deleted with each release.

- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the negative edge by setting bit 1 in parameter "M1-0007, Control word",
 - Activating the time stamp recording by setting bit 3 in parameter "M1-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".

M-0-0200, Dead-time compensation, positive edge

- Function** The value of this parameter defines the reaction time to be compensated for the positive edge (output in μs).
Time delays arising through I/Os or through the sensor used can be compensated



The reaction time can have a maximum value of 128 cycles.

M-0-0201, Dead-time compensation, negative edge

- Function** The value of this parameter defines the reaction time to be compensated for the negative edge (output in μs).
Time delays arising through I/Os or through the sensor used can be compensated



The reaction time can have a maximum value of 128 cycles.

M-0-0202, Starting position of the expectation window, positive edge

- Function** This parameter determines the lower limit value of the expectation window for the measured value acquisition of the positive edge. The expectation window restricts the range where touch probe signal edges cause a measured value acquisition.

In order to activate the measurement with the expectation window, bit 5 must be set in parameter "M-0-0007, Control word".



If the value of the start position of the expectation window is higher than that of the end position, the expectation window exceeds the modulo value.

The expectation window can only be used if modulo format is parameterized for the selected axis (cf. "A-0-0045, Modulo value").

M-0-0203, End position of the expectation window, positive edge

- Function** This parameter determines the upper limit value of the expectation window for the measured value acquisition. The expectation window restricts the range where touch probe signal edges cause a measured value acquisition.

In order to activate the measurement with the expectation window, bit 5 must be set in parameter "M-0-0007, Control word".



If the value of the end position of the expectation window is smaller than that of the start position, the expectation window exceeds the modulo value.

The expectation window can only be used if modulo format is parameterized for the selected axis (cf. "A-0-0045, Modulo value").

M-0-0204, Starting position of the expectation window, negative edge

Function This parameter determines the lower limit value of the expectation window for the measured value acquisition of the negative edge. The expectation window restricts the range where touch probe signal edges cause a measured value acquisition.

In order to activate the measurement with the expectation window, bit 5 must be set in parameter "M-0-0007, Control word".



If the value of the start position of the expectation window is higher than that of the end position, the expectation window exceeds the modulo value.

The expectation window can only be used if modulo format is parameterized for the selected axis (cf. "A-0-0045, Modulo value").

M-0-0205, End position of the expectation window, negative edge

Function This parameter determines the upper limit value of the expectation window for measured value acquisition of the negative edge. The expectation window restricts the range where touch probe signal edges cause a measured value acquisition.

In order to activate the measurement with the expectation window, bit 5 must be set in parameter "M-0-0007, Control word".



If the value of the end position of the expectation window is smaller than that of the start position, the expectation window exceeds the modulo value.

The expectation window can only be used if modulo format is parameterized for the selected axis (cf. "A-0-0045, Modulo value").

M-0-0206, Maximum number of marker failures

Function This parameter is used to define the registration threshold for the registered number of marker failures.

If the limit value is set to "0", exceeding the limit value is not monitored.

The number of marker failures completed consecutively is displayed in parameter "M-0-0224, Number of marker failures, positive edge".

If the number of marker failures reaches the specified limit value, bit 3 of "M-0-0021, Status word" is set.

M-0-0224, Number of marker failures, positive edge

Function There is the option to define an expectation window for the touch probe where the touch probe edges must be located in order to be evaluated.

When the "expectation window" is active, a "marker failure monitoring" can be activated in "M-0-0007, Control word". This marker failure monitoring registers the passing of the expectation window for the positive edge without a marker been registered, and increments the value of "M-0-0224, Number of marker failures, positive edge, touch probe".

The content of the parameter is reset to "0" if

- an edge has been registered when passing the expectation window,
- the marker failure monitoring has been deactivated by resetting bit 6 in "M-0-0007 Control word",
- the touch probe function is enabled in "M-0-0405, Enabling" touch probe 1.

Motion Functionality



The edges activated in "M-0-0007, Control word" are called markers.

A marker error is present as soon as the expectation window has been passed and an activated edge has not been registered.

Marker failure monitoring is only possible if modulo format is set for the selected axis (cf. "A-0-0045, Modulo value").

See also Functional Description "Touch Probe Function".

M-0-0225, Number of marker failures, negative edge

Function There is the option to define an expectation window for the touch probe where the touch probe edges must be located in order to be evaluated.

When the "expectation window" is active, a "marker failure monitoring" can be activated in "M-0-0007, Control word". This marker failure monitoring registers the passing of the expectation window for the negative edge without a marker been registered, and increments the value of "M-0-0224, Number of marker failures, positive edge, touch probe".

The content of the parameter is reset to "0" if

- an edge has been registered when passing the expectation window,
- the marker failure monitoring has been deactivated by resetting bit 6 in "M-0-0007 Control word",
- the touch probe function is enabled in "M-0-0405, Enabling" touch probe 1.



The edges activated in "M-0-0007, Control word" are called markers.

A marker error is present as soon as the expectation window has been passed and an activated edge has not been registered.

Marker failure monitoring is only possible if modulo format is set for the selected axis (cf. "A-0-0045, Modulo value").

See also Functional Description "Touch Probe Function".

M-0-0401, Input status

Function The switching status of the probe input signal is indicated in this parameter.

M-0-0405, Enabling

Function By means of this parameter the enable signal of the measurement with the probe is applied. From the point of time of the enabling the signal edges are registered at the assigned input and the parameter values are recorded.



The touch probe function can only be enabled if the values of parameter C-0-0400 and C-0-0503 are identical.

Structure

Value	Release of the touch probe	Comment
0	No	
1	Yes	

Fig. 9-76: Relevant bits of M-0-0405, Enabling

Use As soon as the enabling signal is applied, the following parameters are deleted:

- M-0-0021, Status word,
- M-0-0130, Measured value, positive edge,
- M-0-0131, Measured value, negative edge,
- M-0-0132, Difference value
- M-0-0140, Timestamp, positive edge,
- M-0-0141, Timestamp, negative edge,
- M-0-0224, Number of marker failures, positive edge.
- M-0-0225, Number of marker failures, negative edge,
- M-0-0409, Counter of measured values, positive edge,
- M-0-0410, Counter of measured values, negative edge,
- M-0-0411, Counter of difference values.



If single measurement is parameterized and no edge is activated, the enable signal is reset immediately, as the condition "every activated edge has been registered once" it is fulfilled.

M-0-0409, Counter of measured values, positive edge

- Function** In this parameter, the number of measurements which were triggered by positive signal edges at the selected entrance is displayed. With every release, the counter is set to "0".
- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the positive edge by setting bit 0 in parameter "M-0-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".

M-0-0410, Counter of measured values, negative edge

- Function** In this parameter, the number of measurements which were triggered by negative signal edges at the selected entrance is displayed. With every release, the counter is set to "0".
- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the negative edge by setting bit 1 in parameter "M1-0007, Control word",
 - Apply enable signal by setting "M-0-0405, Enabling".

M-0-0411, Counter of difference values

- Function** This parameter indicates the number of the registered difference values.
If the extended difference measurement is activated by setting bit 11 of parameter "M-0-0007, Control word", the difference is not calculated each time a measured value is output. Instead, the difference is only calculated if the edge matches the configured "difference to" edge.
With every release, the counter is set to 0.
- Use**
- Activating the probe function through parameter "M-0-0024, Command: Activation",
 - Activating the negative edge by setting bit 1 in parameter "M1-0007, Control word"
 - Apply enable signal by setting "M-0-0405, Enabling".

Motion Functionality

9.10.8 Function Blocks for Touch Probes

The touch probe functionality of the control hardware is provided via the following function blocks of the ML_PLCOpen library:

Function block	Description
ML_InitTouchProbe	Enables and configures the touch probe in the control
ML_WriteExpectWindow	Configures the expectation window for the touch probe functionality
ML_TouchProbe	Enables the selected touch probe, analyzes the status and provides the measured values after the trigger event
ML_TouchProbeContinuous	Enables the selected touch probe, analyzes the status and provides the measured values after the trigger event
ML_AbortTrigger	Cancels an active measurement

Fig.9-77: Function blocks for the touch probe functionality of the control hardware

This function block is described in /23/, [Required and Supplementing Documentations, page 14](#).

9.11 Programmable Limit Switch

The programmable limit switch generates binary switching signals depending on the position of an axis.

The switches of a programmable limit switch can be triggered independently of the direction of rotation of the axis.

- Triggering in a positive direction of rotation
- Triggering in a negative direction of rotation
- Triggering in both directions of rotation

Moreover, switches can be operated in two different switch track modes:

- Position-related mode: The rising and/or falling edge of the active switch, along with the correction times and distances, determine the status of the switch track.

Time-related mode: The switch remains active for a specified time based on the rising edge of the switch and possible corrections. If a slave switch becomes active during this time, the active time of the switch is extended accordingly.

To compensate for internal processing times and for delay in the device connected, a lead time for switching on and off each switch track can be defined for every switch track.

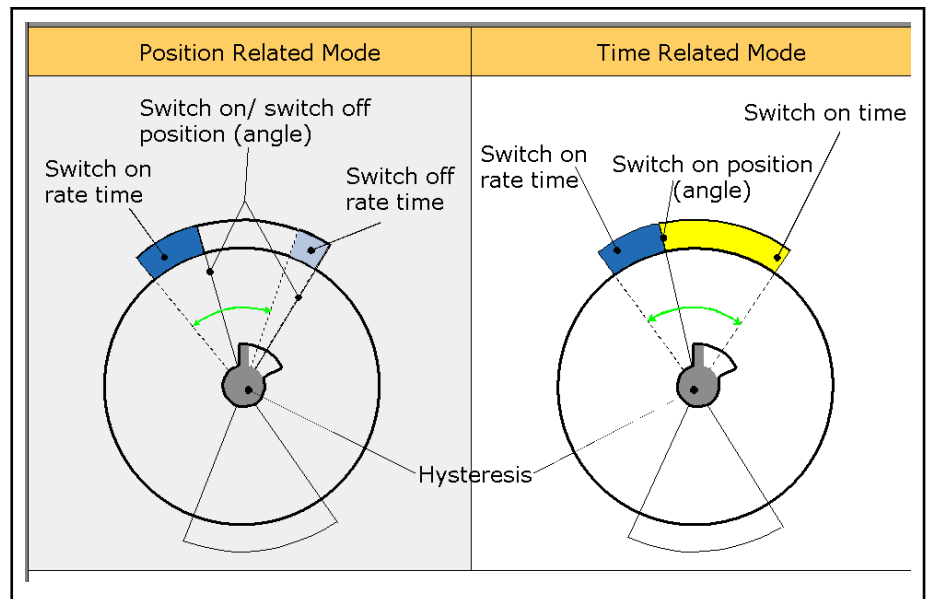


Fig.9-78: Operating states

The programmable limit switch is available both as hardware and as software.

Software-based programmable limit switch

The software-based programmable limit switch is implemented by the function block MC_DigitalCamSwitch in the PLC library ML_TechBase. This function block enables up to 32 switches on eight binary outputs per axis.

The resolution of this software-based solution depends on the cycle time of the PLC task in which the function block runs. A task with 2 ms of cycle time can, for instance, resolve a switch with 12° at 1,000 rpm.

For a detailed description of the function block MC_DigitalCamSwitch, refer to /30/, [Required and Supplementing Documentations, page 14](#).

Hardware-based programmable limit switch

The hardware-based programmable limit switch is a function module connected on the left side of the control. Each of the four function modules allows 16 binary outputs. It is parameterized via the Programmable Limit Switch dialog directly in the function module. The resolution is 125 μ s.

The function module is described in detail in /24/, [Required and Supplementing Documentations, page 14](#)).

10 Kinematics

10.1 General Information

Predefined kinematics (Cartesian, Scara, Deltapack or user-defined kinematics) can be created in the "Robot" folder. Up to 16 kinematics with a total of

- 16 axes for an IndraMotion MLC L25
- 32 axes for an IndraMotion MLC L45
- 64 axes for an IndraMotion MLC L65

including auxiliary axes (belt) and orientation axes.

The user-defined kinematics can only be selected and parameterized via an assigned kinematic number.

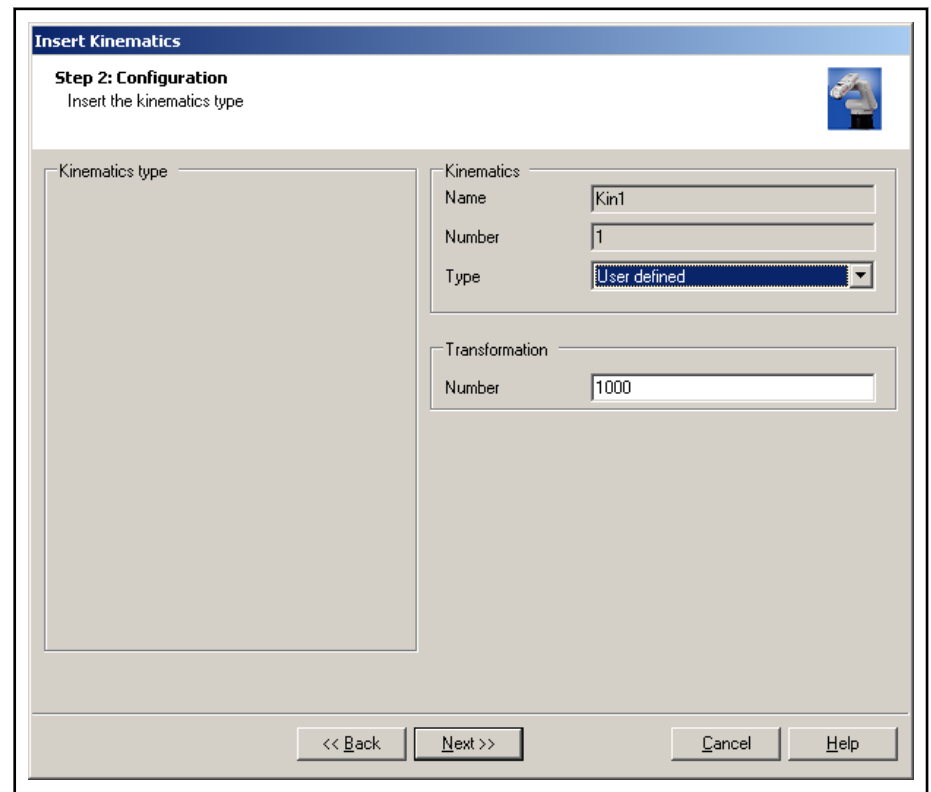


Fig. 10-1: Options for user-defined kinematics

Programming environment

RobotControl is the programming environment for the Bosch Rexroth robot family. The advantages are:

- Faster implementation of robot applications
- Visualization of the implementation
- Diagnostics of the implementation

Characteristics

The programming environment contains the following components:

- **User interface**

The user interface is oriented at the Windows applications and thus allows the user to work intuitively and effectively.

All RobotControl components are integrated into the interface and can be optionally shown or hidden. All toolbar icons are provided with help

Kinematics

texts. These are displayed when holding the mouse cursor on the icon for a moment.

Additionally, information on the respective functions is shown in the status bar

- **Text editor**

The text editor is used to create and check user programs in the programming language RCL (**R**obot **C**ontrol **L**anguage). The text editor can also be used for any other ASCII text file

- **Compiler and Debugger**

The compiler compiles a source code to a machine-readable code. The output texts and the error messages are redirected to an output window of the RobotControl and displayed. Double-click to jump to the position in the program.

The debugger facilitates the finding of program errors. The values of previously declared variables can be displayed in the debug mode and monitored during the program execution

- **Point editor**

The point editor enters, deletes or modifies points. The points are loaded from the point file (online or offline) and represented in a table. Points can also be declared as point arrays

- **File transfer**

Files can be transferred to the control and deleted in the control

- **Control diagnostics**

All relevant status information of the respective kinematics is displayed in the "Status" dialog

10.2 Menu Structure of the Kinematics

10.2.1 Menu Item: Robot

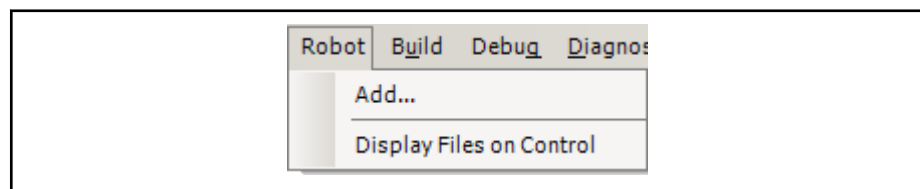


Fig. 10-2: Menu Item: Robot

Select the "Robot" node to enable the menu item **Robot** in the IndraWorks main menu. Start the wizard to configure a new kinematics via **Add...** Display the RCL programs in the control via **Display Files on Control**. Open the previously described functions also via the context menu of the "Robot" node.

10.2.2 Menu item: Kinematics

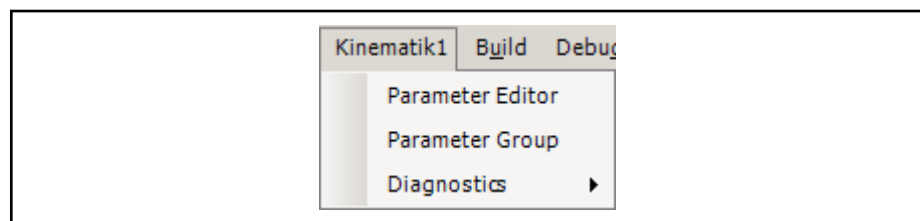


Fig. 10-3: Menu Item of a Kinematics (in the example "Kinematik1")

The menu item (in the example **Kinematik1** in the IndraWorks main menu is activated by selecting the a node of a kinematics. The name of the respectively created selected kinematics is displayed as name in the menu item or node.

10.2.3 Menu Item: Programs

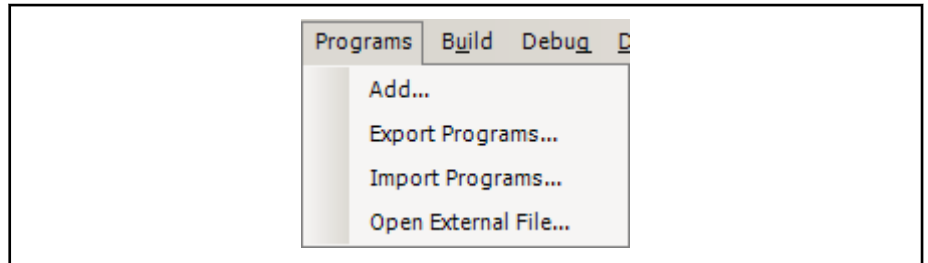


Fig. 10-4: Menu Item: Programs

Select the "Programs" node to enable the menu item **Programs** in the IndraWorks main menu.

Create a new RCL program via **Add...**

Store all RCL programs outside the project via **Export Programs....**

All RCL programs outside the project can be imported into the project via **Import Programs....**

Open an RCL program outside the project in the editor via **Open External File....**

10.2.4 Menu item of a Program

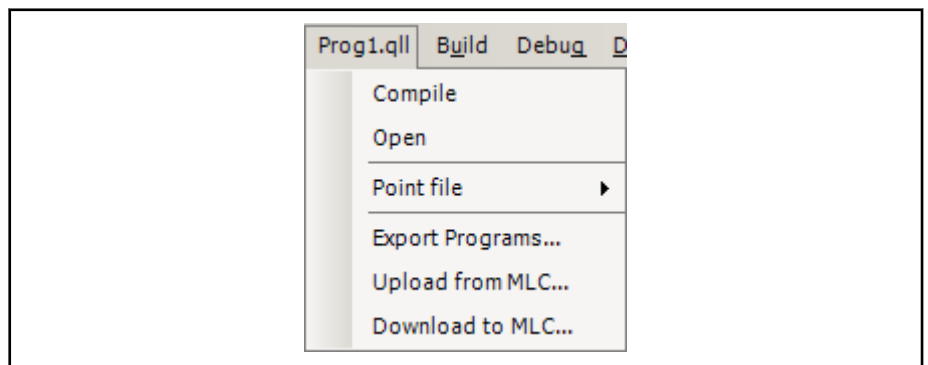


Fig. 10-5: Menu item of a program (in the example "Prog1.qll")

By selecting a program in the project (below the node Robot/<Kinematic-Name>/Programs), the menu item of the program (in the example **Prog1.qll**) is enabled in the IndraWorks main menu. The following menu items are available:

Create an executable robot program in the source text (.QLL) via **Compile**,

Open the selected RCL program in the editor via **Open....**

With **Point File**, and a respective sub menu, a point file can be opened or stored via the submenus of a point file.

Store the selected RCL program outside the project via **Export Programs....**

Upload the selected RCL program from the control to the project via **Upload from MLC....**

Download the selected RCL program to the control via **Download to MLC....**

Kinematics

10.3 Kinematics - Context menu

10.3.1 Overview

The context menu of a kinematics comprises the following menu items:

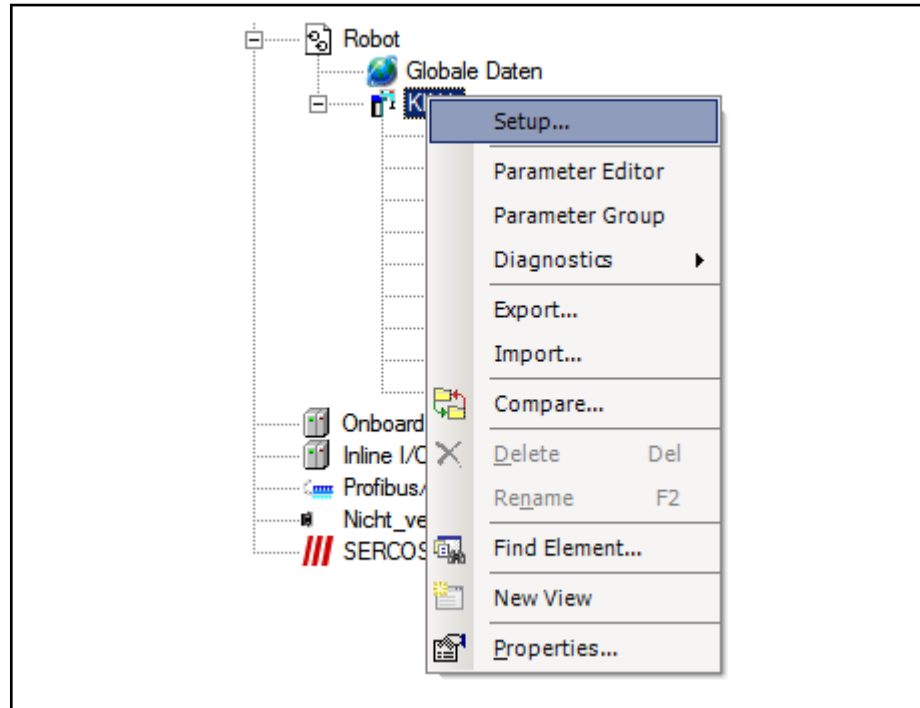


Fig. 10-6: Context menu of a kinematics in online mode

- **Setup...** - Commissioning cycle of the kinematics, see [Setup](#), page 448
- **Parameter editor** - Edits parameters, see [Parameter Editor](#), page 133
- **Parameter group** - Edits multiple parameters, see [Parameter Group](#), page 135
- **Diagnostics** - Diagnostic functions of the kinematics, see [Diagnostics](#), page 449
- **Export...** - Saves robot elements in a file, see [Exporting](#), page 172
- **Import...** - Imports robot elements from a file, see [Importing](#), page 174
- **Compare...** - Compares and merges objects, see [Comparing/Merging](#), page 176
- **Find element...** - Allows searching for elements below the node
- **New view** - Displays the kinematics in an individual window
- **Properties...** - Provides information on the kinematics (such as name or type)

10.3.2 Setup

To set up a kinematics, the following dialogs are processed by a wizard:

- For axis lengths and coupling, see [Axis Lengths and Coupling](#), page 450
- For motion limit values, see [Motion Limit Values](#), page 452
- For belt coupling factors, see [Belt Coupling Factors](#), page 452
- For the coordinate offset, see [Coordinate Offset](#), page 453
- For jogging, see [Jogging](#), page 454

- For safe zones, see [Safe Zones, page 456](#)
- For workpiece coordinate assignment, see [Workpiece Coordinate Assignment, page 458](#)

10.3.3 Diagnostics

This dialog box graphically displays the status information of the respective kinematics. The upper part of the window displays general information on the kinematics.

The middle part displays the kinematic-specific information on all axes of a kinematics in a table. The information comprises the current position, the programmed end point (only in automatic mode) and the lagging of the individual axes.

The lower part of the window displays the kinematic status in detail.

Kinematics number	Kinematics name	Kinematics type
1	Kinematik 1	Carthesis

Status: Kinematik ist in 'Standstill Incomplete'

Axis No.	Axis name	Position[mm]	End position[mm]	InPos	Lag[mm]
1	A01	0	0	☐	0
2	A02	0	0	☐	0

Coordinate system:
☒ Joint coordinates (JC)
☐ World coordinates (WC)

Status word: Standstill is not complete

Power:
☐ Axes ready ("AH/AF")
☐ Feed enable

Errors/Warnings:
☐ Warning exists
☐ Error exists

Messages:
☒ Automatic
☐ Manual
☐ Target position reached
☐ Block processing enabled
☐ Error reaction is active

Details >>

Fig. 10-7: Diagnostics of a kinematics

Coordinate system

The setting under "Coordinate system" specifies the coordinate system in which the axis values or belt values are to be displayed.

There are two settings:

- **JC** = Joint coordinates - They describe the individual axis positions in [°] (rotary axes) or [mm] (linear axes)

Axis No.	Axis name	Position[°]	End position[°]	InPos	Lag[°]
1	A01	0	0	☐	0
2	A02	0	0	☐	0

Fig. 10-8: Display in joint coordinates

- **WC** = World coordinates - They describe a freely definable local world coordinate system, also called workpiece coordinate system

Default setting: Coordinate origin at the base point of the robot.

Kinematics

Axis No	Axis name	Position[mm]	End position[mm]	InPos	Lag[mm]
1	X1	2	0	☐	0
2	Y1	0	0	☐	0

Fig. 10-9: Display in world coordinates

List of axis coordinates

The list of the axis coordinates shows

- Axis number
- Axis name
- Axis position
- End point
- Lag

An enabled checkbox in the "InPos" provides information whether the respective axis is "in Position".

10.4 Kinematics – Dialogs

10.4.1 Overview on dialogs

When opening the folder of the kinematics in the Project Explorer, individual dialogs - display of data from the kinematics and adjustment possibilities for kinematic parameters - are provided. These dialogs are disabled at first and enabled when the project is switched to online mode.

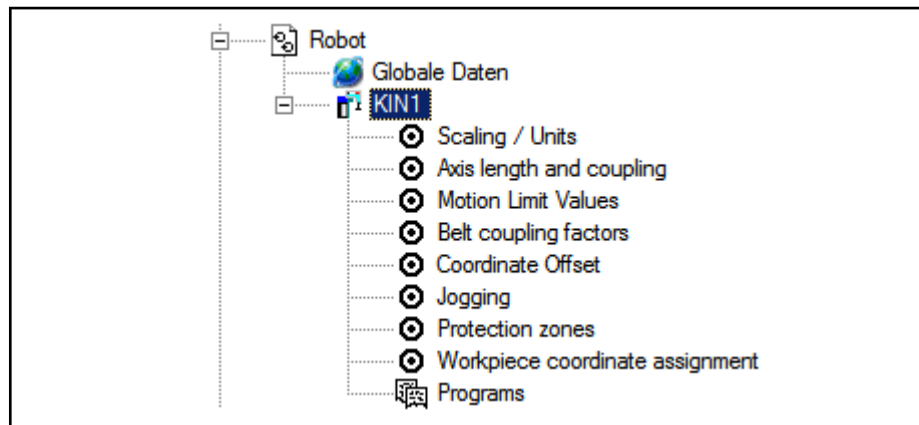


Fig. 10-10: Kinematic dialogs

- For scaling/units, see [Settings: Scaling/Units, page 277](#)
- Axis length and coupling, see [Axis Lengths and Coupling, page 450](#)
- For motion limit values, see [Motion Limit Values, page 452](#)
- For belt coupling factors, see [Belt Coupling Factors, page 452](#)
- For the coordinate offset, see [Coordinate Offset, page 453](#)
- For jogging, see [Jogging, page 454](#)
- For safe zones, see [Safe Zones, page 456](#)
- For workpiece coordinate assignment, see [Workpiece Coordinate Assignment, page 458](#)
- Programs, see [Creating a New RCL Program, page 459](#)

10.4.2 Axis Lengths and Coupling

Set the transformation lengths and transformation factors required to transform axis positions into Cartesian coordinates in this dialog.

Kinematics

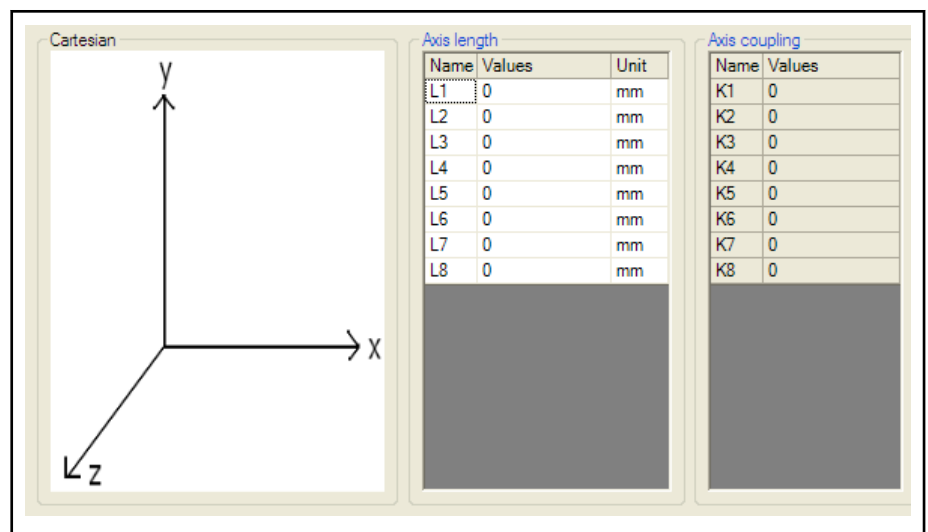


Fig. 10-11: Defining axis lengths and axis couplings

The following parameters are evaluated:

- K-0-0040, Transformation, lengths of the mechanics
- K-0-0041, Transformation, factors

Kinematics

10.4.3 Motion Limit Values

The following limit values can be entered:

- Maximum path velocity
- Axis monitoring
- Software limit switch, positive WC
- Software limit switch, negative WC

The dialog box provides an overview on the preset motion limit values of the kinematics.

Motion Limit Values - Kinematik		
Maximum Path Speed	10	mm/s
WC axis monitoring	In Linear mode	
Software limit switch		
	Positive	Negative
Limit value 1	1 mm	-1 mm
Limit value 2	1 mm	-1 mm
Limit value 3	1 mm	-1 mm
Limit value 4	1 mm	-1 mm
Limit value 5	1 mm	-1 mm
Limit value 6	1 mm	-1 mm
Limit value 7	1 mm	-1 mm
Limit value 8	1 mm	-1 mm

Fig. 10-12: Overview on motion limit values

The following parameters are evaluated:

- K-0-0003, Maximum path velocity
- K-0-0036, WC axis monitoring
- K-0-0005, Positive software limit switch WC
- K-0-0006, Negative software limit switch WC

10.4.4 Belt Coupling Factors

The belt coupling factors specify how belt position changes of the respective belt affect coordinate changes in the program sections between SYNCHRON and SYNCHRON_END.

The dialog box provides an overview on the preset belt coupling factors of the kinematics.

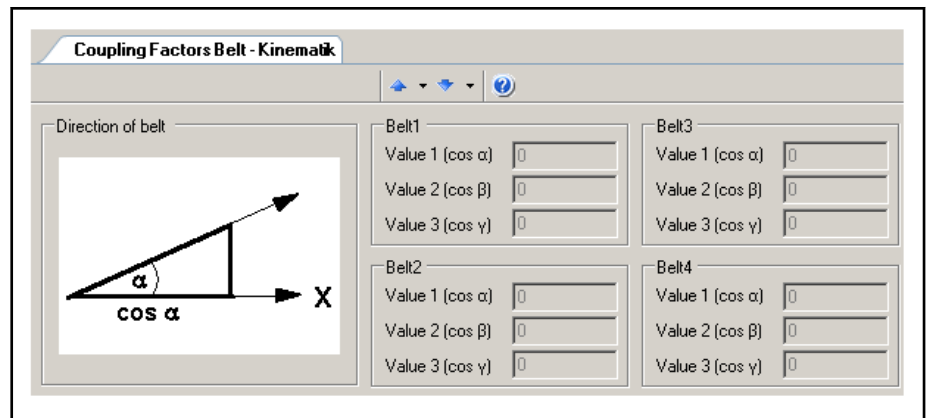


Fig. 10-13: Overview on the belt coupling factors

The following parameters are evaluated:

- K-0-0071, Coupling factors, belt 1
- K-0-0072, Coupling factors, belt 2
- K-0-0073, Coupling factors belt 3
- K-0-0074, Coupling factors, belt 4

10.4.5 Coordinate Offset

With this dialog, the coordinate system origin of the transformation schema can be shifted in order to work with the coordinate values corresponding to a reference point in the editing cell.

The dialog box provides an overview on the preset coordinate offset of the kinematics.

Kinematics

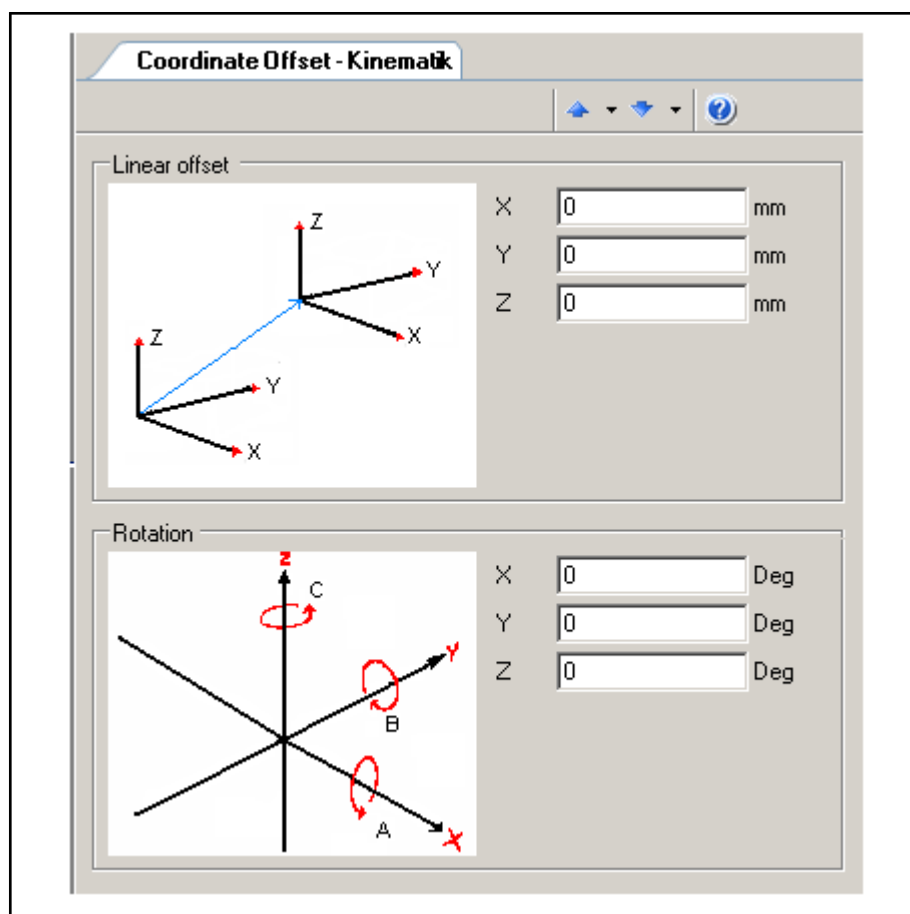


Fig. 10-14: Overview on the coordinate offset

The following parameter is evaluated:

- K-0-0042, World coordinate system offset

10.4.6 Jogging

In this dialog, the velocities settings for the manual operation can be made.

The screenshot shows a software window titled "Joggen - Kinematik". Inside, there is a "Jogging" section with a "Coordinate" dropdown menu set to "y1". Below this, there are four main parameter groups:

- Distance:**
 - Short: 1 mm
 - Long: 10 mm
- Velocity:**
 - Fast: 25 mm/s
 - Slow: 10 mm/s
- Acceleration:**
 - Acceleration: 370 mm/s²
 - Jerky acceleration movements: 0 mm/s³
- Braking deceleration:**
 - Braking deceleration: 370 mm/s²
 - Jerky braking movement: 0 mm/s³

Fig. 10-15: Overview on jogging

The following parameters are evaluated:

Coordinates

- Selection of coordinates (axis) for which the corresponding jog values are to be displayed

Distance

- K-0-0080, Jogging, distance short in WC system
This parameter provides the short distance for the incremental jogging in the world coordinate system
- K-0-0081, Jogging, distance long in WC system
This parameter provides the long distance for the incremental jogging in the world coordinate system

Velocity

- K-0-0082, Jogging, velocity low in WC system
This parameter provides the low velocity for continuous jogging in the world coordinate system
- K-0-0083, Jogging, velocity high in WC system
This parameter provides the fast velocity for continuous jogging in the world coordinate system

Acceleration

- K-0-0084, Jogging, acceleration in WC system
The acceleration for the jogging (incremental and continuous) in the world coordinate system is stored in this parameter
- K-0-0086, Jogging, acceleration jerk in WC system
The maximum acceleration jerk for jogging (incremental and continuous) in the world coordinate system is stored in this parameter (indirect definition of the acceleration change time)

Kinematics

Deceleration

- K-0-0085, Jogging, deceleration in WC system
The deceleration for the jogging (incremental and continuous) in the world coordinate system is stored in this parameter
- K-0-0087, Jogging, deceleration jerk in WC system
The maximum deceleration jerk for jogging (incremental and continuous) in the world coordinate system is stored in this parameter (indirect definition of the deceleration change time)

10.4.7 Safe Zones

This dialog allows setting the safe zones. Safe zones are areas blocked for the robot motion.

Protection zones							
Zone	Active	Minimum Value			Maximum Values		
		X [mm]	Y [mm]	Z [mm]	X [mm]	Y [mm]	Z [mm]
1	☒	0	0	0	0	0	0
2	☒	0	0	0	0	0	0
3	☐	0	0	0	0	0	0
4	☐	0	0	0	0	0	0
5	☐	0	0	0	0	0	0
6	☐	0	0	0	0	0	0
7	☐	0	0	0	0	0	0
8	☐	0	0	0	0	0	0
9	☐	0	0	0	0	0	0
10	☐	0	0	0	0	0	0
11	☐	0	0	0	0	0	0
12	☐	0	0	0	0	0	0
13	☐	0	0	0	0	0	0
14	☐	0	0	0	0	0	0
15	☐	0	0	0	0	0	0
16	☐	0	0	0	0	0	0
17	☐	0	0	0	0	0	0
18	☐	0	0	0	0	0	0
19	☐	0	0	0	0	0	0
20	☐	0	0	0	0	0	0
21	☐	0	0	0	0	0	0
22	☐	0	0	0	0	0	0
23	☐	0	0	0	0	0	0
24	☐	0	0	0	0	0	0
25	☐	0	0	0	0	0	0
26	☐	0	0	0	0	0	0
27	☐	0	0	0	0	0	0
28	☐	0	0	0	0	0	0
29	☐	0	0	0	0	0	0
30	☐	0	0	0	0	0	0
31	☐	0	0	0	0	0	0
32	☐	0	0	0	0	0	0

Fig. 10-16: Specifying the safe zones

The following parameters are evaluated:

- K-0-0155, Safe zone on/off
- K-0-0156, Safe zone 1 min. point
- K-0-0157, Safe zone 1 max. point

The parameter for the safe zones 2 - 32 are numbered consecutively. A maximum of 32 safe zones can be parameterized for each kinematics.

Functional description: A motion command in the RCL program consists of a starting point A and the programmed end point B.

In the block preparation, a linear distance AB is interpolated from the points A and B. It is checked additionally, whether the distance AB has one intersection or one boundary point in common with one side of the safe zone. This is checked for each motion command and for all workspaces assigned to the kinematics.

If the block preparation identifies a violation of a safe zone in the next block, the kinematics is traveled up to the next starting point of the incorrect block and then, the RCL program is aborted. The error message "F12C0433 Workspace violation" is also output.

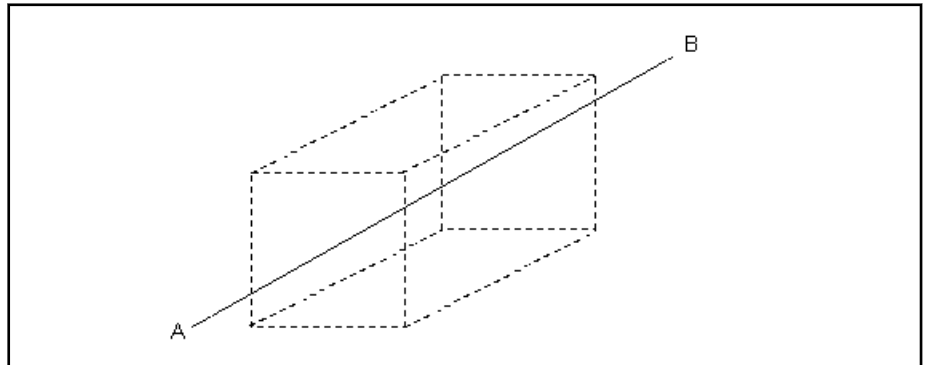


Fig. 10-17: Safe zone with minimum point (A) and maximum point (B)

Conditions: When programming the safe zones, the following has to be considered:

- Each kinematics considers only its individual safe zones
- The safe zones consider only the TCP of the robot, not the mechanics
- In case of the PTP interpolation, the monitoring of the safe zones is strictly restricted since the distance to be traveled is not defined. Deviations from the linear distance AB are not detected
- In case of a CIRCULAR interpolation, the monitoring of the safe zones is strictly restricted. The checking of the safe zones is restricted on the linear distance AB as well. The circular path and the programmed inter-point are not checked
- When blending with VIA or APPROX, deviations from the linear distance AB are not detected. Deviations due to blending radii are not considered
- The safe zone monitoring is only effective in Automatic mode. The safe zones are not monitored in Manual mode
- In case of a belt-synchronous travel motion, the belt-synchronous motion part is not considered when checking the safe zones
- Since the safe zones are checked in the block preparation, safe zones cannot be dynamically adapted. A change of the safe zones becomes only effective if the block buffer was processed. Therefore, a change or activation / deactivation of a safe zone can delay when the RCL program is running
- The maximum point has to be bigger than the minimum point to ensure that the safe zone check is not active

Use: The activation and adaptation of the safe zone is executed via the parameters K-0-0155 to K-0-0219.

The K-parameters can be switched via the respective PLC functions blocks or directly via the parameter editor to P4 (BB RUN).

Switching to P2 is not necessary.

The parameter K-0-0155 activates the checking of the safe zones (from bit 1 for 1st safe zone up to bit 32 for 32nd safe zone).

Kinematics

Example: K-0-0155 0000 0000 0000 0000 0000 0000 0100 $\equiv$ decimal: 4 $\Rightarrow$ safe zone 3 is active.

The following parameters K-0-0156 to K-0-0219 can be used to specify the maximum and minimum points by which cuboids are defined as safe zones.

Example: K-0-0155 0000 0000 0000 0000 0000 0000 0100 $\equiv$ decimal: 4 $\Rightarrow$ safe zone 3 is active.

K-0-0156 minimum point of safe zone 1 (0, 0, 0)
K-0-0157 maximum point of safe zone 1 (20, 10, 10)
K -0-0158 minimum point safe zone 2 (.....)
K -0-0159 maximum point safe zone 2 (.....)
.....
K -0-0219 maximum point safe zone 32 (.....)

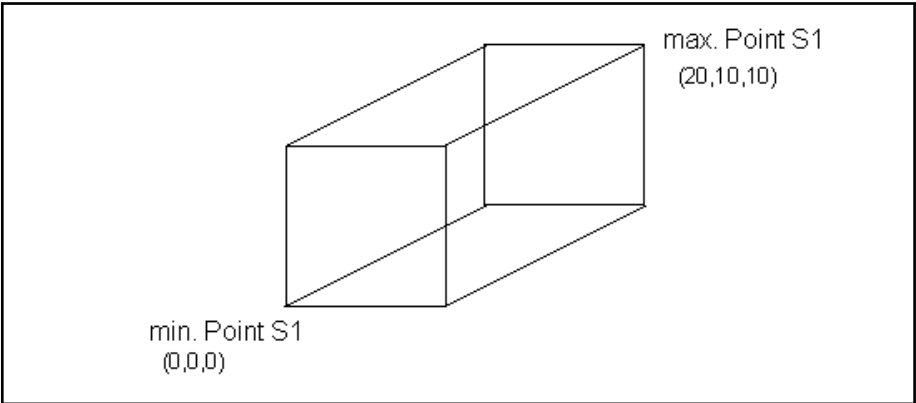


Fig.10-18: Maximum and minimum point of safe zone 1

10.4.8 Workpiece Coordinate Assignment

This dialogs allows assigning axis coordinates to workpiece coordinates.

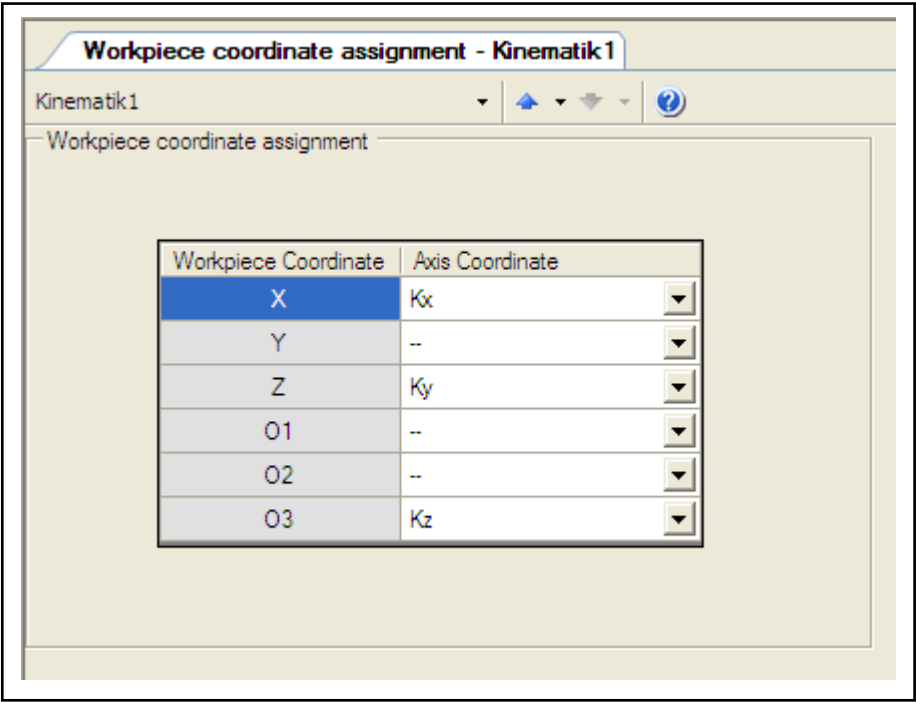


Fig.10-19: Specifying the workpiece coordinate assignment

The following parameters are evaluated:

- K-0-0043, Assignment list of workpiece coordinates

10.5 RCL Editor

10.5.1 General Information

The text editor is used to create, check and debug user programs in the programming language RCL (**R**obot **C**ontrol **L**anguage).

Options can be used to set the behavior, appearance, colors, line numbering, tabulator widths, indentations, etc. (see [RoCo-Editor](#), page 469).

10.5.2 Characteristics

Special editor characteristics:

- Syntax highlighting (can be freely configured in the "Options" dialog)
- Copy, Paste, Find, Replace, etc.
- Bookmark management
- Line numbering
- Tooltip
- Automatic keyword specification
- Code folds
- Intellisense
- Code templates
- Debugging

10.5.3 Creating a New RCL Program

Create a new RCL program via the following menus:

- In the IndraWorks main menu via **Programs** ► **Add** or
- via the program context menu item **Add...** of the kinematics

Depending on the execution of the respective menu item, the specified program name can be overwritten by default. If the entry is confirmed with **OK**, the RCL Editor is started containing the program body of the new RCL program.

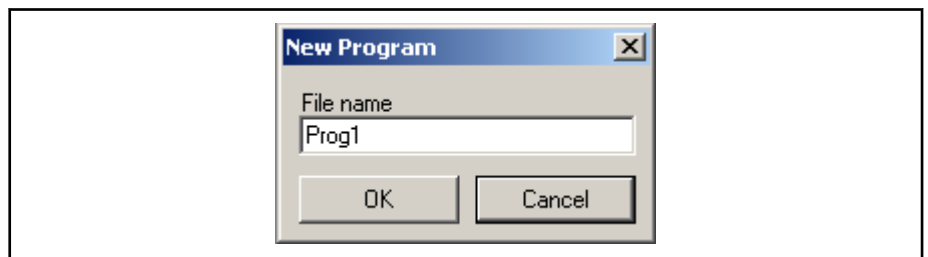


Fig. 10-20: "Add New Program" dialog

Kinematics

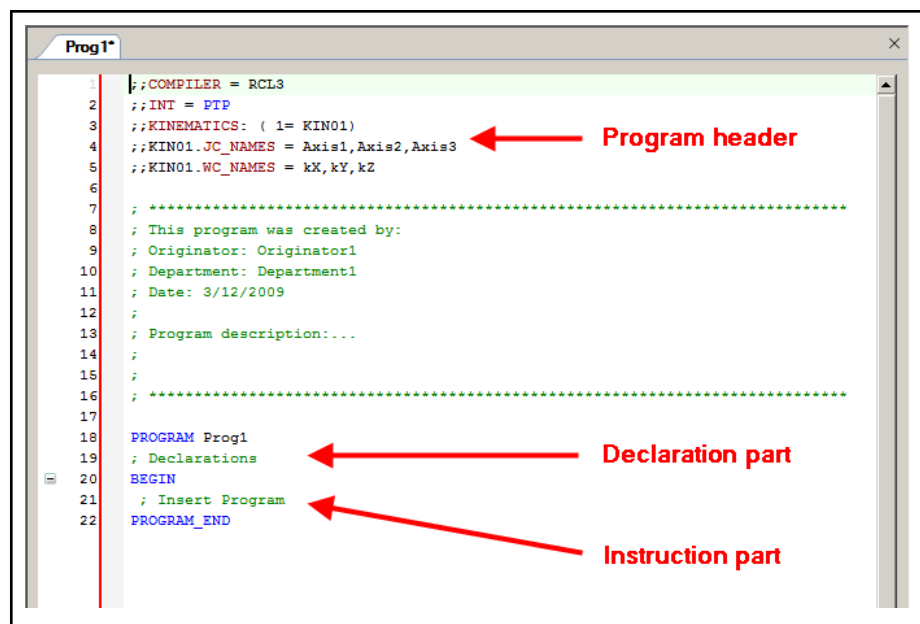


Fig.10-21: General structure of a RoCo program

10.5.4 Working with the Editor

General editor functions

The RoCo Editor integrated useful techniques of common Windows editors:

- Find
- Find files
- Replace
- Block highlighting
- Match case
- Drag&drop

Find and Replace



Fig.10-22: Menu item: "Find and Replace"

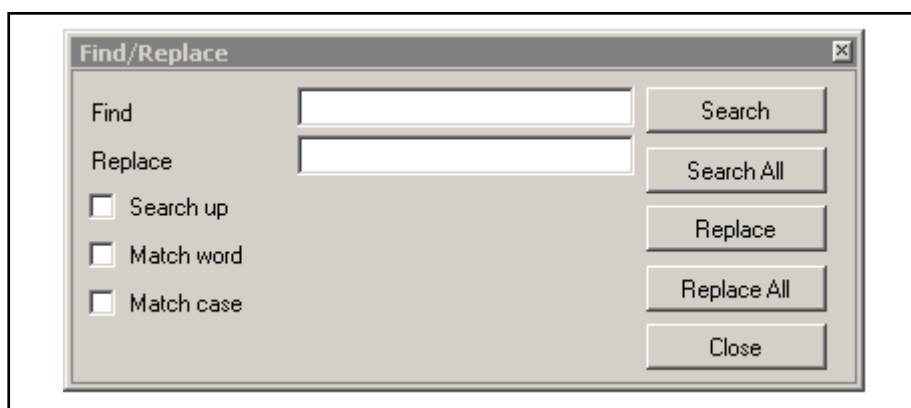


Fig.10-23: Dialog "Find/Replace"

Bookmark

The text editor provides bookmark management. Add bookmarks to lines to facilitate navigation in long programs. The commands described below allow the setting and navigation of bookmarks. They are included by default in the "Bookmarks" submenu in the "Edit" menu. Additionally, they can be found via "Bookmarks" in the toolbar.

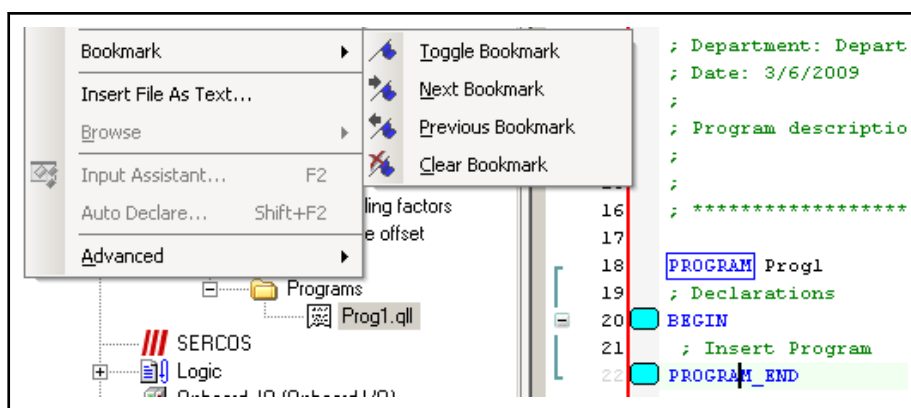


Fig.10-24: Menu item: "Bookmark"

Comments

Comments can be inserted. The comment starts with ";" and ends at the end of the line.

Cut, Copy, Paste

Highlight the area in the editor with the left mouse button or via the Shift cursor shortcut. The commands "Cut", "Copy", "Paste" can then be used in these areas.

Find the commands in the "Edit" menu or via the context menu of the editor.

Supporting the source text input

- **Parameter information**

If the mouse pointer is set on a subroutine, the transfer parameters are displayed.

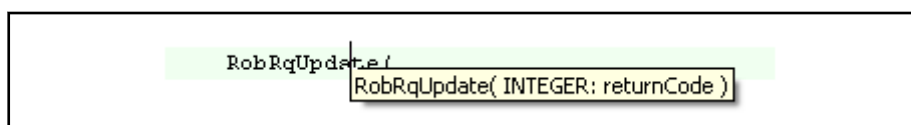


Fig.10-25: Parameter information

Kinematics

If a subroutine name and an open parenthesis are entered, the transfer parameters are also displayed.

- **Code completion**

This function supplements the user input. The input options reduce with the progress and might also with the semantics of the user input. Enter the first character or press <CTRL> + <Space> and a selection list is shown containing the RoCo command block as well as the already declared variables. Apply the object selected in the list with <Return>.

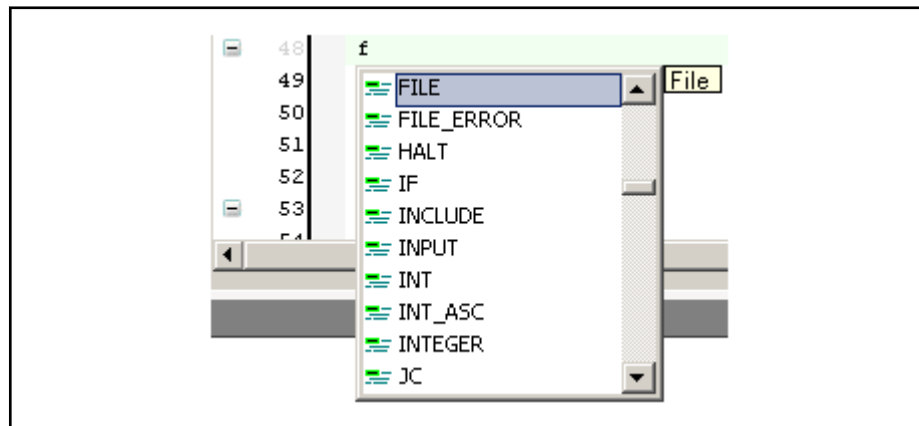


Fig. 10-26: Code completion

- **Element listing**

Enter a link variable and a subsequent point and the elements of the link variables are displayed for selection. Select the desired element using the arrow keys. The selection is applied with <Return> (refer to [RoCo-Editor](#), page 469).

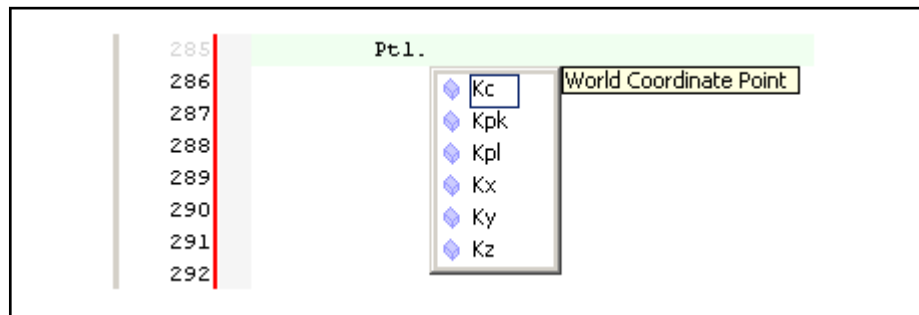


Fig. 10-27: Element listing

- **Type information**

Move the mouse cursor across the variable and the type information is shown as information.

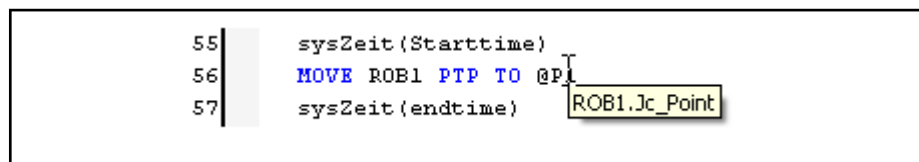


Fig. 10-28: Type information

- **Indenting the cursor automatically**

When writing a source text, the indentation of the previous line is automatically applied to the next line

- **Tabulators**

The length of the tabulator jumps can be set in the "Options" dialog (see [RoCo-Editor, page 469](#))

- **Syntax highlighting**

The keywords in the source text are highlighted. The default specifications can be changed by the user in the "Options" dialog (see [Keywords, page 476](#))

- **Special functions**

Entering special functions for the code completion is supported.

Enter the RoCo command for special functions and it is selected in the drop-down list.

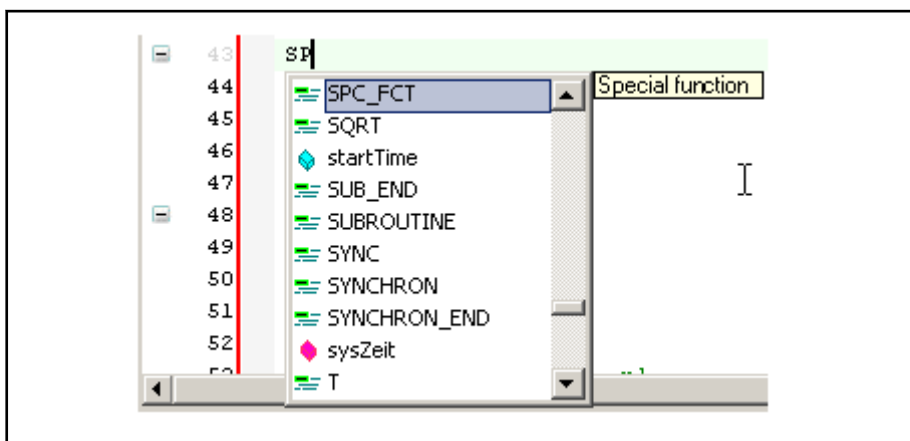


Fig. 10-29: Selecting the "Special Function" command

Apply the object selected in the list with <Return>. A drop-down list displays the special functions available for selection.

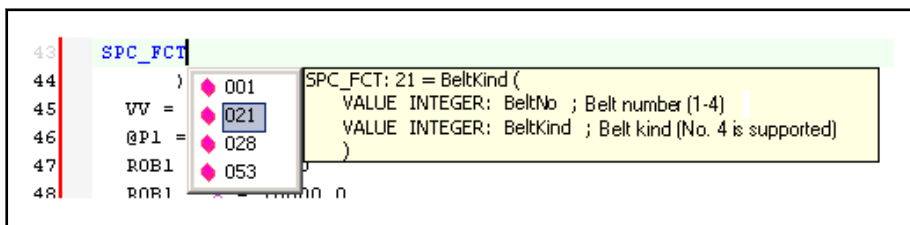


Fig. 10-30: Selecting the special function

<Return> applies the special function selected in the list. The default transfer parameter names can be changed.

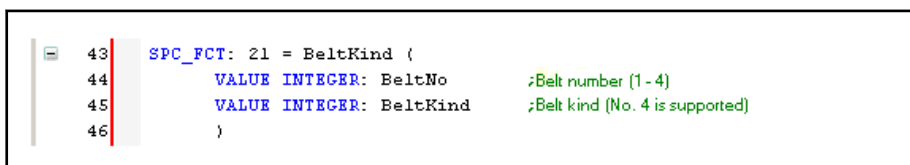


Fig. 10-31: Inserted special function

10.6 Compiler

The compiler creates the program in a machine-readable form from the source text and checks each single line for a correct command syntax. The source text can be compiled in the editor if an RCL program is selected.

Start the compilation as follows:

- via the main menu

Kinematics

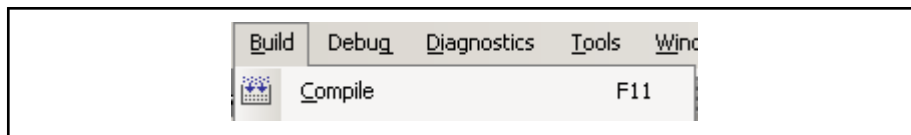


Fig. 10-32: Selecting the compiler

- via the context menu in the editor

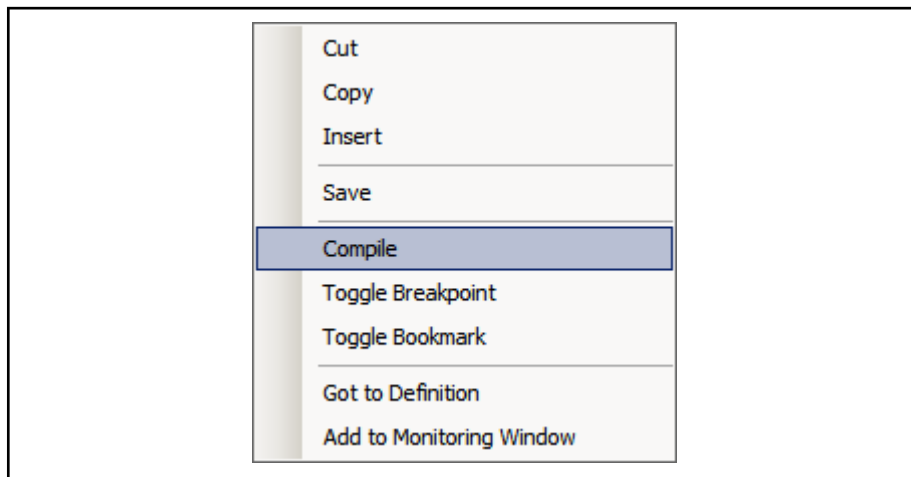


Fig. 10-33: Selecting the compiler

- via <F11>
- a .QLL program can also be compiled by selecting the menu item "Compile" in the context menu of the program.

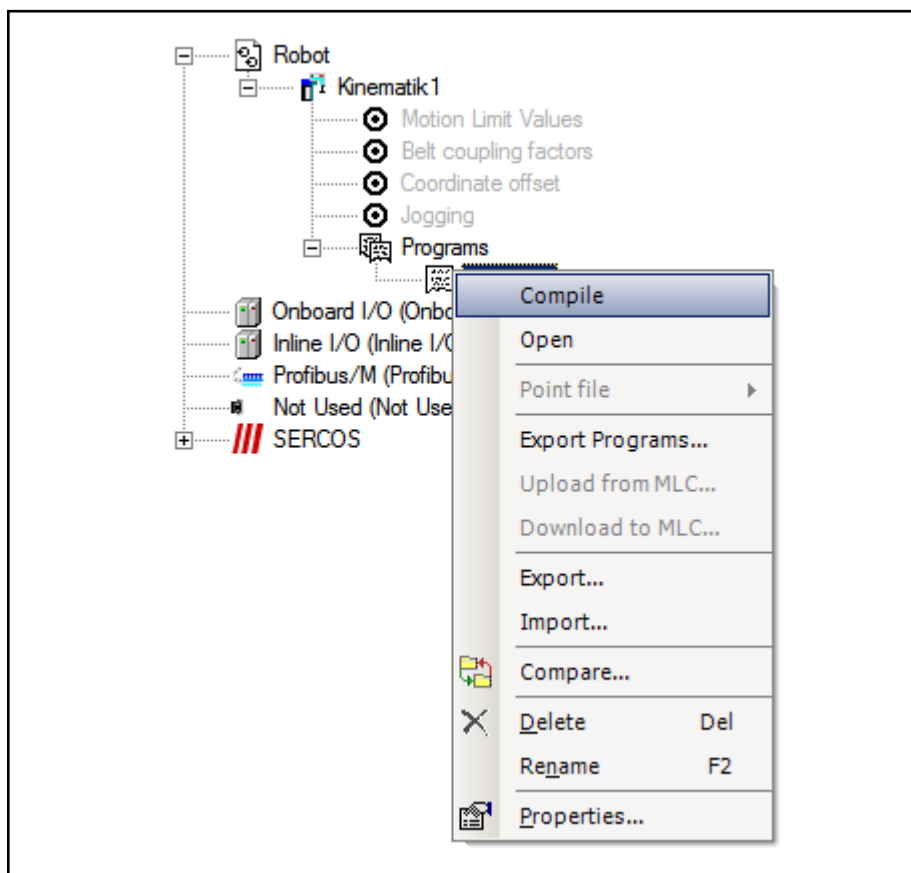


Fig. 10-34: Calling the compiler via the context menu of the program

The selected file is compiled in machine language and saved in the project. An additional immediate saving in the control can be enabled via the option "RoCo Compiler" (see [RoCo Compiler](#), page 467).

The output texts as well as error messages are redirected to the dockable "Output" window.

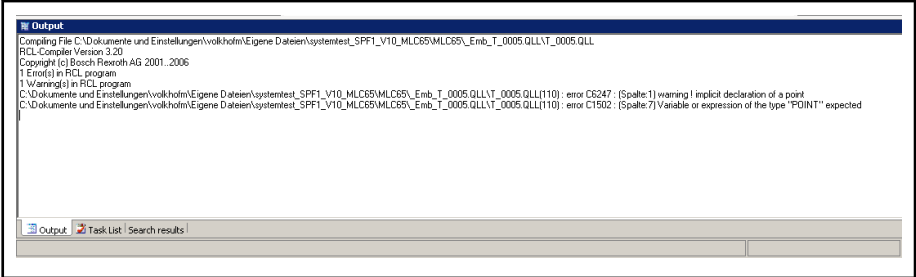


Fig.10-35: Output window of the compiler

If error messages are displayed in the output range, you can jump directly to the position in the program where the error was identified via a double-click on the error message in the **Task list** window.

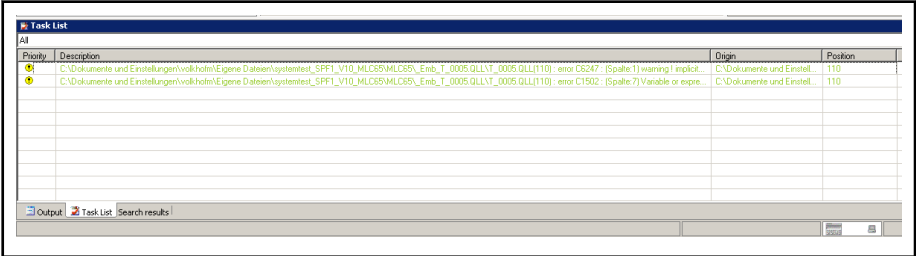


Fig.10-36: Outputting compiler error messages

10.7

Debugger

10.7.1

General Information

The execution of a program in "Debug" mode can be checked for logic programming errors and the content of the variables in the dockable "Monitoring" window can be displayed.

10.7.2

Setting and Deleting Breakpoints

Set breakpoints to interrupt the running program at a specified location.

Set the cursor to the location in the program where the breakpoint should be set.

Kinematics

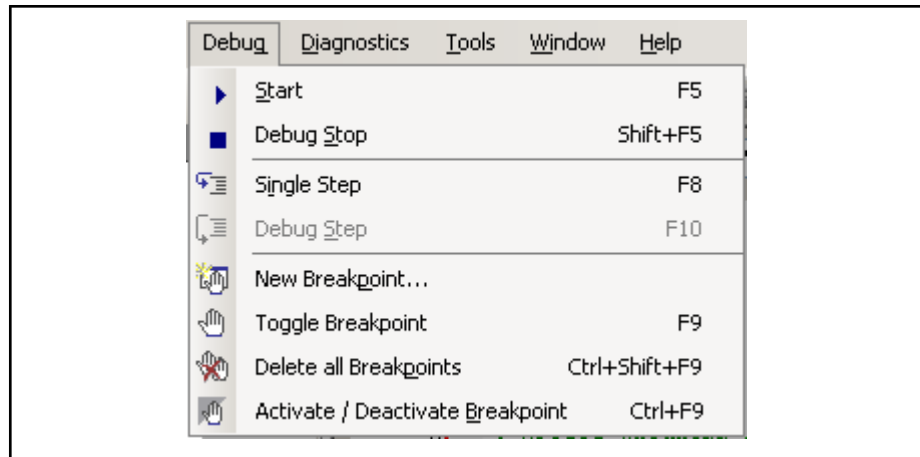


Fig.10-37: "Debug" menu

Click on the **New Breakpoint...** function in the **Debug** menu to set the breakpoint or to delete it again. Set and delete a breakpoint using the **Toggle Breakpoint** function or <F9>. Breakpoints can also be set and deleted by clicking on the indicator bar (grey margin on the left of the source text window).

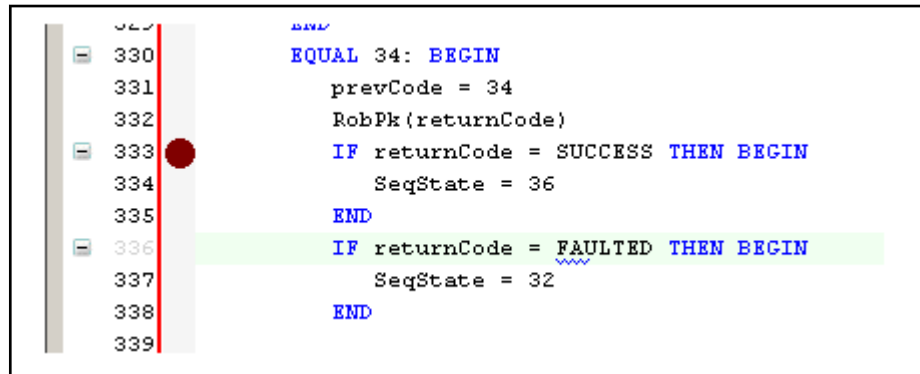


Fig.10-38: Set breakpoint

Delete all breakpoints using the **Delete all Breakpoints** function in the **Debug** menu or the shortcut <CTRL>+<Shift>+<F9>.

10.7.3 Debugging a Program

Execute the program in the debug mode via **Debug ▶ Start** or <F5>.

Add the debugger to an active program via the  icon in the RoCo tool-bar.

The program sequence is interrupted at first breakpoint.

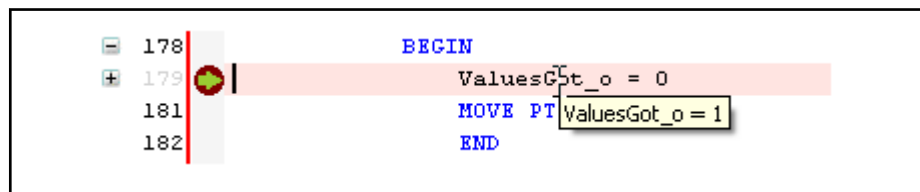


Fig.10-39: Program tracing stops at first breakpoint

Continue the program after a stop at a breakpoint with <F5>. Test the program in single steps with <F8>. Close the debugger with <Shift> + <F9>.

Set the cursor on a variable to show the current variable value in the tooltip.

10.7.4 Monitoring Variables

Variables can be transferred to the monitoring window in the **debug mode**. The variables can also be entered directly into the monitoring window.

The  icon in the RoCo toolbar opens the monitoring window.

10.8 Settable Options for RoCo

10.8.1 General Information

Via the IndraWorks menu item **Tools ► Options** under "XLC/MLC", the default settings for the following can be changed:

- RoCo Compiler (see [RoCo Compiler, page 467](#))
- RoCo online settings (see [RoCo Online Settings, page 469](#))
- RoCo Editor (see [RoCo-Editor, page 469](#))

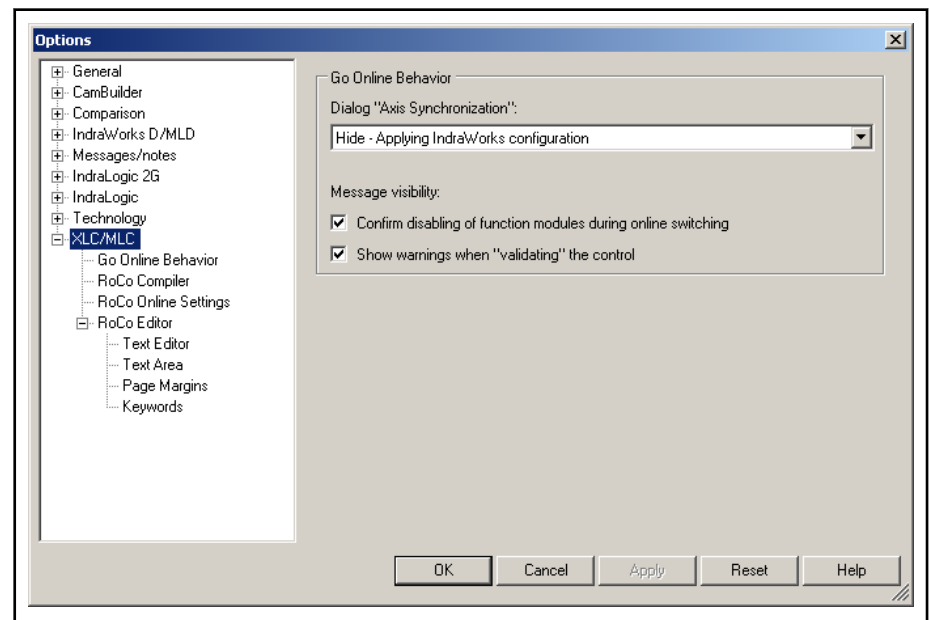


Fig. 10-40: RoCo "Options" dialogs

10.8.2 RoCo Compiler

This "Options" dialog provides settings to add compiler instructions to a new RCL program and to enable the option "Copy to control after compiling (without confirmation)".

Kinematics

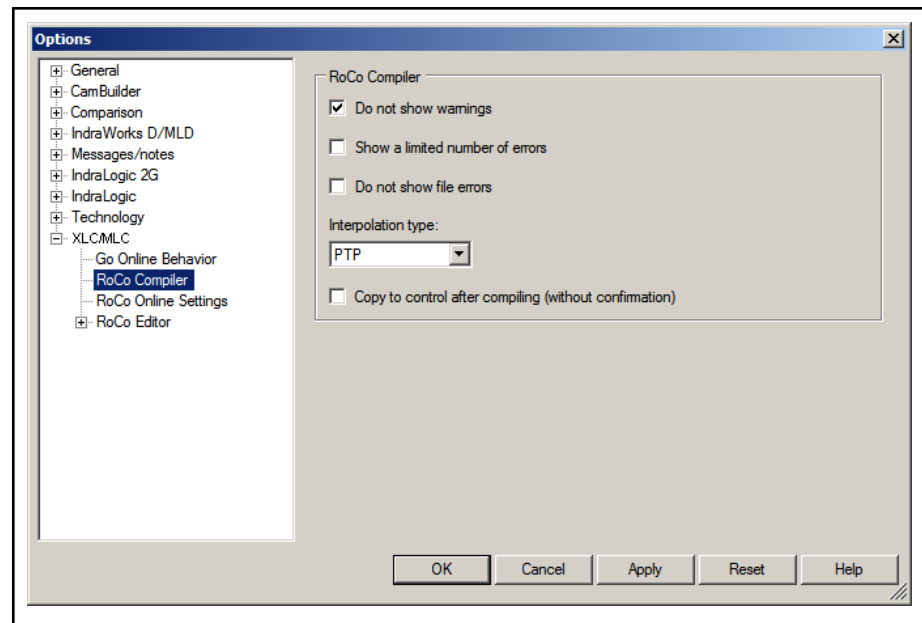


Fig. 10-41: RoCo compiler settings

Compiler instructions

Settings:

1. Adding "WARNING" compiler instruction

If the "Do not output warnings" checkbox is selected, the "WARNING" instruction is added when a new RCL program is created (see [Added compiler instructions, page 469](#)).

With this setting, possible warnings are not displayed after a faulty compilation.

2. Adding "Error=100" compiler instruction

If the "Do not output limited number of errors" checkbox is selected, the "Error = 100" instruction is added when a new RCL program is created. The maximum number of errors to be output can be determined by changing the value "100" in the RCL program.

This instruction can limit the number of errors to be output to a specified number after a faulty compilation.

3. Adding "FILE_ERROR" compiler instruction

If the "Do not output file error" checkbox is selected, the "File error" instruction is added when a new RCL program is created (see [Added compiler instructions, page 469](#)).

This setting impedes the output of file errors at program runtime. That means that a program abort can be avoided and individual error messages can be implemented in the program.

4. Selecting the interpolation type

If a new RCL program is created, the selected interpolation type is added as compiler instruction (see [Added compiler instructions, page 469](#)).

This setting remains active until another interpolation type is selected by the ";;INT=" instruction.

5. Copy to control after compiling (without confirmation)

If this checkbox is selected, the RCL program and the files generated by the control are loaded to the control after a correct compilation. If this option is not selected, the program is only compiled.

```

1  ;;COMPILER = RCL3
2  ;;WARNING- ← Instruction "don't show warnings"
3  ;;FILE_ERROR- ← Instruction "don't show file errors"
4  ;;ERROR = 100 ← Instruction "limit displayed errors to 100"
5  ;;INT = PTP ← Interpolation type selection "PTP"
6  ;;KINEMATICS: ( 1= Kinematics1)
7  ;;Kinematics1.JC_NAMES = vaxis1, vaxis2
8  ;;Kinematics1.WC_NAMES = Ka,Kb

```

Fig.10-42: Added compiler instructions

10.8.3 RoCo Online Settings

The time-out values for the RoCo during file transfer to the control can be changed using this dialog.

The currently set value is displayed in "seconds".

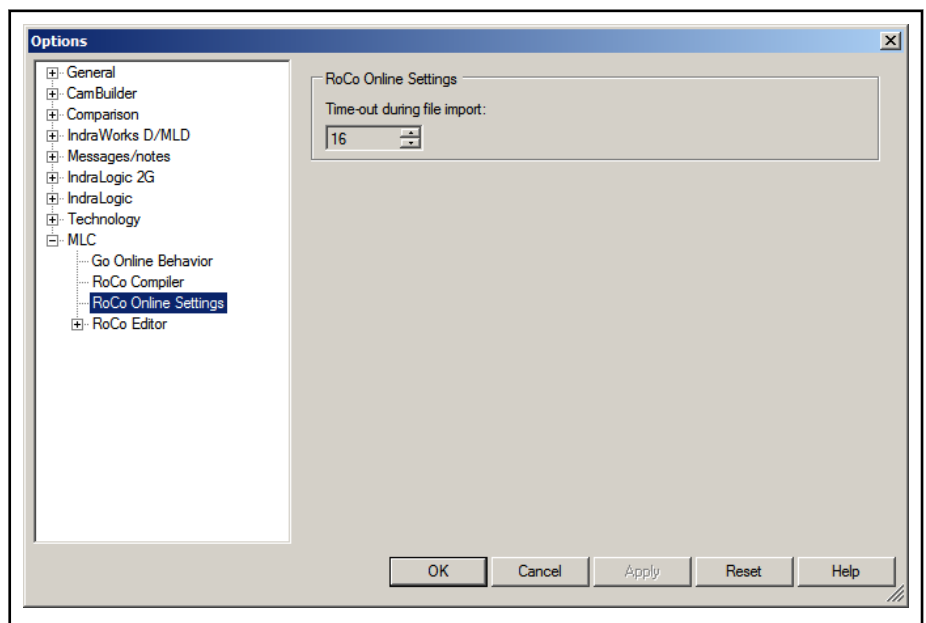


Fig.10-43: RoCo online settings

10.8.4 RoCo-Editor

Text Editor

In this "Options dialog", text editor properties such as

- Undo
- Auto completion
- Line break
- Tab size
- Indent width

Kinematics

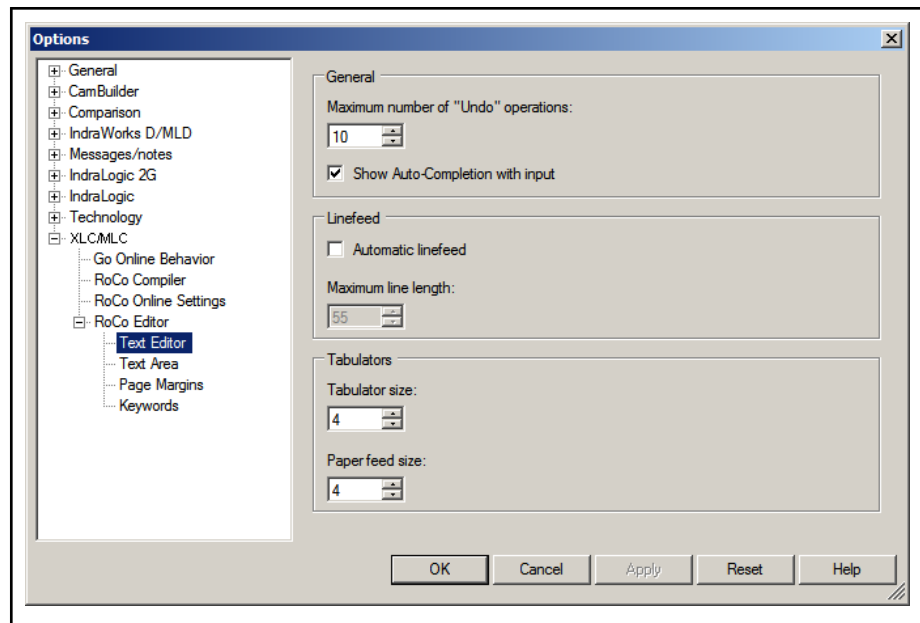


Fig. 10-44: Text editor settings

Setting options in the text editor "Options" dialog:

1. **Max. no. of undo**

The maximum number of undo steps can be set in the RoCo-Editor.

2. **Show auto-complete upon entry**

If this checkbox is selected, a window opens while writing a word. In this window, the word with the highest match is highlighted. Words in this list can be selected. The highlighted word is applied with "Return".

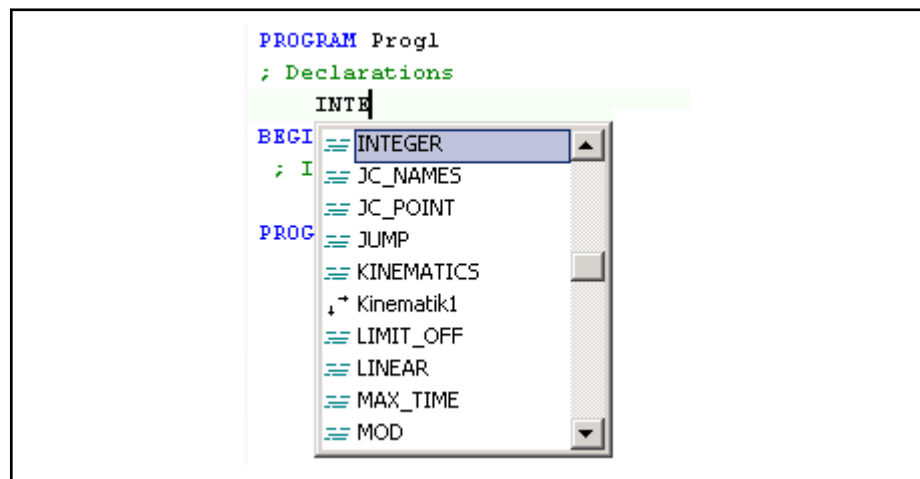


Fig. 10-45: Auto completion

3. **Automatic linefeed**

If this checkbox is selected, a linefeed is executed automatically in the RoCo-Editor if the number of characters in the current line exceeds the value set in "Maximum line length".

4. **Tabulator size**

The size of the tabulator step width can be set in the "Tabulator size" selection box.

Example: In [fig. 10-45 "Auto completion" on page 470](#), the tabulator size is set to "4". That means that a tabulator step is the size of four blanks (see line "INTE").

5. Indent width

If "Return" is pressed, there is an automatic linefeed in the editor. This automatically generated linefeed is called "indent" or "indentation". The number of characters for the indentation can be set in the "Indent width" selection box.

Text Area

In this "Options" dialog, colors for highlighting as well as font type and font style in the RoCo-Editor can be set.

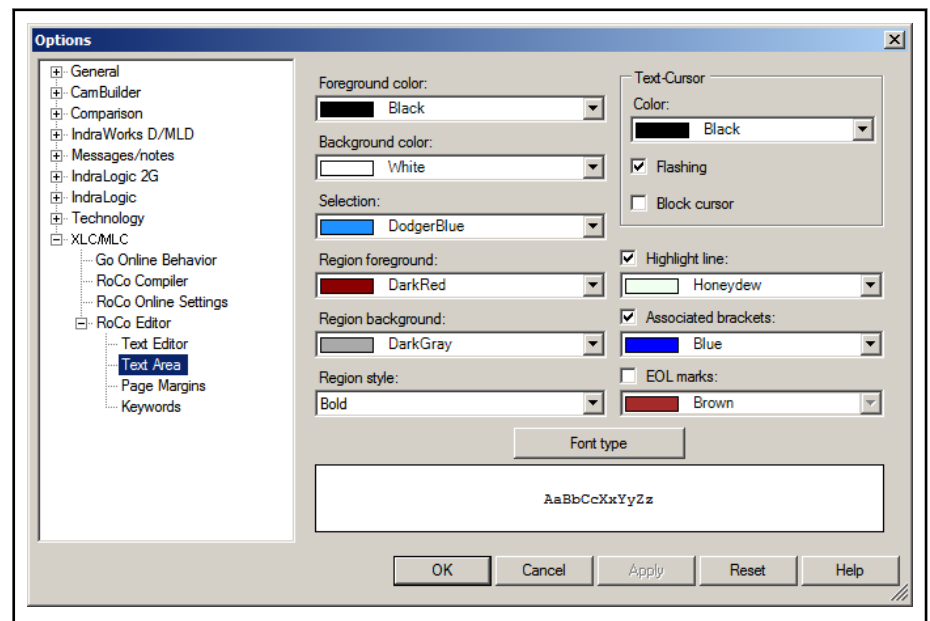


Fig. 10-46: Text area settings

Functions of the "Text area" "Options" dialog:

1. Foreground color, background color, font style of the editor

The foreground colors (text color) and the background colors of the RoCo-Editor can be selected with the combo boxes "Foreground color" and "Background color".



The colors for keywords, comments, etc. cannot be changed in this dialog. To change the above mentioned, open the "Keywords" (see [Keywords, page 476](#)).

Press the **Font type** button and the Windows dialog to select font type, style and size is displayed.

2. Selection, Region foreground, Region background, Region style

The color can be set for a highlighted area via "Selection" (see [fig. 10-48 "Display of the selected text area" on page 472](#)). The minimized representation, e.g. between the keywords BEGIN and END or SUB-ROUTINE and SUB_END can be changed in color and in font style (bold, italic, etc.) via "Region foreground", "Region background" and "Region style" (see [fig. 10-49 "Representation in color and text of the minimized area between BEGIN and END" on page 472](#)).

Example:

Kinematics

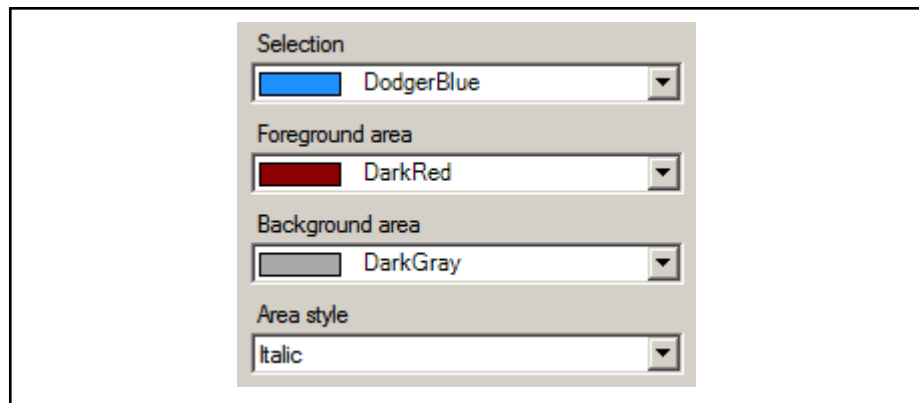


Fig. 10-47: Settings for selection, foreground, background and style

The display of the areas in the following figures is carried out according to the settings in fig. 10-47 "Settings for selection, foreground, background and style" on page 472.

```

PROGRAM Prog1
; Declarations
DEF POINT: P0, P1, P2, P3, P4, P5, P6

BEGIN
; Program
KIN01.V_PTP = 0.4           ; Velocity for point to point movements
KIN01.A = 1000             ; Acceleration
KIN01.V = 1200             ; Velocity

MOVE TO P0                 ; Move to P0
KIN01.V = 400              ; Velocity
WAIT 3                     ; Wait 3 seconds
REPEAT 4 TIMES             ; Repeat 4 times
    MOVE TO P0
    MOVE TO P1
    MOVE TO P2
    MOVE TO P3
    MOVE VIA P4, P5, P6 TO P0 ; Move via P4, P5, P6 to P0
REPEAT_END

PROGRAM_END

```

Fig. 10-48: Display of the selected text area

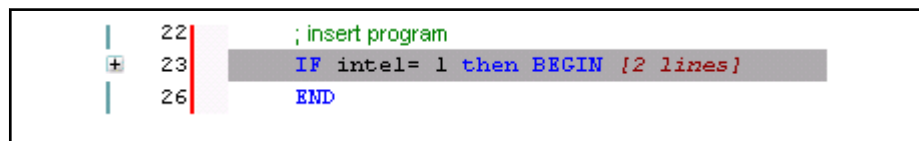


Fig. 10-49: Representation in color and text of the minimized area between BEGIN and END



If an area is minimized, this is indicated by a "-" symbol at the page margin. The area is minimized when clicking on the "-" symbol. This is indicated by a "+" symbol at the page margin. Minimization can be undone by clicking on the "+" symbol.

3. Cursor settings

The display of the text cursor in the RoCo-Editor can be changed. The following settings can be made:

- Cursor color

- Cursor flashing
- Block cursor

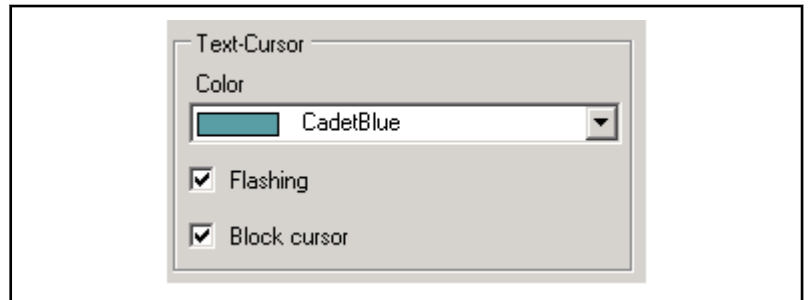


Fig. 10-50: Setting option for the text cursor

The display of the cursor in the following figure is carried out according to the settings in [fig. 10-50 "Setting option for the text cursor" on page 473](#).

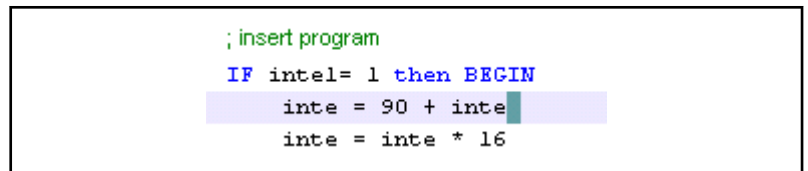


Fig. 10-51: Display of the text cursor in the RoCo-Editor

4. Highlight line

If the checkbox "Highlight line" is selected, the line in the cursor is highlighted with the selected color.

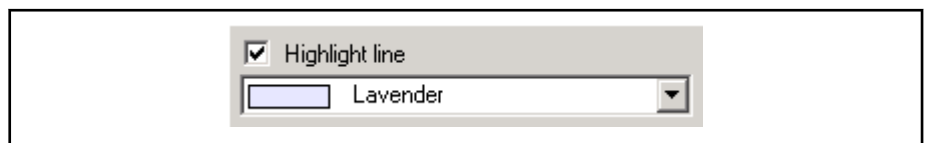


Fig. 10-52: Setting options for the current line

The display of the current line in the following figure is carried out according to the settings in [fig. 10-52 "Setting options for the current line" on page 473](#).

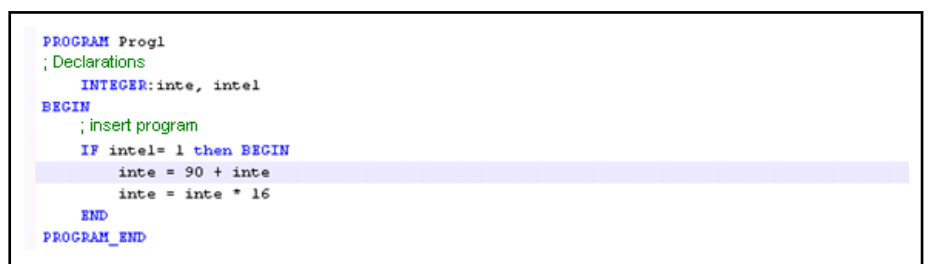


Fig. 10-53: Colored highlighting of the current line

5. Associated brackets

If the cursor is located directly behind an END, REPEAT_END etc., the respective beginning of the instruction block (BEGIN, REPEAT, etc.) is highlighted with a frame if the checkbox is selected. Additionally, the color of the frame can be selected.

Kinematics

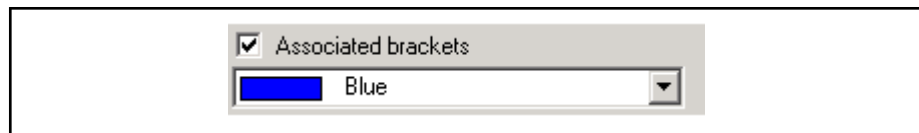


Fig.10-54: Setting options for a program block

Highlighting the block beginning in the following figure is carried out according to the settings in [fig. 10-54 "Setting options for a program block" on page 474](#).

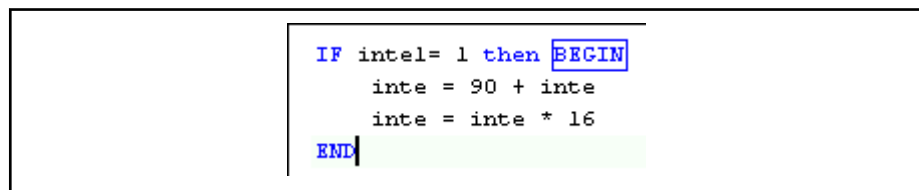


Fig.10-55: colored highlighted of an instruction block

6. EOL marks

The display of the EOL (End Of Line) can be carried out here. If the option is enabled, it highlights a point at the end of the line. Additionally, the color of the point can be selected.

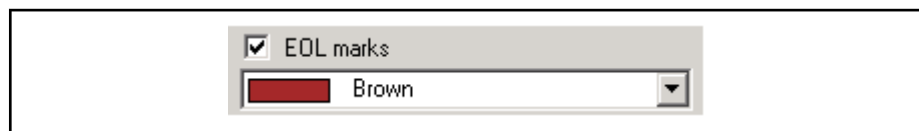


Fig.10-56: Setting options for the EOL display

The display of the EOL in the following figure is carried out according to the settings in [fig. 10-56 "Setting options for the EOL display" on page 474](#).

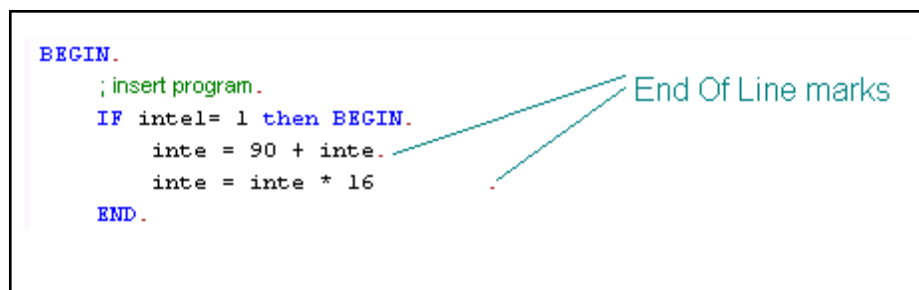


Fig.10-57: EOL marks in the text

Page Margins

In this "Options" dialog, the page margins in the RoCo-Editor can be set.

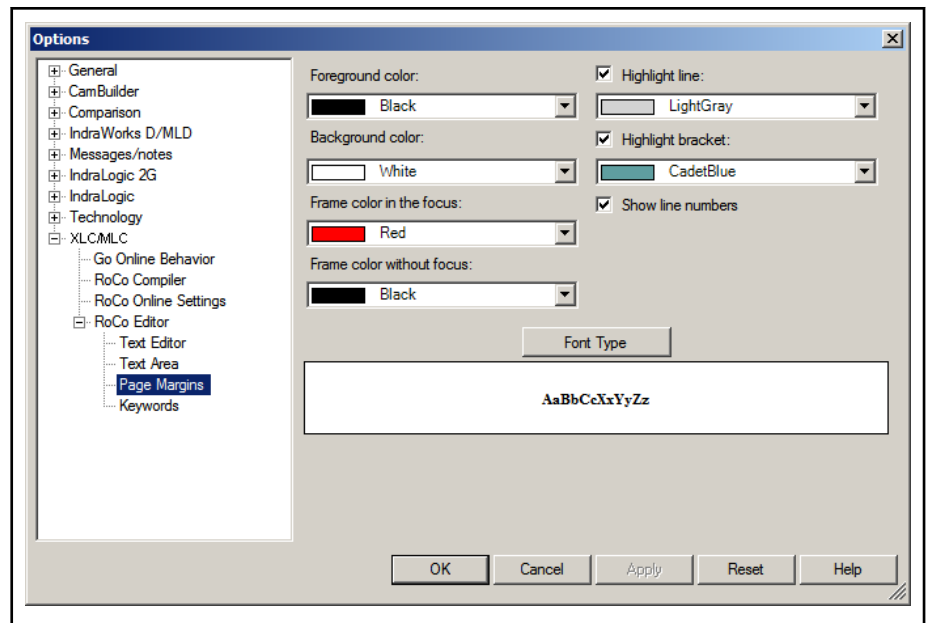


Fig. 10-58: Page margin settings

Functions of the "Page margins" "Options" dialog:

1. **Foreground color, background color, font style of the page margin**

The foreground colors (text color) and the background colors of the page margin (on the left) can be set using the combo boxes "Foreground color" and "Background color" (see [fig. 10-59 "Setting option for the page margin in the RoCo-Editor" on page 476](#)).

2. **Frame color "in the focus", Frame color "without focus"**

The vertical frame line (see [fig. 10-59 "Setting option for the page margin in the RoCo-Editor" on page 476](#)) on the right next to the line numbers indicates whether the window of the RoCo-Editor has the focus. This option can specify the line color for both states.

3. **Highlight line**

The color of the line number to be highlighted on which the current cursor is located can be set in this option (see [fig. 10-59 "Setting option for the page margin in the RoCo-Editor" on page 476](#)).

4. **Highlight brackets**

If this option is selected, the area in the brackets ("(...)") is indicated with a square bracket ("[]") while editing at the page margin (see [fig. 10-59 "Setting option for the page margin in the RoCo-Editor" on page 476](#)). Additionally, the color of the indicated bracket can be selected.

5. **Show line numbers**

The display of the line numbers at the left page margin can be switched on/off via this option (see [fig. 10-59 "Setting option for the page margin in the RoCo-Editor" on page 476](#)).

Kinematics

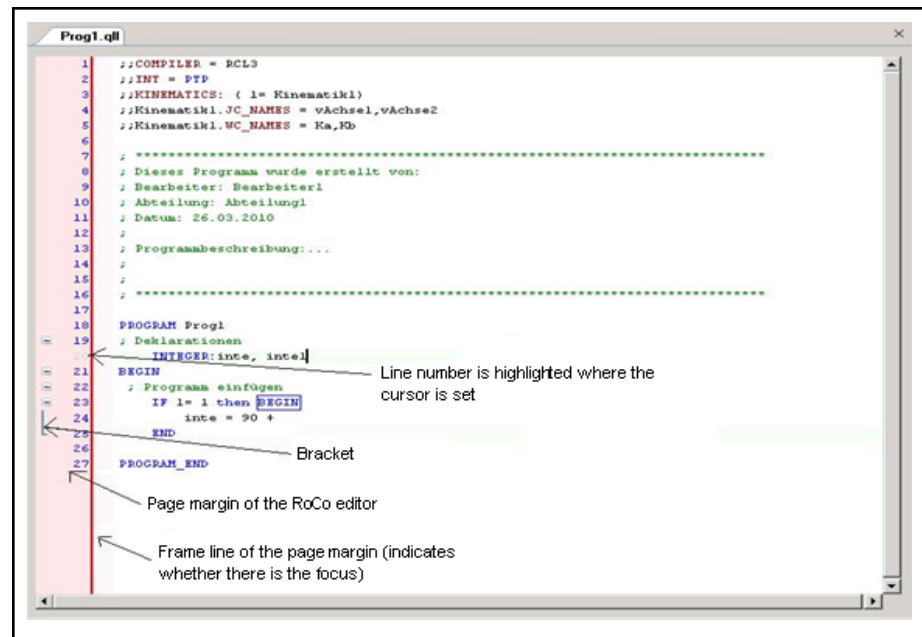


Fig. 10-59: Setting option for the page margin in the RoCo-Editor

Keywords

The colored display of the keywords can be changed in this "Options" dialog. The settings options include the colored representation (foreground color) the writing cut of the text (bold, italics, underlined, strikethrough) as well as the background color.

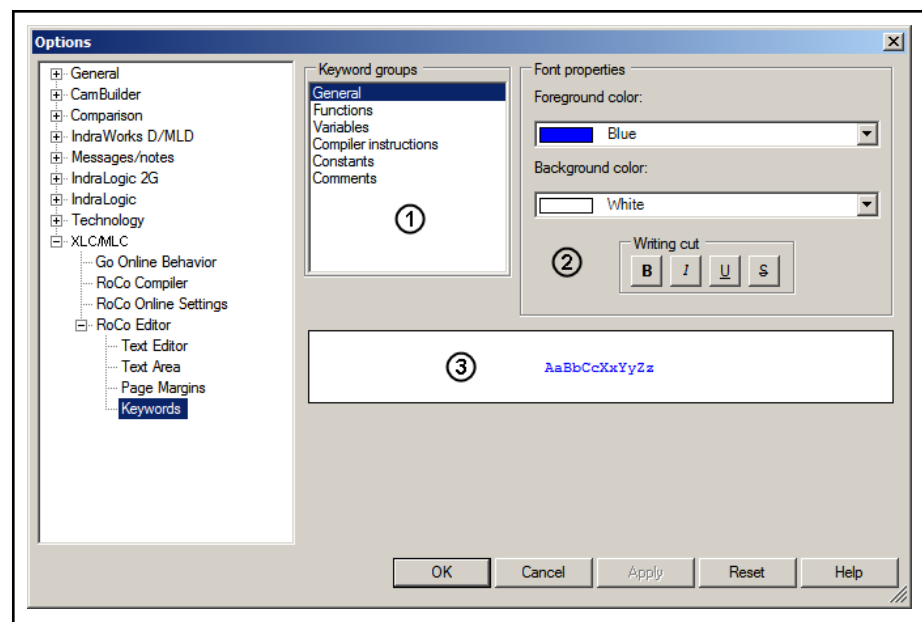


Fig. 10-60: Keyword settings

The dialog consists of three sections:

- Keyword groups ①
- Font properties ②
- Example display ③

Procedure for setting changes:

1. Marking the keyword group in the list ①

2. Changing the properties in ②
3. The example display indicates the representation with changed properties ③
4. Click on **Apply Comment** or **OK..**

10.9 Creating a RoCo Project



The application of the RoCo functionality (RobotControl) requires the following hardware for a permanent data backup (saving the RoCo program):

- an IndraControl L45/L65 with 8 MB S-RAM on board (battery-buffered)
- or - if the IndraControl L control has no onboard S-RAM -
- an **S-RAM module (CFL1.1-Y1)** function module (8 MB).

This can be created in the "Configuration Dialog" of the control (or via "drag" from the **Device Library ▶ FM** and "drop" it to the control in the Project Explorer.

For details, see /24/, [Required and Supplementing Documents, page 14](#).

With an easy example, the basic procedure to create a RobotControl project with the IndraMotion MLC is described.

From creating a project up to generating and loading the RoCo source code as well as creating the PLC program, some keywords are given regarding the minimum configuration required in IndraWorks.

For the "RoCoTestV12" example project, it is assumed that only virtual drives are available.

However, it is now described briefly what to do if real drives are available.

Therefore, the virtual drives in the project are substituted by real drives. It is to be considered that

1. the real drives are always provided with the same axis name Axis1, Axis2 and Axis3 and
2. the axis numbers 1, 2 and 3 have been assigned in the same order

Thus, it is guaranteed that the PLC program runs without any adaptations.



The user can thus load the example project containing

- Parameter export file
- RCL program
- the PLC program and
- the visualization data

If the user wants to program the entire project, the attachment with the detailed data can be used.

Procedure:

1. Creating a project, see [Creating a Project, page 51](#).
2. To create and configure an IndraMotion MLC control, see [Creating a Control, page 53](#).

Kinematics

3. Creating and configuring three virtual axes or real axes with the names Axis1... Axis3, refer to [Creating a Virtual Axis \(Virtual Master Axis\)](#), page 243, or [Creating a Real Axis](#), page 234.
4. Adding and configuring a kinematics.

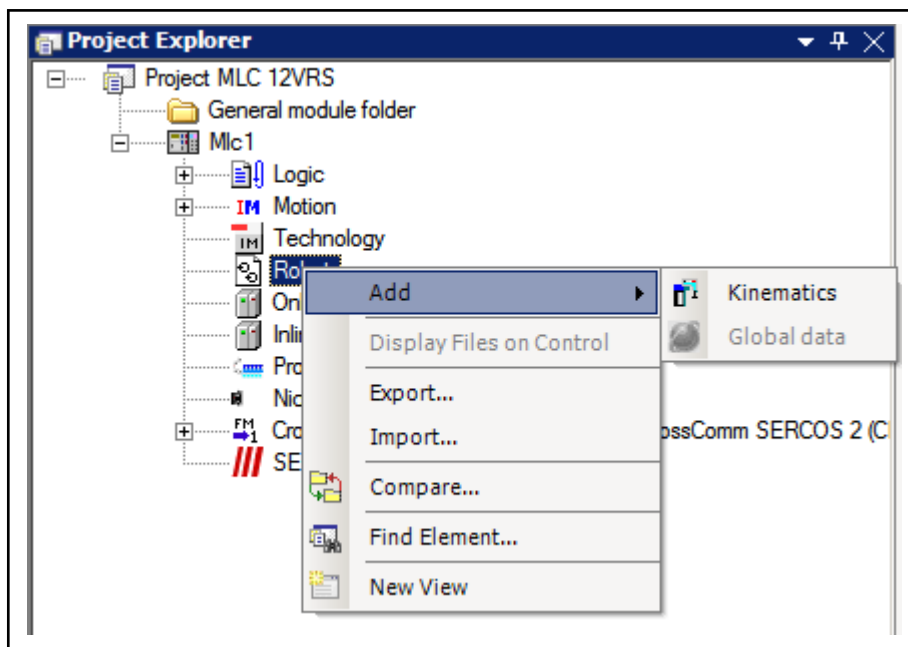


Fig. 10-61: Creating a kinematics

Assigning kinematic name "KIN01".

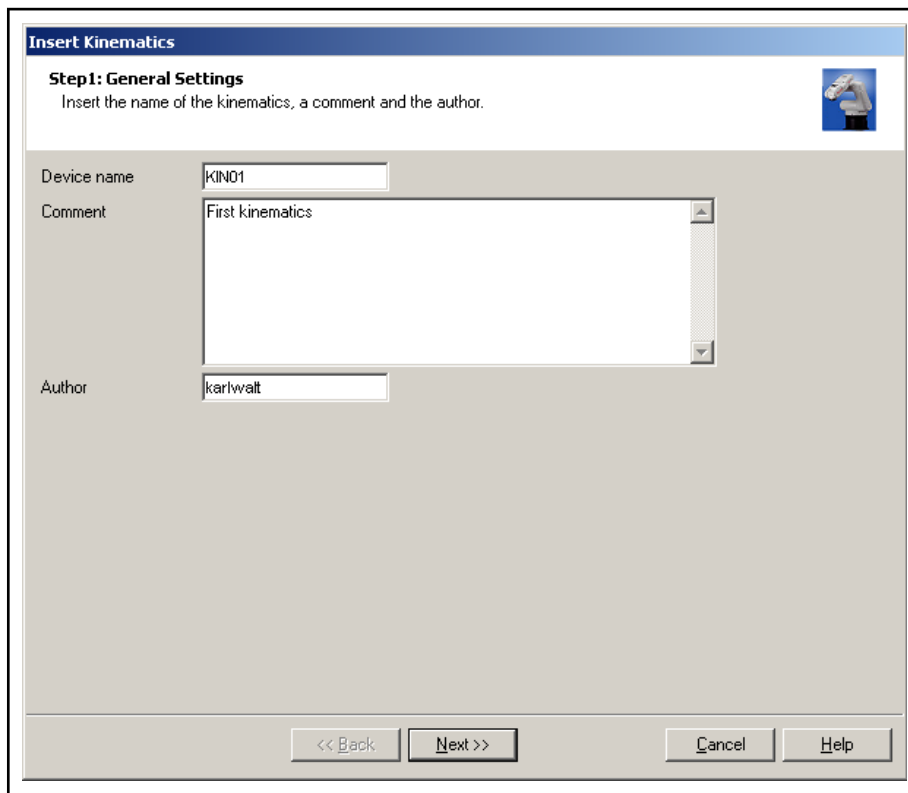


Fig. 10-62: Assigning kinematic name

Selecting kinematic type

In this example, the **Cartesian** kinematic type is used.

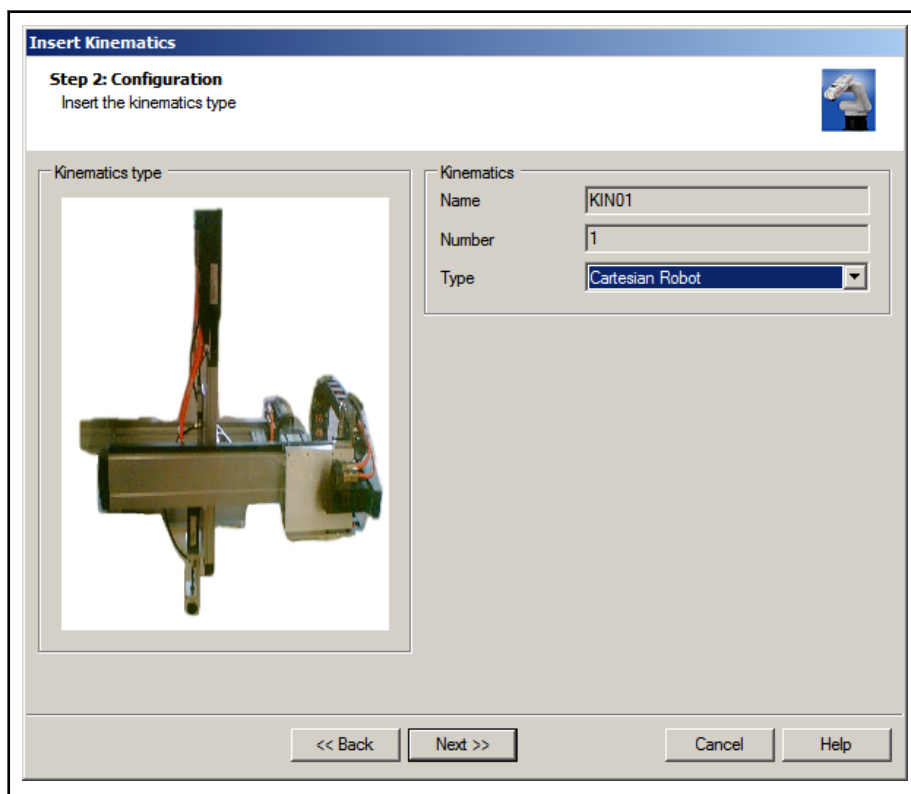


Fig. 10-63: Selecting kinematic type

Assigning axes to the kinematics

All (real or virtual) axes physically belonging to the kinematics have to be assigned here.

Kinematics

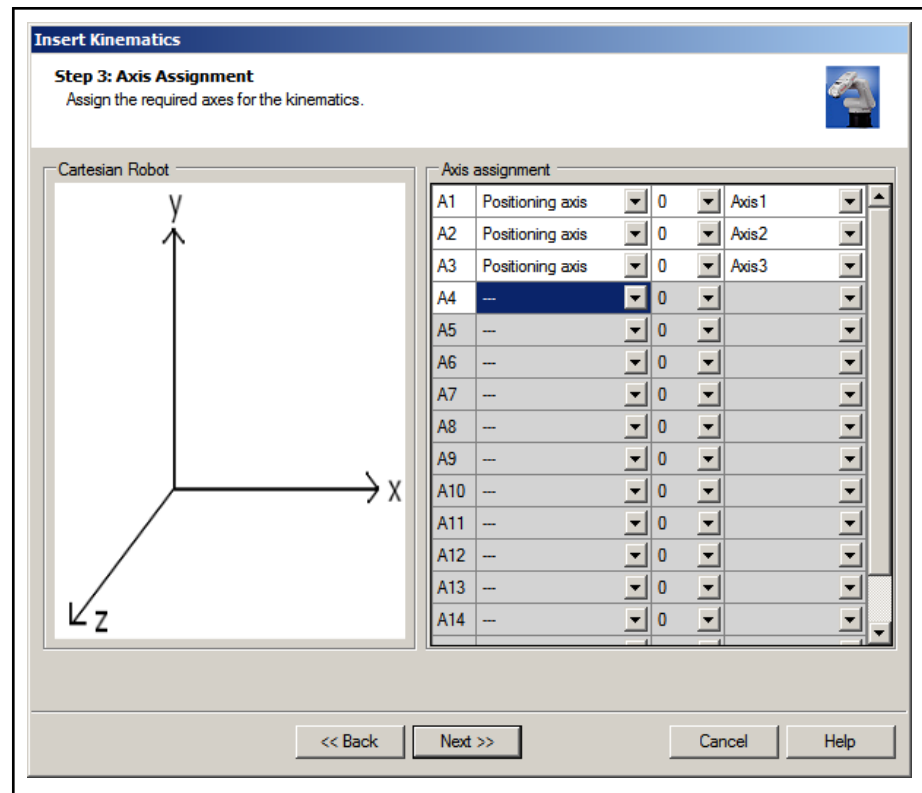


Fig. 10-64: Assigning axes belonging to the kinematics

Defining axis lengths and axis couplings

The typical axis couplings and axis lengths for the robots are entered here. This is not required for the discussed example.

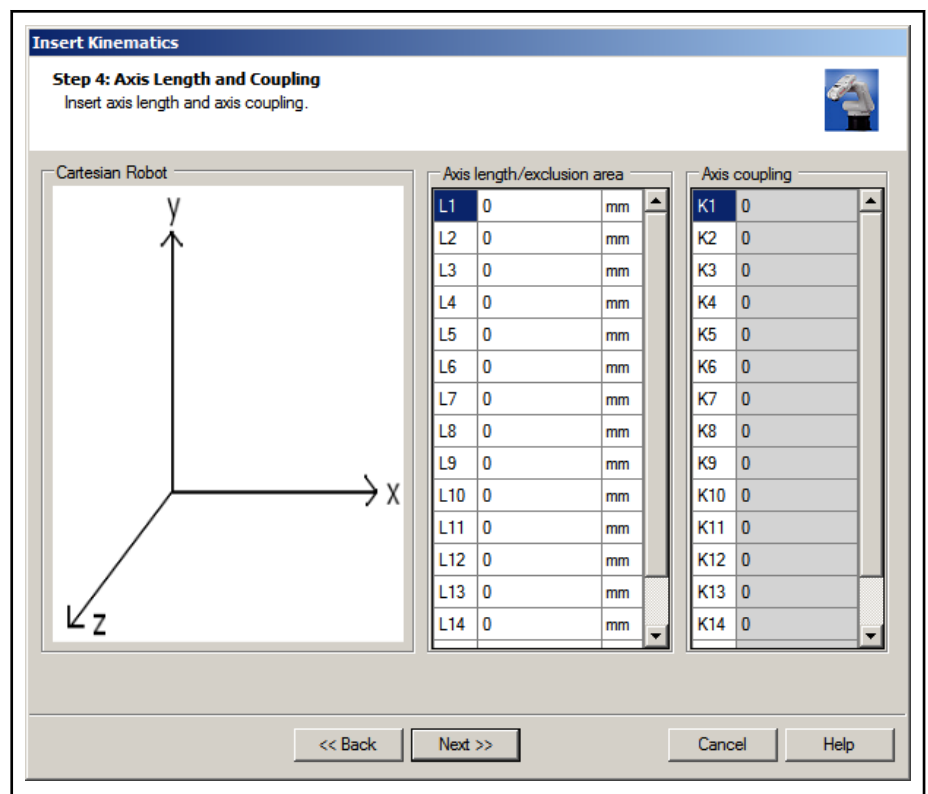


Fig. 10-65: Defining axis lengths and axis couplings

Defining belt coupling factors

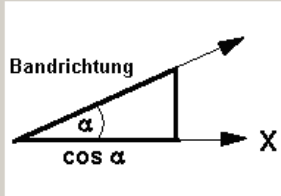
Enter coupling factors for available belts. This is not required for the discussed example.

Kinematics

Insert Kinematics

Step 5: Belt - Coupling Factors
Enter the belt axis coupling factors

Bandrichtung



Band1

Wert 1 (cos α)	0
Wert 2 (cos β)	0
Wert 3 (cos γ)	0

Band2

Wert 1 (cos α)	0
Wert 2 (cos β)	0
Wert 3 (cos γ)	0

Band3

Wert 1 (cos α)	0
Wert 2 (cos β)	0
Wert 3 (cos γ)	0

Band4

Wert 1 (cos α)	0
Wert 2 (cos β)	0
Wert 3 (cos γ)	0

<< Back Next >> Cancel Help

Fig. 10-66: Belt coupling factors

Coordinate offset

Offsets and zero point rotations of the world coordinate system can be entered here. This is not required for the discussed example.

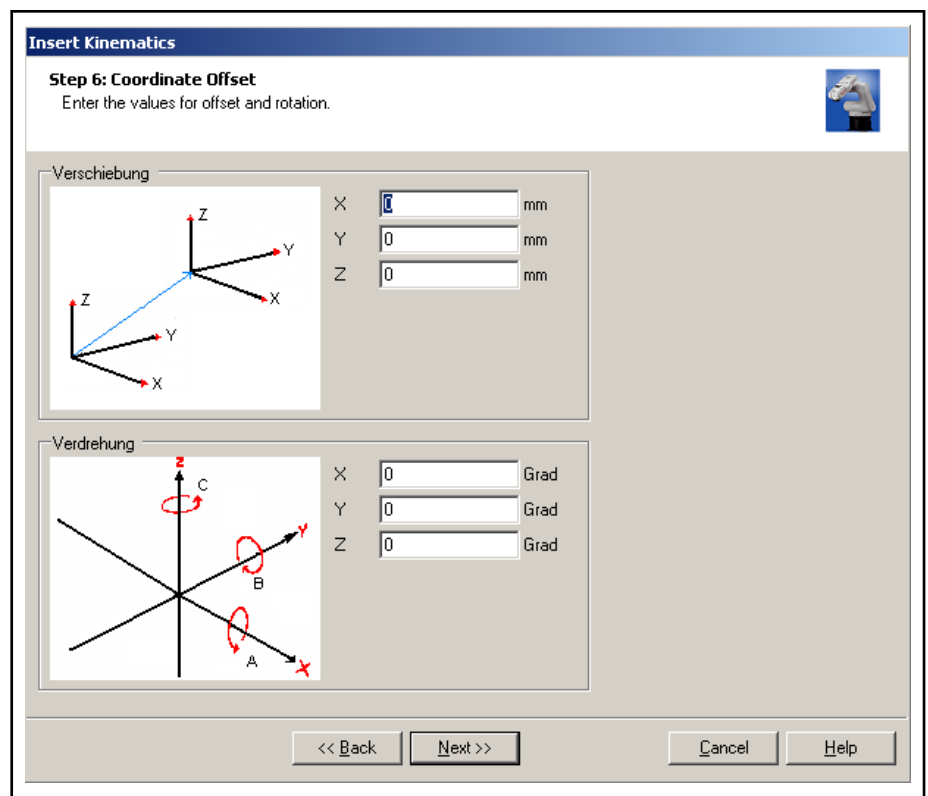


Fig.10-67: Coordinate offset

Defining axis coordinate names

The defined coordinate names kX, kY and kZ are used in the subsequent RoCo programming.

Kinematics

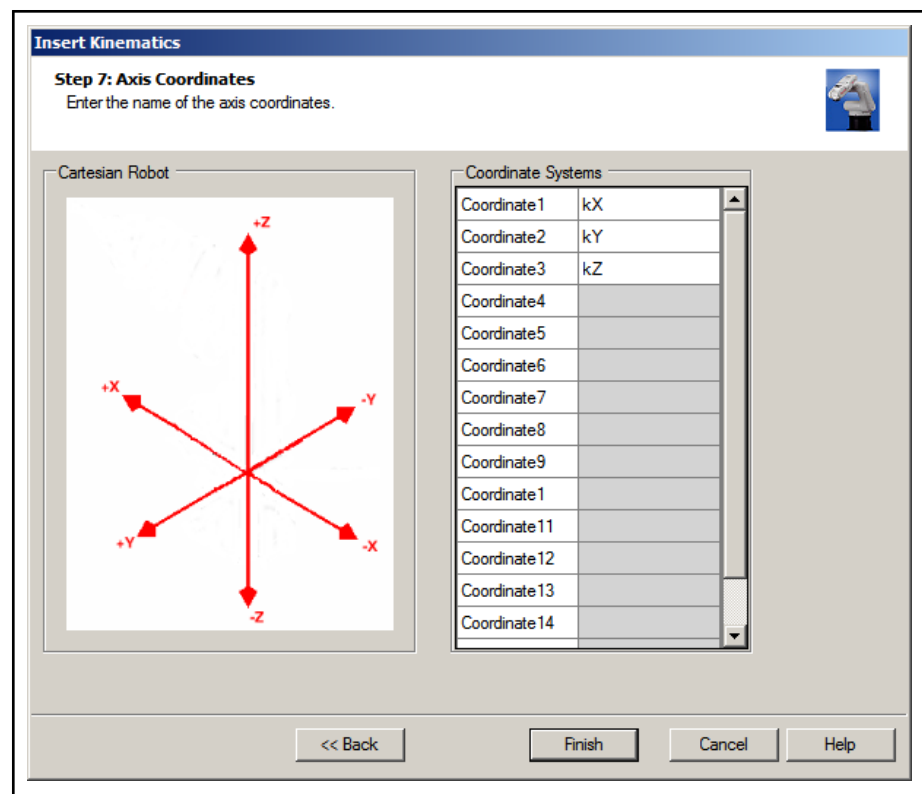


Fig. 10-68: Defining axis coordinate names

5. Going online.

The control is switched to P2 and the settings made up to now are applied to the control.

6. Importing parameters.



Parameter sets (RoCoVirtuell.par/RoCoReal.par) that can be imported are available for both the scenarios.

For the example project, the C-, A- and K-parameters can be imported.

After a successful parameter import, it should be possible to switch the control to **BB** (P4).

7. Creating a RoCo motion program for the kinematics.

Click on the "+" symbol on the left next to the "Robot" folder and on the "+" symbol on the left next to the "KIN01" subfolder. To create a motion program, highlight the "Programs" subitem and right click.

Select "Add..." and apply the name "Prog1" for the new program.

It can also be imported from the **RoCo-TestV12 ▶ mlc208 ▶ _Emb_Prog1.qll ▶ Prog1.qll** project with

- Prog1.QLL
- Prog1.PKT
- Prog1.IRD
- Prog1.SYM

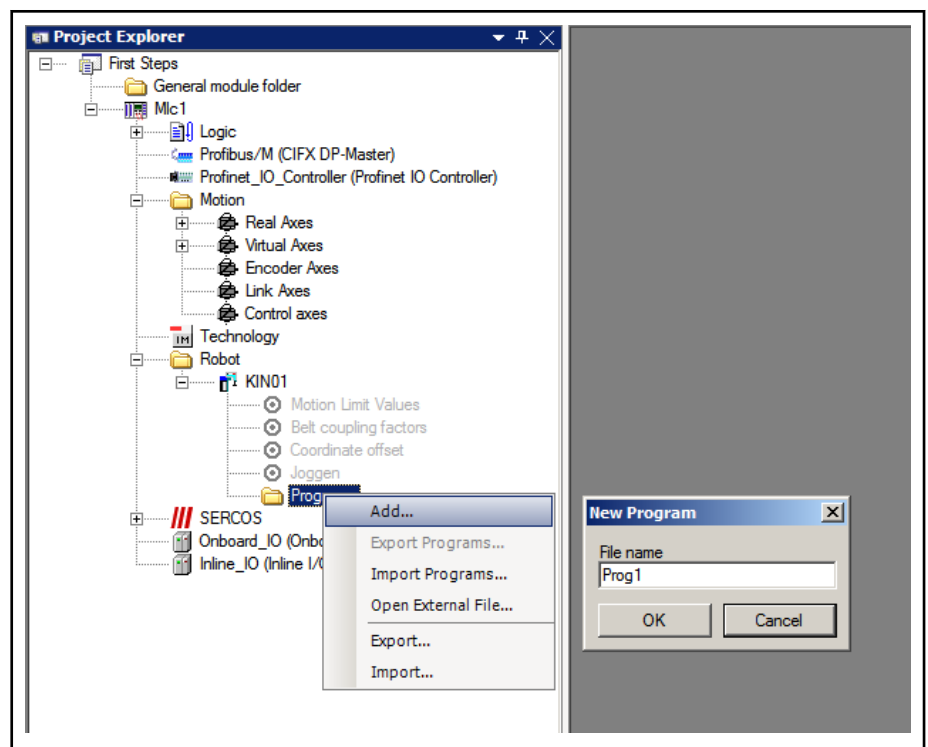


Fig.10-69: Creating RoCo motion program

8. Editing and compiling the loaded RoCo motion program.

For the compilation, right click in the editor window and select "Compile" in the context menu.

The objective is to keep the program simple to have a running program as soon as possible. The intention is to demonstrate the general operational function without going into the complexity of other functions.



For more information on the RoCo programming, refer to /23/, [Required and Supplementing Documentations](#), page 14.

Kinematics

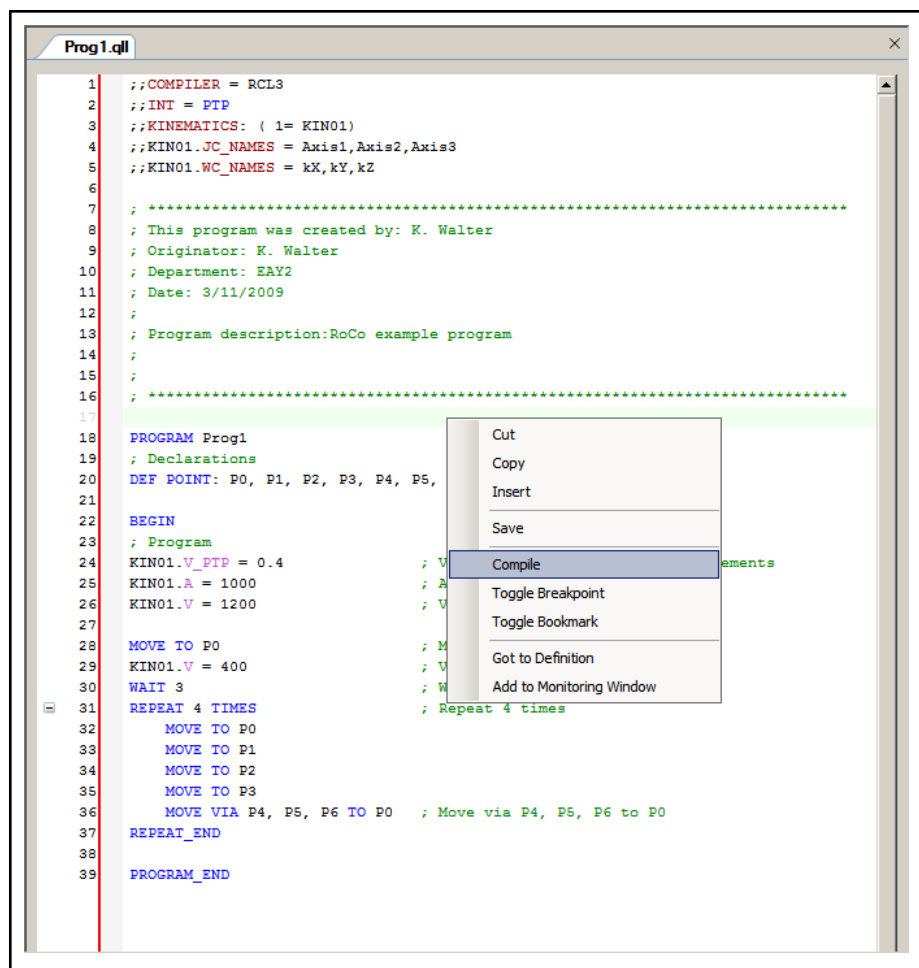


Fig.10-70: Editing and compiling RoCo motion program

The compiler sends a status message to an output window (in this case "No Errors"). Any program errors are listed here indicating the error and the respective line number.

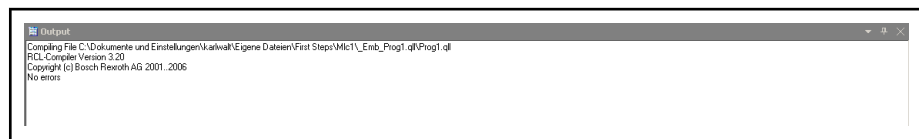


Fig.10-71: Output window of the compiler

9. Editing point file.

After the compilation, a point file is created automatically if the declaration "DEF POINT" was used in the program.

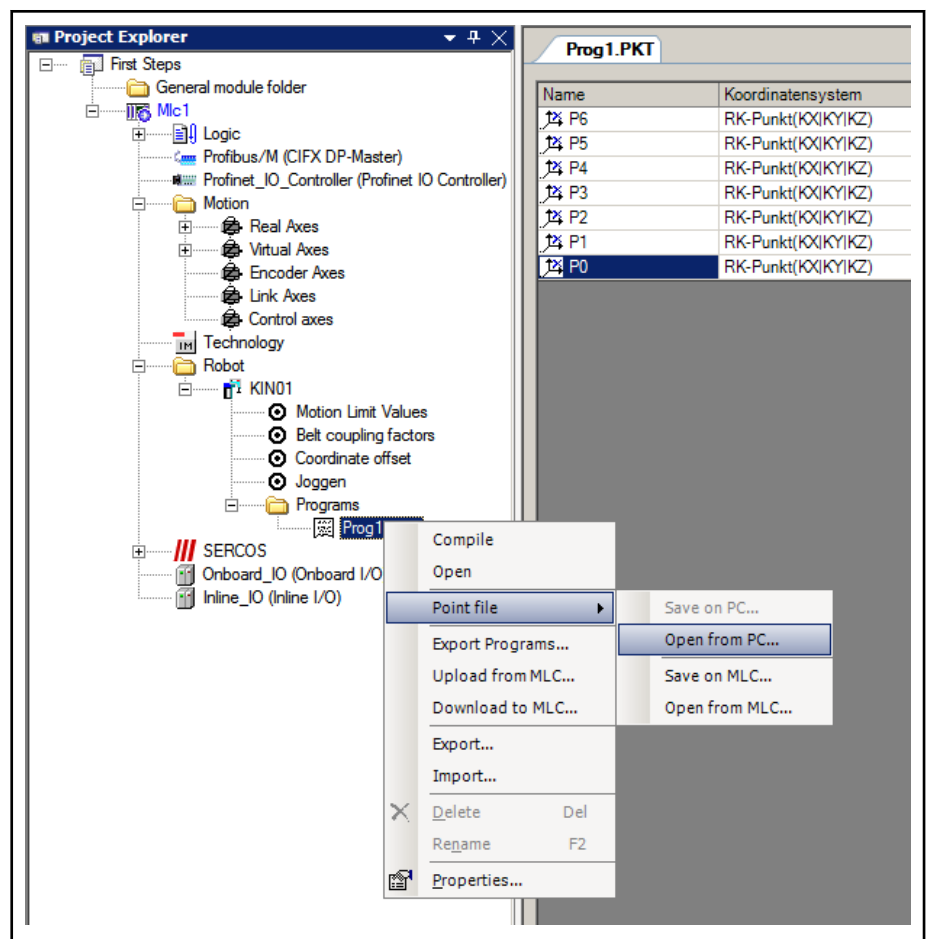


Fig.10-72: Point file created after compilation

In this table, the coordinate values (kX, kY, kZ) have to be entered for each defined point (P0...P6).

When closing the window, a query is displayed whether the values entered in the file are to be saved. Right click in the window to save the file values at any given point in time.

10. Loading motion program, point file, etc. to the control.

Kinematics

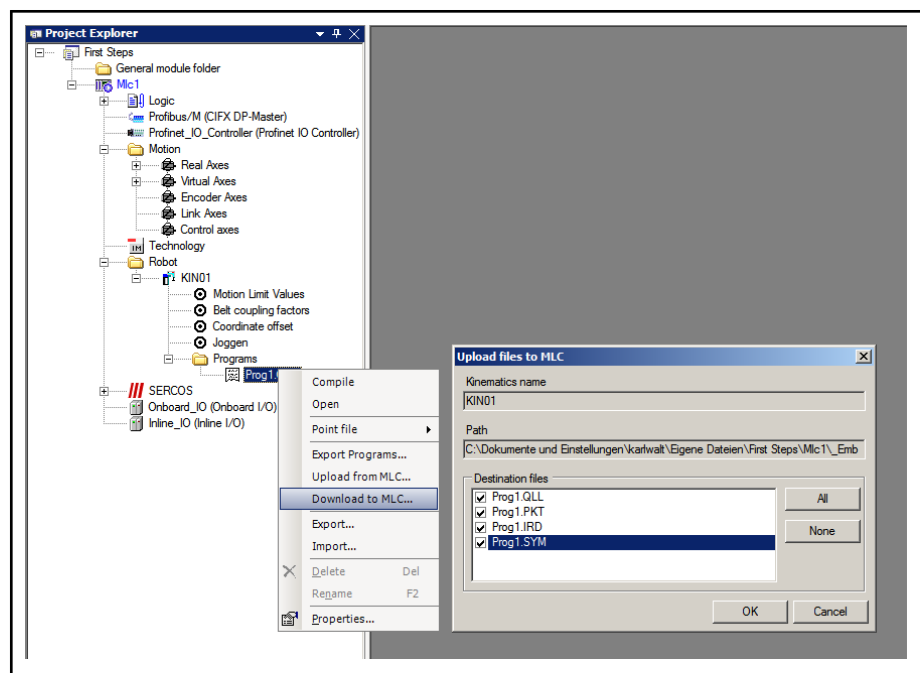


Fig. 10-73: Loading motion program, the point file, etc. to the control



If no SRAM module exists (CFL01.1-Y1) or the control has no on-board S-RAM, this step is to be repeated after each control reboot since the data is located in the volatile memory.

11. RoCo in the PLC program.

Only the minimum PLC code required for the RoCo program execution is described here.

There is a small user interface (visualization) in the example project to control the PLC part. Thus, all operating actions required, from enabling the power supply for real drives up to the RoCo program start and kinematic state can be read as well.

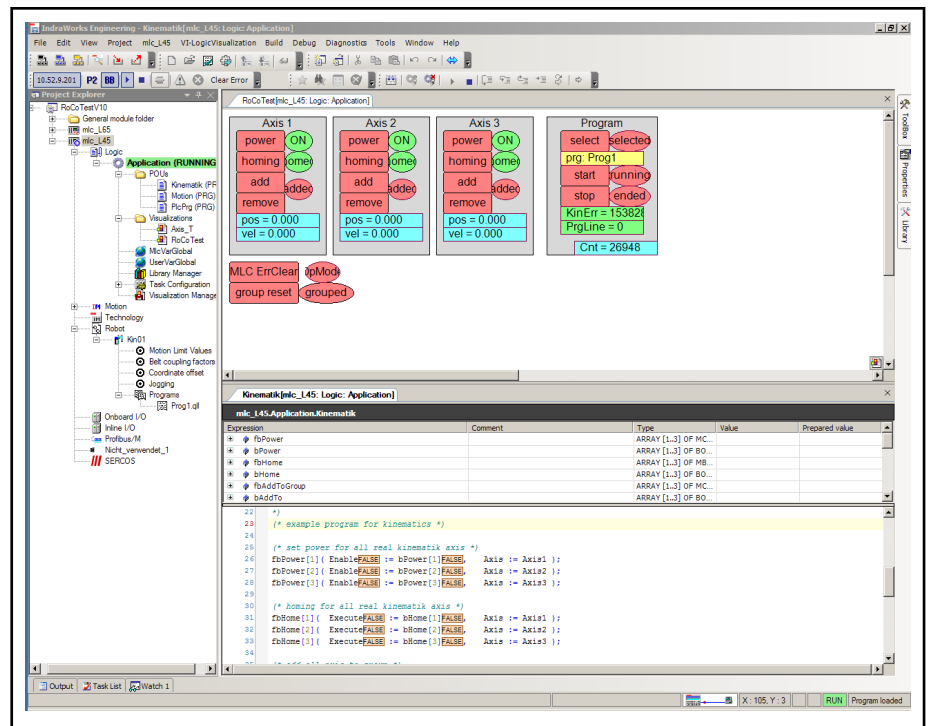


Fig.10-74: RoCo in PLC program

Brief description of the PLC functionality in the programmed sequence.

1. Connecting power.

Execute this command only for real drives.

This bit is always set for virtual axes.

Whether power can be connected to an axis, refer to the "AxisData[n].Axis_Power" bit.

2. Executing homing.

Homing is only required for real drives without absolute encoder evaluation.

The status if an axis is homed can be seen at the bit "AxisData[n].Axis_Homed".

3. Assign kinematic axes.

All axes belonging to the defined kinematics are grouped in this step.

The grouped axes cannot traverse using the normal PLC motion blocks, but receive their traversing commands directly from the RoCo motion program.

This is only possible after all axes belonging to the kinematics are grouped.

Whether all required axes are grouped, refer to the status "KinData[n].dwKinExtStatus_i.3=FALSE" (incomplete).

4. Removing axes from the kinematics.

This is required to traverse the individual axes outside the RoCo motion program.

Kinematics

5. Selecting RoCo motion program.

To execute a RoCo motion program, it has to be selected. This has to be performed after each program end and after each program stop.

6. Starting RoCo motion program.

This command starts the selected program.

7. Stopping RoCo motion program.

All axis motions are stopped and the selected program is reset.

In case of error:

Delete IndraMotion MLC control errors, drive errors and kinematic errors.

11 CamBuilder Connection

11.1 CamBuilder Functionality

The CamBuilder application is integrated in IndraWorks. It is described in detail in the documentation "Rexroth IndraWorks 12VRS CamBuilder" (see /7/, [Required and Supplementing Documentations](#), page 14).

12 Oscilloscope Connection

12.1 Oscilloscope Functionality

The oscilloscope functionality is integrated in IndraWorks. It allows the signal recording of the control or axes (drives).

The use of the oscilloscope is described in detail in the documentation "IndraWorks 12VRS Engineering" in the chapter "Operating the Oscilloscope Function" (see /5/ [Required and Supplementing Documentations](#), page 14).

13 User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts

13.1 General Information

In 12VRS or higher, a user-defined language (language 3) can be used to load parameter names, units and diagnostics and to display the menu structure of the display in this language.

The texts are stored in XML files:

Parameter names and units in the parameter file, diagnostics in the diagnostic file and the menu for the display in the menu file.

13.2 File Management

For the online application, store the files on the user partition of the Compact Flash card in the folder "Languagefile".

For the offline application, the files are saved in the folder "Languagefile" in the IndraWorks project directory below the selected control (e.g. IndraMotionMlc1).

The file name is arbitrary and can be transferred to the system for the files currently used with the parameters C-0-0002 for the parameter file, C-0-0003 for the diagnostic file and C-0-0004 for the menu file.

Different files with different language packages can thus be saved and the desired language files can be selected.

13.3 File Structure

The XML files require the following basic structure:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<device vendor="xxx" name="xxx" desc="xxx" hw="xxx" bsp="xxx">
  <packagelist>
    <package name="">
      <idx>0</idx>
      <key>Axis-Parameter</key>
      <additional>
        <A-0-0000 name="Userdefined Text 1"/>
        <A-0-0001 name="Userdefined Text 2"/>
        <A-0-0002 name="Userdefined Text 3"/>
      </additional>
    </package>
  </packagelist>
</device>
```

Fig. 13-1: XML language files; here parameter file

The first line is the documentation declaration and has to be executed exactly as shown in the example above.

The attributes of the "Root" node "device" are secondary for file parsing.

The structure below the "Root" node is important. The structure is equal for all files.

- The first node is "packagelist". This node declares several packages
The packages contain nodes with "idx", "key" and "additional"

User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts

- The index "idx" is continuous for multiple packages and starts with "0"
- The key "key" indicates the current general search text
- All texts are stored in their individual child node below the "additional" node. The node name indicates the parameter or diagnostic number and the user-defined text is stored behind, below the attribute "name"

- Parameter.xml**
- Each parameter type (A, C, K, O, N, M) and the units are provided with an individual package
 - The keys for the individual packages have to be named as follows:
 - A: "Axis parameter"
 - C: "Control parameter"
 - K: "Kinematic parameter" (only IndraMotion MLC)
 - O: "Oscilloscope parameter"
 - N: "ProgLimSwitch parameter"
 - M: "Touch probe parameter"
 - Units: "Units"
 - The node name below the "additional" node names the parameter or unit
 - The user-defined text is stored under the "name" attribute at the respective parameter/unit

For the present node names for the units, refer to [chapter 13.5 "Node Names for Units" on page 498](#).

- Diagnoses.xml**
- Only one package is required for the diagnostic file
 - The key for the package has to be "Diagnosis"
 - The node name below the "additional" node designates the diagnostic number
 - The user-defined text is stored under the "name" attribute at the respective diagnostic number
- Menue.xml**
- Only one package is required for the display file
 - The key for the package has to be "Menu"
 - The node name below the "additional" node names the menu item or display text
 - The user-defined text is stored under the "name" attribute at the respective menu item/display text

Node names

- [Node names for Units, page 498](#)
- [Node Names for Display Texts, page 499](#)

13.4 Function

Select the language using the parameter C-0-0001.

Furthermore, German, English, user-defined and "empty" (easier editability) XML files can be created for parameters, diagnostics and display.

C-0-0001 can assume the following values:

- 0: Loads German from the installation storage location
- 1: Loads English from the installation storage location
- 99: Parses XML files with the user-defined language and applies it as language 3
- 100: Creates XML files from the installation storage location

All other values are disabled.

Options to load a user-defined language

1. **When booting** the control, the value of the parameter C-0-0001 is set to "99".

All parameter names, diagnostics and display texts are read from the files defined in the parameters C-0-0002, C-0-0003 and C-0-0004 and saved in the installation storage location under the user value (language 3).

2. **At runtime**, the value of the parameter C-0-0001 is set to "99".

It is not important whether the control is in phase 0, phase 2 or phase 4. Thus, different language packages can be loaded at runtime considering the file parameters.

If certain parameters, units, diagnostics or display texts are not defined in the XML file or if errors occur while parsing or searching in files, the English texts are saved as default language under the user value.

All texts found are stored in the user-defined language. This ensures that every text is displayed.

The **XML files can only be created from the installation storage location at runtime**:

If the value of the parameter C-0-0001 is set to "100", this is only a command to create the files and the value is reset to the previous existing value after creation.

For parameters, diagnostics and the display, German, English, user-defined and "empty" XML files are created each. The "empty" files only contain the basic structure of the XML files and no texts at the attributes. Thus, they can be edited more easily.

Restrictions

- Parameter names are restricted to a maximum length of 60 characters
- diagnostics to 61 characters and
- units to a maximum length of 20 characters



These lengths refer to characters only without the zero terminator "\0".

Characters, that is "BYTE" and not UTF-8-letter. These can either be 1, 2, 3 or 4 BYTE!

Three different maximum lengths are defined at the display:

1. 8 characters for menu items (that means non-scrolling displays).
2. 60 characters for scrolling displays such as menu, note and error texts.

User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts

3. 4 characters for three special texts, that is the change of first number of the IP address, subnet mask and gateway address.

The lengths also refer to characters without zero terminator.

If texts are stored in the XML files exceeding the maximum length, the text is limited to a maximum length and the rest is cut off.

Special features Due to UTF-8 coding of XML files, Greek, Cyrillic, Chinese and Japanese characters can also be displayed like German umlauts.

This does not apply for the display, since the standard 7-bit ASCII character set can only be implemented here.

13.5 Node Names for Units

No.	Node-Name	Unit German	No.	Node name	Unit German
1	Unit_1_100K	1/100K	42	Unit_MasterUnit	nicht_aktiviert
2	Unit_1_GradK	1/K	43	Unit_Mbaud	Mbaud
3	Unit_1_min	1/min	44	Unit_mH	mH
4	Unit_1000_min	1000/min	45	Unit_min	min
5	Unit_A	A	46	Unit_mm	mm
6	Unit_A_eff	A eff	47	Unit_mm_min	mm/min
7	Unit_A_V	A/V	48	Unit_mm_s	mm/s
8	Unit_A_Vmin	A/Vmin	49	Unit_mm_ss	mm/s ²
9	Unit_BeschlVorst_Lin	mN/(mm/s ²)	50	Unit_mm_sss	mm/s ³
10	Unit_Byte	Byte	51	Unit_mm_U	mm/U
11	Unit_DrehMomKonst_Lin	N/A	52	Unit_mNms2_rad	mNm/(rad/s ²)
12	Unit_Grad	Grad	53	Unit_ms	ms
13	Unit_Grad_s	Grad/s	54	Unit_N	N
14	Unit_Grad_ss	Grad/s ²	55	Unit_N_Aeff	N/Aeff
15	Unit_Grad_sss	Grad/s ³	56	Unit_Nm	Nm
16	Unit_GradC	°C	57	Unit_Nm_A	Nm/A
17	Unit_GradF	°F	58	Unit_Nm_Aeff	Nm/A eff
18	Unit_h	h	59	Unit_Nm_rad_s	Nm/(rad/s)
19	Unit_Hz	Hz	60	Unit_Nm_U_min	Nm/U/min
20	Unit_Hz_100A	Hz/100A	61	Unit_NoUnit	--
21	Unit_Hz_s	Hz/s	62	Unit_Ohm	Ohm
22	Unit_in	in	63	Unit_Polpaare	Polpaare
23	Unit_in_min	in/min	64	Unit_Prozent	%
24	Unit_in_s	in/s	65	Unit_rad_ss	rad/s ²
25	Unit_in_ss	in/s ²	66	Unit_rad_sss	rad/s ³
26	Unit_in_sss	in/s ³	67	Unit_s	s
27	Unit_in_U	in/U	68	Unit_ss	s ²

User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts

No.	Node-Name	Unit German	No.	Node name	Unit German
28	Unit_Inkr	Inkr	69	Unit_TP	TP
29	Unit_Inkr_ss	Inkr/s ²	70	Unit_TP_U	TP
30	Unit_Inkr_sss	Inkr/s ³	71	Unit_Traegheit_Lin	Kg
31	Unit_Inkr_Umdr	Inkr/U	72	Unit_U	U
32	Unit_inlbf	inlbf	73	Unit_U_min	U/min
33	Unit_kBaud	kBaud	74	Unit_U_s	U/s
34	Unit_KByte	KByte	75	Unit_us	µs
35	Unit_kgm2	kgm ²	76	Unit_V	V
36	Unit_kHz	kHz	77	Unit_V_A	V/A
37	Unit_kW	kW	78	Unit_V_eff	V eff
38	Unit_lbf	lbf	79	Unit_V_ms	Geschw * ms
39	Unit_m	m	80	Unit_VreglerPVerst_Lin	N/(mm/min)
40	Unit_mA	mA	81	Unit_Watt	Watt
41	Unit_mA_rad_s	mA/(rad/s)			

13.6 Node Names for Menu Texts (Display Texts)

No.	Node name	Text German	Characters
1	Menue_MlState	ML:OK	8
2	Menue_PlState	SPS:OK	8
3	Menue_RiState	RI:OK	8
4	Menue_DpState	DP:OK	8
5	Menue_Ethernet	Ethernet	8
6	Menue_ShowIpAAA	IP=	4
7	Menue_ShowSubNetAAA	SN=	4
8	Menue_ShowGatewayAAA	GW=	4
9	Menue_OK	OK: ?	8
10	Menue_Device	Geraet	8
11	Menue_Info	Info	8
12	Menue_OHCCtrlFanReset	ResetOHC	8
13	Menue_Hardware	Hardware	8
14	Menue_PlState	SPS APP	8
15	Text_IpAddress	IP-Adresse	60
16	Text_SubnetMask	Subnet-Maske	60
17	Text_GatewayAddress	Gateway-Adresse	60
18	Text_MacAddress	Mac-Adresse	60
19	Text_Firmware	Firmware	60

User-Defined Languages for Parameter Names, Units, Diagnostics and Menu Texts

No.	Node name	Text German	Characters
20	Text_HwMatNr	Materialnummer	60
21	Text_HwTypeCode	Typen Code	60
22	Text_HwMatIndexNr	Index	60
23	Text_HwSerNr	Seriennummer	60
24	Text_Temperature	Temperatur	60
25	Text_OHCctrl	OHC Steuerung	60
26	Text_OHCctrlFan	OHC Luefter	60
27	Text_ShowDHCPState	DHCP	60
28	Text_ShowDHCPStateEnabled	AKTIVIERT	60
29	Text_ShowDHCPStateDisabled	DEAKTIVIERT	60
30	Text_ChangeDHCPStateEnable	AKTIVIEREN	60
31	Text_ChangeDHCPStateDisable	DEAKTIVIEREN	60
32	Text_DHCPisActiv	DHCP ist aktiviert, keine Aenderungen moeglich	60
33	Text_LoadBaseParameter	Kommando: Netzwerk-StandardEinstellungen laden	60
34	Text_LoadBaseParameterDone	Kommando-Ausfuehrung fertig	60
35	Text_Axis	Achse	60
36	Text_Sercos	sercos Device	60
37	Text_Kinematic	Kinematik	60
38	Error_IpAddress	FEHLER: Lesen IP-Adresse	60
39	Error_SubnetMask	FEHLER: Lesen Subnet-Maske	60
40	Error_GatewayAddress	FEHLER: Lesen Gateway-Adresse	60
41	Error_MacAddress	FEHLER: Lesen Mac-Adresse	60
42	Error_BaseParameter	FEHLER: Waehrend Ausfuehrung des Kommandos	60
43	Error_SerialNumber	FEHLER: Lesen Seriennummer	60
44	Error_OHCctrl	FEHLER: Lesen OHC STRG Zeit	60
45	Error_Temperature	FEHLER: Lesen Temperatur	60
46	Error_TemperatureScaling	FEHLER: Lesen Temperatur Skalierung	60
47	Error_FwVersion	FEHLER: Lesen Firmware Version	60
48	Error_TypeCode	FEHLER: Lesen Typen Code	60
49	Error_Plc	FEHLER: Lesen SPS Applikations-Name	60

14 Remote Access via Modem

14.1 General Information

14.1.1 Definition of Terms and Introduction

Term definitions

Service PC	A PC that can either remotely control a remote client PC or establish a direct connection to a control via a router in the remote network
Client PC	A PC that can be remotely controlled from the service PC
I-Remote	Remote maintenance software that is subject to license, a component of the IndraWorks automation platform
MoRoS	Modem Router Switch. A router-switch combination with integrated modem manufactured by INSYS Microelectronics GmbH. Data telegrams are routed between modem and switch

Fig. 14-1: Definition of terms

The controls "IndraLogic XLC" and "IndraMotion MLC" are all provided with an Ethernet interface. The control can be programmed and applications can be commissioned (debugging, monitoring) using this interface. Data can also be exchanged with other controls as well as operating and visualization devices via this Ethernet interface. The protocols are IP-based.

The task of enabling remote access to services of the controller consists of connecting two separated IP networks.

Various standardized methods are available for this purpose.

- To connect via internet, see [chapter 14.1.2 "Connection via Internet" on page 501](#)
- To connect point-to-point (analog or digital telephone network, GSM or UMTS mobile network), see [chapter 14.1.3 "Point-to-Point Connection" on page 502](#)

More information

Description	Parts number	Reference
Rexroth IndraWorks I-Remote, Remote Control Software	R911310612	Via internet under http://www.boschrexroth.com on "Products" ⇒ "Documentation and Resources"
Bosch Rexroth service sites online	-	Via internet under http://www.boschrexroth.com on "Service"

Fig. 14-2: More information

14.1.2 Connection via Internet

When establishing a connection via internet, the networks of both communication partners are connected to the internet. As soon as a connection has been established, the services of the control can be used by every other internet user. This is a risk of unauthorized access to the control. "VPN connections" use the "unsecure" internet but the data is encrypted by "tunneling". Thus, a secure connection is established.

Remote Access via Modem

There are many tunnel variants (for instance: PPTP, OpenVPN, IPsec) which are based on different logs. They differ in configuration complexity and security.

14.1.3 Point-to-Point Connection

With a point-to-point connection, one of the communication partners specifically logs on to the remote network via a dial-up connection. The dial-in server can either be a correspondingly configured PC (client PC) or a stand-alone device (for instance the INSYS Microelectronics "MoRoS"). In both cases, the control network is connected to the dial-up connection network. It is important that both communication partners use the same technology (analog or ISDN).



A dial-up connection between ISDN and analog telephone connections is not possible.

The cabled point-to-point connection is preferably used due to its excellent availability, simple configuration and minor security risks. The following chapters only describe the cabled point-to-point connection.

14.1.4 Accessibility to the Control

IndraControl controls can be accessed differently. Only two access variants are described here. The access variants differ with regard to their configuration effort as well as the required communication devices and remote control possibilities:

- For a dial-up connection between service PC and client PC, see [chapter 14.2.4 "Possible Remote Maintenance Tasks" on page 503](#)
- For a dial-up connection between service PC and MoRoS, see [chapter 14.3 "Dial-Up Connection Between Service PC and MoRoS" on page 503](#)

14.2 Dial-Up Connection Between Service PC and Client PC

14.2.1 General Information

A dial-up connection is established between service PC and client PC in this configuration. The client PC is also provided with the Ethernet connection to the control and an RS232 connection to an IndraDrive if necessary. For an exemplary connection establishment, see [fig. 14-3 "Dial-up connection between service PC and client PC" on page 502](#).

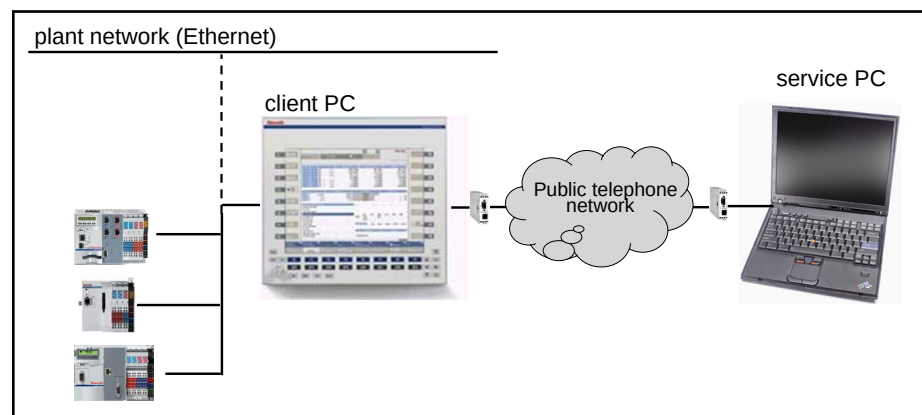


Fig. 14-3: Dial-up connection between service PC and client PC

Remote Access via Modem

The client PC hosts all software to be operated (IndraWorks Engineering, IndraWorks Operation) and the respective data (engineering project, firmware files, drive data).

The inputs and outputs of the client PC are lengthened to the service PC via modem connection. Thus, it can perform the same operation from a remote service PC as it can from the client PC.

14.2.2 Required Equipment

To operate a dial-up connection between service PC and client PC, the following equipment is required:

- Suitable remote maintenance software

Bosch Rexroth provides I-Remote as remote maintenance software. I-Remote is copied when installing IndraWorks. The installation can be directly started from the folder with the same name of the program group IndraWorks.



A license is required to operate I-Remote.

- Suitable modems

The selection of suitable modems depends on the connection technology (analog, ISDN, GSM) that is available at both end points and has to be equal and it also depends on the requested degree of protection

14.2.3 Installation and Configuration

To install and configure a dial-up connection between a service PC and a client PC, proceed with the following steps.

1. Install I-Remote on service and client PC.
2. Install the modem on the client PC and configure the dial-up connection.



The I-Remote installation and configuration of the the RAS services are described in the I-Remote manual, parts number R911310612, type code "DOK-IWORKS-IREMOTE*V01-AW01-EN-P". To install and configure I-Remote, read chapter 4 and to configure RAS services, read chapter 9 in the I-Remote manual.

14.2.4 Possible Remote Maintenance Tasks

During a dial-up connection between service and client PC, the client PC is remote-controlled. Thus, all programs installed on the client PC can be remote-controlled. A prerequisite is a completely functioning operating system on the client PC.

14.3 Dial-Up Connection Between Service PC and MoRoS

14.3.1 General Information

A dial-up connection is established between the service PC and the MoRoS in this configuration type. MoRoS establishes a direct connection to the subnet of the controls. The programs (such as IndraWorks Engineering) and data (such as IndraWorks projects) are now located on the service PC that can directly diagnose and maintain the control connected via MoRoS. An exemplary connection establishment is shown in the following figure.

Remote Access via Modem

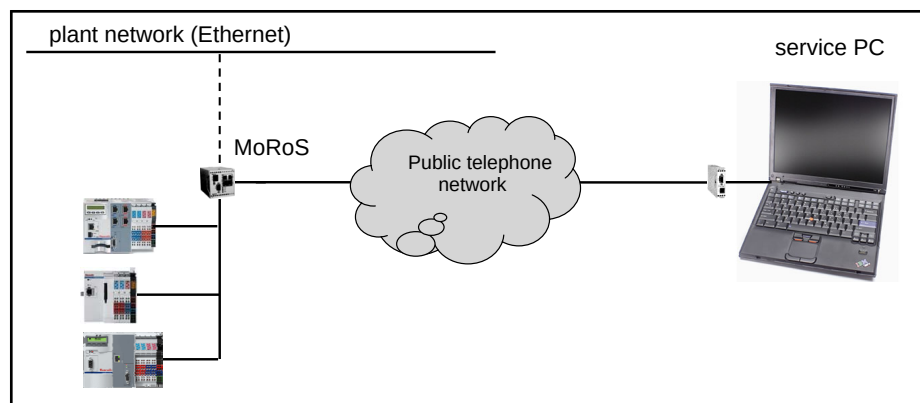


Fig. 14-4: Dial-up connection between service PC and MoRoS

14.3.2 Required Equipment

To operate a dial-up connection between service PC and client PC, the following equipment is required:

- Suitable remote maintenance software

Bosch Rexroth provides I-Remote as remote maintenance software. I-Remote is copied when installing IndraWorks. The installation can be directly started from the folder with the same name of the program group IndraWorks.



A license is required to operate I-Remote.

- Suitable router with dial-in option ("MoRoS" for example)
- Suitable modem for the service PC

The selection of suitable modems depends on the connection technology (analog, ISDN, GSM) that is available at both end points and has to be equal and it also depends on the requested degree of protection.

If a notebook is used as service PC, a modem is generally built-in

14.3.3 Remote Maintenance Actions

In general, all IndraWorks online functions are possible. Not included are drive commissioning and drive diagnostics via the serial interface.

14.3.4 Installation and Configuration

General Information

The following instructions refer to the MoRoS (Modem Router Switch) of the company INSYS Microelectronics. Similar devices are also provided by other parties. The principles in the instructions, especially the network configurations can also be used for similar devices. The most decisive factor to select a certain device is its suitability for the given environmental conditions.

Initial MoRoS Configuration

MoRoS is configured via an internet browser. The network settings of the MoRoS and the configuration PC have to be compatible. Upon delivery, MoRoS is set as DHCP server. A PC getting its IP configuration as usual via net can load it directly from MoRoS. Thus, it gets a suitable setting to the MoRoS.

- Connect the communication PC and MoRoS with an Ethernet cable

Remote Access via Modem

- Check the IP configuration by executing "ipconfig" in the Windows command prompt (**Start ► Programs ► Accessories ► Command prompt**)

The port connected to the MoRoS should be provided with an IP address from the range 192.168.1.xx. The configuration interface of the MoRoS is triggered via IP address displayed under "Standard gateway".

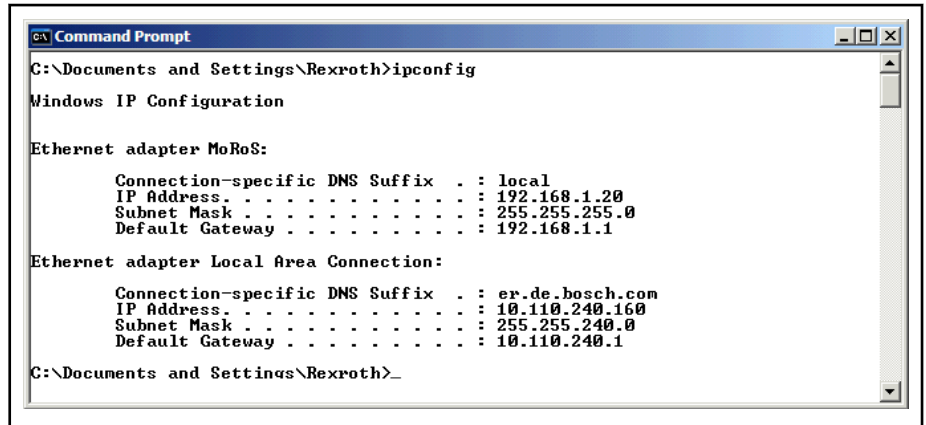


Fig. 14-5: Output of the "ipconfig" command in the command prompt at set up MoRoS

If no other addresses are displayed, use it for configuration.

If no useful addresses are displayed, check the cable connection between PC and MoRoS first. If the connection is not ok, the DHCP server functionality might be switched off. Read in the MoRoS user manual how to set the device in its state upon delivery.

Start the internet browser and enter the URL "http://192.168.1.1" in the address field.

Enter "insys" as user name and "moros" as password for authentication purposes.

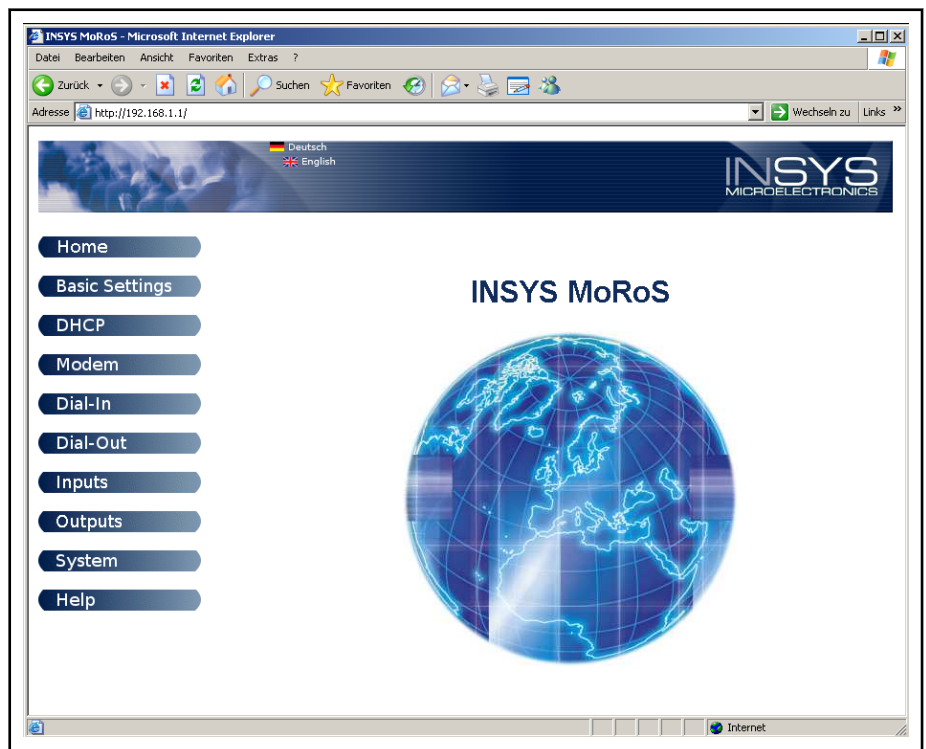


Fig. 14-6: Homepage of the MoRoS configuration interface

Remote Access via Modem

The homepage of the configuration interface provides access to all parameters. The following section describes only parameters that have to be changed to create the example configuration. For more detailed information, refer to the MoRoS manual.

Network Configuration

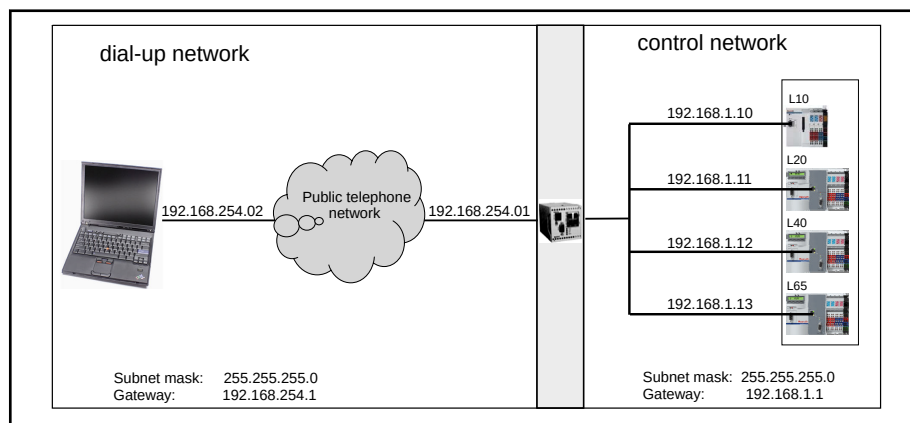


Fig. 14-7: Coupling two spatially separated subnets via router with built-in modem (MoRoS)

The MoRoS is configured to accept incoming dial-up connections, (Dial-In). When dialing in, a subnet from the address range 192.168.254.x is set up.

The controls connected to Ethernet ports form a second subnet from the address range 192.168.1.x together with the MoRoS.

The IP addresses of both networks have to differ.

MoRoS Network Configuration

The network settings for the integrated switch can be reached from the main menu via the menu item "DHCP".

DHCP

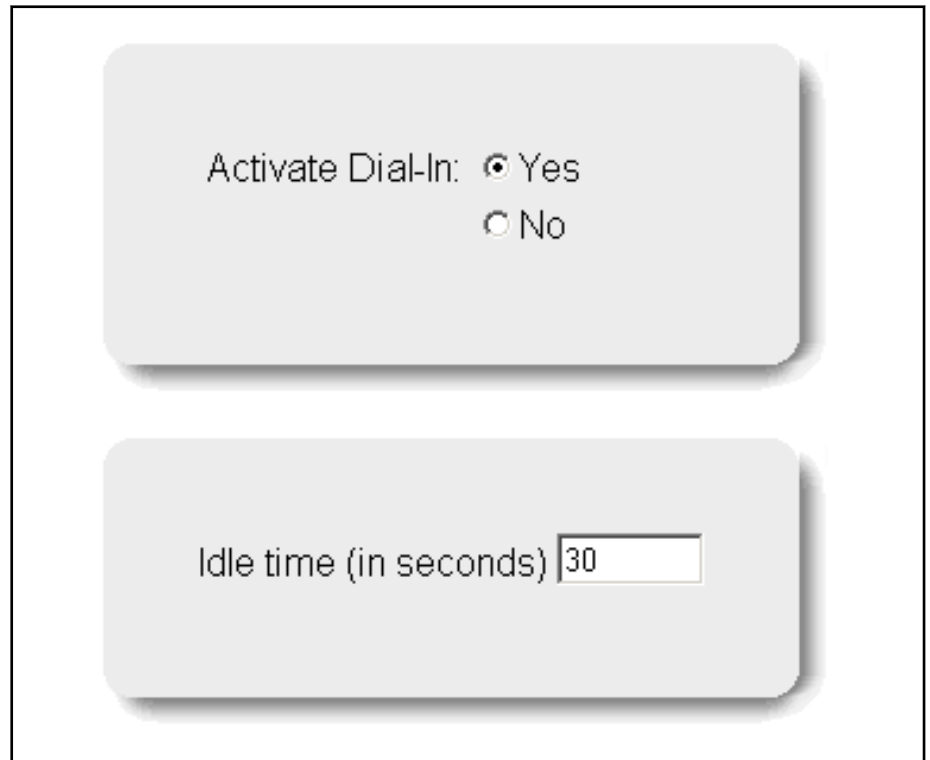
Fig. 14-8: DHCP settings

DHCP is activated upon delivery. MoRoS specifies IP addresses between 192.168.1.20 and 192.168.1.254. The controls IndraLogic XLC and

Remote Access via Modem

IndraMotion MLC operate with fix IP addresses. If the settings remain, the range from 192.168.1.2 to 192.168.1.19 is available for the controls.

Dial-In The network configuration for the dial-up connection is opened from the main menu via "Dial-In". Apart from the addresses for authentication purposes and the maximum "idle time", the IP addresses of the dial-up connections are set there if not specified by the device dialing-in.

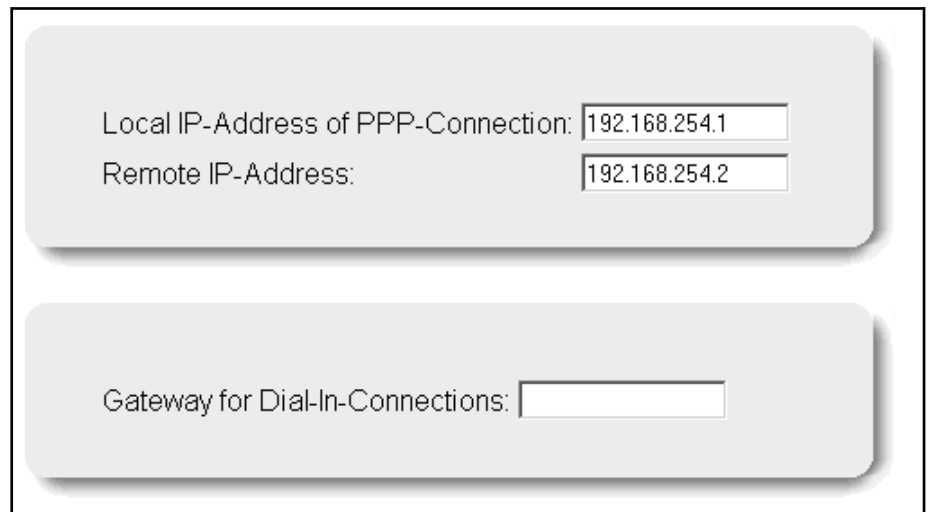


Activate Dial-In: ☒ Yes
☐ No

Idle time (in seconds)

Fig. 14-9: Enable incoming connection

IP settings



Local IP-Address of PPP-Connection:
Remote IP-Address:

Gateway for Dial-In-Connections:

Fig. 14-10: IP settings of the dial-up connection

Remote Access via Modem

Access data

Authenticate Dial-In: ☒ Yes
☐ No

Authentication for Dial-In:

Username: Password: Authentication with:

		PAP	CHAP
insys	moros	☒	☐
		☐	☐
		☐	☐
		☐	☐

Fig. 14-11: Access data of the incoming connection

Network Configuration of the Controls

The IndraMotion MLC controls use static IP addresses. To modify the network settings of the control, refer to the System Description manual, chapter "Display and Operating Components".

In the example in [fig. 14-7 "Coupling two spatially separated subnets via router with built-in modem \(MoRoS\)" on page 506](#), three controls are connected to MoRoS:

Control	IP address	Subnet Mask	Gateway address
L1	192.168.1.10	255.255.255.0	192.168.1.1
L2	192.168.1.11	255.255.255.0	192.168.1.1
L3	192.168.1.12	255.255.255.0	192.168.1.1

Fig. 14-12: Network Configuration of the Controls

Configuration Service PC

The following steps are required for the configuration.

1. Set up a modem
2. Set up a dial-up connection on the service PC
3. Settings in IndraWorks

Setting up a modem on the service PC

A modem to dial in the MoRoS has to be connected to the service PC on which IndraWorks is installed. There are no special requirements. Notebooks often have an integrated and already set up modem. If a modem has to be set up, refer to chapter 8 of the I-Remote user manual.

If the modem should be operated behind a private branch exchange, the modem might not detect a dial tone after accepting the call. In this case, switch off the dial tone detection. The dial tone detection is switched off at the modem properties via the dial option "Wait for dial tone". If still no dial tone is detected after these settings have been made, enter the initialization command "ATX3" under "Advanced".

Setting up a dial-up connection on the service PC

The dial-up connection is set up on the service PC as follows:

1. Open the network connections in the control panel.

Remote Access via Modem

2. Start the wizard for new connections in the "File" menu by selecting "New Connection..."



Fig. 14-13: Starting window of the connection wizard

3. Select connection type and method in the following dialogs.

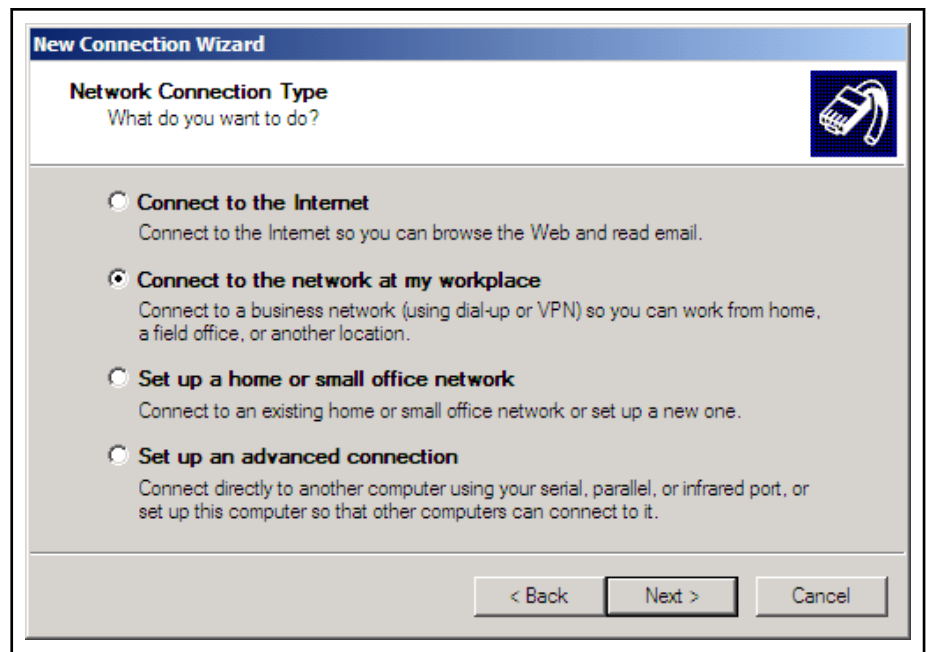


Fig. 14-14: Selecting network connection type

Remote Access via Modem

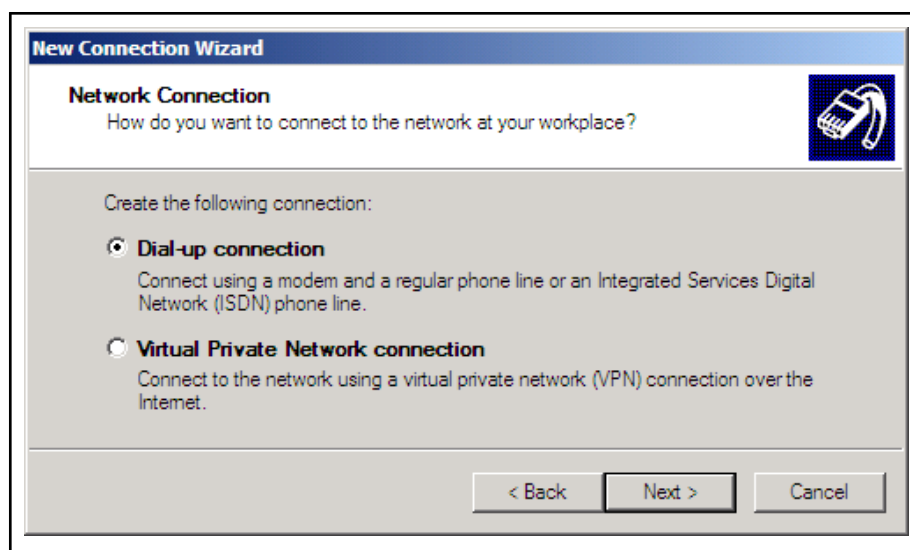


Fig. 14-15: Selecting network connection

4. Specify a name for the connection to be established.

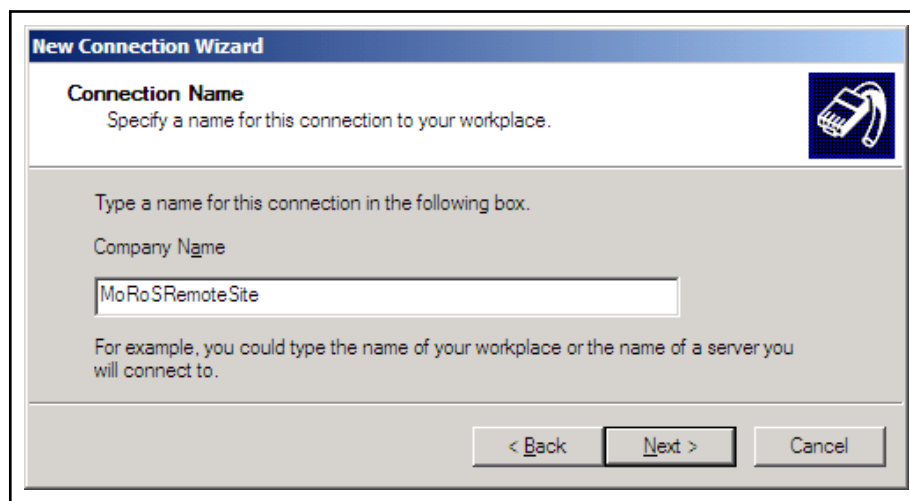


Fig. 14-16: Specifying connection name

5. Enter the phone number under which MoRoS is reachable.

Remote Access via Modem

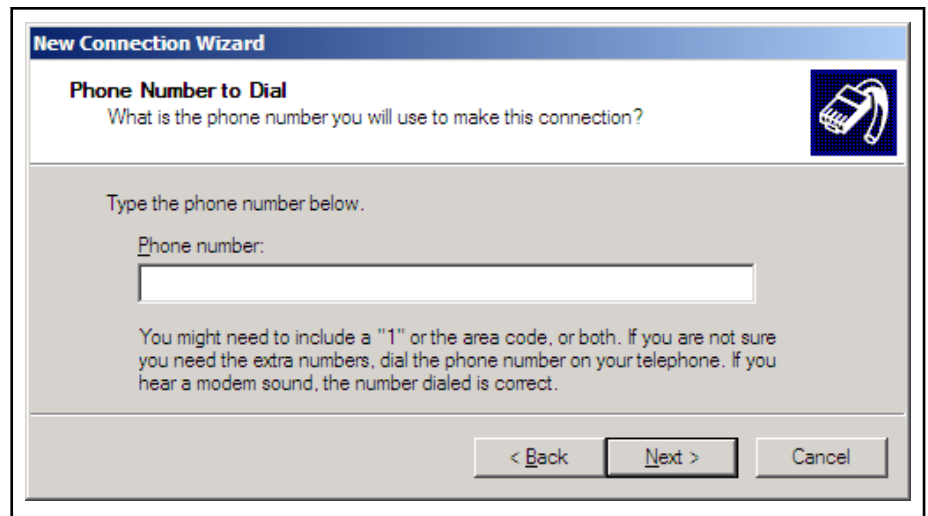


Fig. 14-17: Entering phone number

6. Close the wizard with **Finish**.

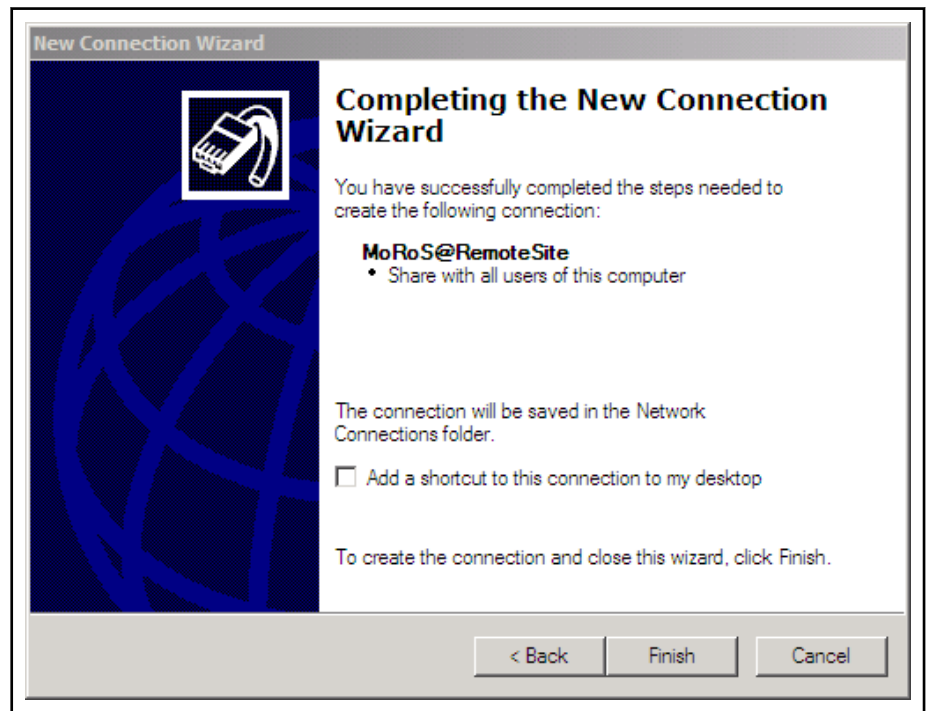


Fig. 14-18: Closing connection wizard

7. The "Connect" dialog opens. Access data can now be entered.

Remote Access via Modem

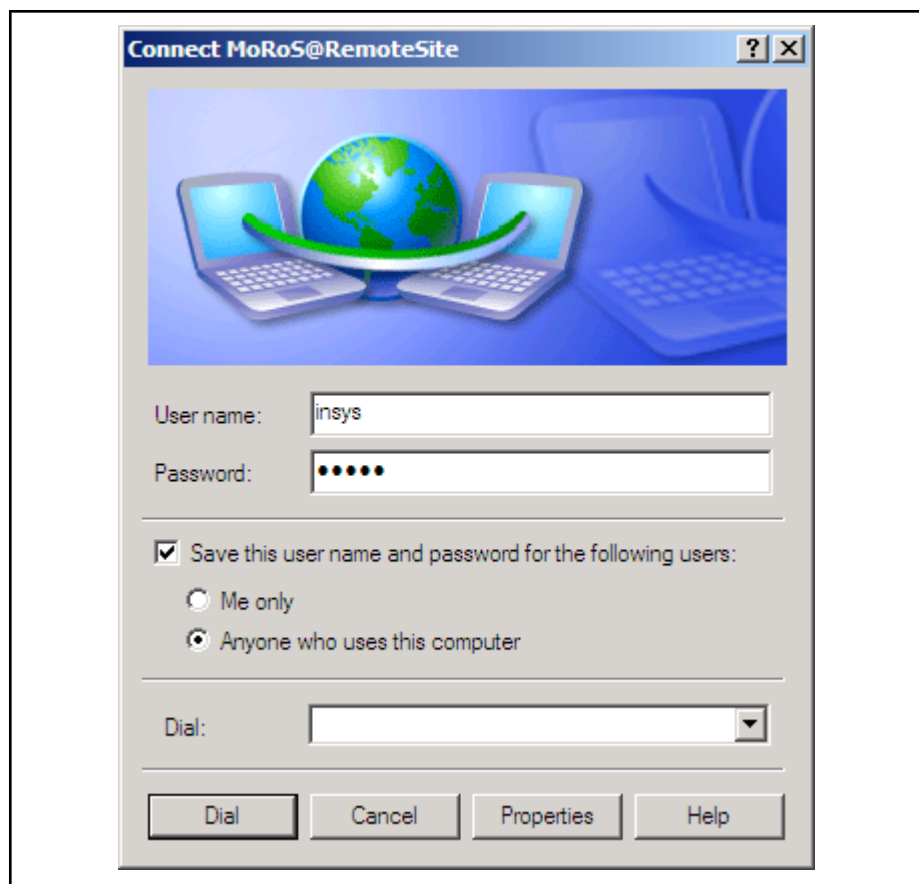


Fig. 14-19: Entering access data

The connection is established via the **Dial** button.

Settings in IndraWorks

The IP address of the control has to be set at the communication parameters in order to allow IndraWorks installed on the service PC to connect to a remote control. In this example configuration, it is 192.168.1.10 (see [fig. 14-12 "Network Configuration of the Controls" on page 508](#)) for the control L1.

An Engineering session with a control connected via a dial-up connection does not differ from one in the local net. Since data transfer is slower, firmware should only be updated at a stable connection.

15 Service and Support

Our worldwide service network provides an optimized and efficient support. Our experts offer you advice and assistance should you have any queries. You can contact us **24/7**.

Service Germany Our technology-oriented Competence Center in Lohr, Germany, is responsible for all your service-related queries for electric drive and controls.

Contact the **Service Helpdesk & Hotline** under:

Phone: **+49 9352 40 5060**
Fax: **+49 9352 18 4941**
E-mail: service.svc@boschrexroth.de
Internet: <http://www.boschrexroth.com>

Additional information on service, repair (e.g. delivery addresses) and training can be found on our internet sites.

Service worldwide Outside Germany, please contact your local service office first. For hotline numbers, refer to the sales office addresses on the internet.

Preparing information To be able to help you more quickly and efficiently, please have the following information ready:

- Detailed description of malfunction and circumstances resulting in the malfunction
- Type plate name of the affected products, in particular type codes and serial numbers
- Your contact data (phone and fax number as well as your email address)

Index

A

Abbreviations.....	19
About this documentation.....	13
Information representation	18
Validity of the documentation	13
Activate watchdog.....	209
Active	
Watchdog	209
Add device	
Information	125
Logging	123
Order number	125
Add POU to task.....	209
Application phases.....	13
Applications on control.....	122
Archive project.....	53
Axes in IndraMotion	
Operation modes for synchronous motions with electronic gear function	372
Single-axis modes	365
Axes – Introduction and overview.....	227
Axis (IndraDrive) - Dialog	
Cam table	295
Cam table, general information	295
Mechanical gear	280
Axis (IndraDrive) – Dialog	
Compensation functions/corrections	293
Drive-integrated safety engineering	323
Gear with phase synchronization - Dynamic synchronization	320
Gear with phase synchronization - Master axis position offset	319
Gear with velocity synchronization - Dynamic synchronization	317
Gear with velocity synchronization - Master axis position offset	315
Axis – Context menu	
Communication, signal control word	339
Communication, signal status word	337
Diagnostics	341
Diagnostics, average value filter display	343
Diagnostics, clear errors	342
Diagnostics, Error/diagnostic memory	342
Diagnostics, status	341
Initial commissioning	328
Axis - Dialog	
Cam table, gear settings	297
Dimensional reference - Optional encoder .	284
Drive Halt	283
E-Stop function	288
Initial values	291
MotionProfile, gear settings	303
Scaling/units extended	278
Axis – Dialog	
Axis configuration	321

A

...Axis – Dialog	
Brake	273
Cam table, cam table phase offset	298
Cam table, cam table selection	299
Cam table, dynamic phase offset	298
Cam table, dynamic synchronization -	
Phase synchronization	299
Cam table, master axis position offset	296
Closed-loop control	281
Compensation functions/corrections	293
Compensation functions/corrections, backlash on reversal correction	294
Drive-integrated command value generator	294
Error reaction	285
Error reaction, drive	285
Error reaction, power section	287
Error reaction can be switched off	286
Establish dimensional reference	283
FlexProfile	307
FlexProfile, cam table phase offset	310
FlexProfile, dynamic synchronization -	
Phase synchronization	313
FlexProfile, gear settings	309
FlexProfile, general information	307
FlexProfile, master axis position offset	309
FlexProfile, overview	311
Gear with phase synchronization - Gear settings	319
Gear with phase synchronization - General information	317
Gear with velocity synchronization - Gear settings	316
Gear with velocity synchronization - General information	314
Jog	327
Measuring systems - Settings motor encoder	274
Measuring systems - Settings of optional encoder	275
Motion limit values	289
MotionProfile	300
MotionProfile, cam table phase offset	303
MotionProfile, dynamic phase offset	304
MotionProfile, dynamic synchronization -	
Phase synchronization	306
MotionProfile, general information	300
MotionProfile, master axis position offset ...	302
MotionProfile, MotionProfile overview	304
Motor	272
Motor temperature monitoring	273
Position switching point	323
Settings of status messages	281
Touch probe	324

Index

A

Axis - Dialogs	
Overview	268
Axis name, encoder axis.....	264
Axis name, real axis.....	237
Axis number, controller axis.....	250
Axis number, encoder axis.....	264
Axis number, real axis.....	237
Axis number, virtual axis.....	245
Axis types.....	227
Axis types and supported function blocks.....	229

B

Basic sequences, general information.....	39
Boot the control.....	40
Byte order.....	224

C

CamBuilder connection.....	491
Cam Pool.....	53
Cam table.....	295
CMP ID.....	123
Command value decoupling	
Axis state dialogs for real axes	363
Display	361
Mode of operation	360
Commission and operate.....	39
Commissioning and operation.....	193
Complete backup	
Complete backup	184
Complete data backup.....	183
General Information	183
Complete project backup	
Restore a project	189
Configuration	
IndraDrive with offline wizard	236
Configuration file	
Network settings	46
Configuration of the measuring encoder.....	265
Configuration task.....	209
Configure drive bus (sercos II).....	118
Container libraries.....	195
Context menu of the project folder.....	52
Control	
"Motion" node	100
"Robot" folder	115
"Robot" folder, context menu	115
Context menu items	119
Folder Profinet IO-Controller folder	116
Function modules folder	117
Inline I/O folder	116
Logger	123
Onboard I/O folder	116
Overview on nodes	95
Profibus/M folder	116
sercos node	117

C

...Control	
Technology node	104
Control – "Logic" node	
Subfolders	96
Control – "Logic" node	
Application – Toolbar	98
Context menu	97
Context menu, "Application" subfolder	99
Control – "Motion" node	
Context menu	101
Context menu, Motion state	104
Control – "sercos " node	
Context menu	118
sercos - Toolbar	119
Control – "Technology" node	
Context menu	105
Dialogs	107
Options menu	106
Control - Context menu	
Diagnostics, task list/configuration	149
Control – Context menu	
Cam Explorer	127
Communication, interfaces	126
Communication, TCP settings	125
Compare/Merge	176
Compare/Merge, Axes	177
Compare/Merge, Control	177
Compare/Merge, Function modules	178
Compare/Merge, Robot	177
Device data	163
Device data, archive	163
Device data, Explorer	170
Diagnostics, Advanced properties	143
Diagnostics, clear errors	144
Diagnostics, Cycle times	102
Diagnostics, device status	141
Diagnostics, Error/diagnostic memory	144
Diagnostics, Invalid parameters	147
Diagnostics, task list/configuration	150
Export	172
Import	174
Language settings	126
Parameters	133
Print	180
Print preview	178
Runtime licenses	178
Simulation	172
Control IndraMotion MLC – Context menu	
Device data, restore	167
Controller axes and supported function blocks.	229
Controller axis – Context menu	
Diagnostics	255
Diagnostics, clear errors	256
Diagnostics, status	255
Initial commissioning	255

C

Controller axis - Context menus	
Menu items	254
Controller axis - Dialogs	
Overview	253
Controller axis - General information.....	248
Control memory deletion.....	43
Create a link axis.....	257
Create control.....	53
Create controller axis.....	248
Central controller step 3: Hardware link	250
Central controller step 4: Signal assign- ment	251
Create axis in Project Explorer	248
Last step: Summary	252
Step 1: General settings	249
Step 2: Configuration	250
Create encoder axis (real master axis).....	262
Create axis in Project Explorer	262
Create function module.....	60
Create Inline I/Os.....	64
Create kinematics.....	88
Create onboard I/Os.....	62
Create PLC program	
Compile program	195
Declare variables	193
Edit instruction	194
Load PLC program to the control	195
Open Plc_Main object	193
Create Profibus I/Os.....	71
Create Profinet IO controller.....	71
Create project.....	51
Create real axis.....	234
Create sercos III I/Os.....	72
Create virtual axis (virtual master axis).....	243
Creating a control	53
Cross-over connection.....	47
Cycle times.....	102
Cyclic sercos data channel.....	336
Cyclic task.....	209

D

Delete applications.....	122
Device category.....	125
Device dialog	
Applications	122
Device editor	
Applications	122
Information	125
PLC settings	124
Device Editor.....	122
Log	123
Device name.....	125
Device order number.....	125
Device-specific library.....	195
Device status.....	141
Device type.....	125

D

Device version.....	125
Display and operating keys.....	44
Documentation of library function blocks.....	195
Documentation structure.....	13
Drive address, sercos.....	236, 264
Drive firmware update.....	329
Drive password – Change.....	335
Drive system.....	23

E

Electric drive system.....	23
Electronic cams.....	396, 408, 417
Cam solutions	396
Centralized and decentralized cam con- cept	399
Electronic cam	398
Fields of application	398
FlexProfile, basic functionalities	399
FlexProfile, configuration	418
FlexProfile, configuration using IndraWorks	419
FlexProfile, configuration using PLC func- tion blocks	418
FlexProfile, event control	414
FlexProfile, Flex step	403
FlexProfile, motion laws	401
FlexProfile, profile configuration	400
FlexProfile, select coordinate system	407
FlexProfile, synchronous operating states .	399
FlexProfile, time master	401
Electronic motion laws.....	399
Enable sercos bus.....	47
Encoder axis - Context menus	
Menu items	267
Encoder axis - Dialogs	
Overview	266
Encoder axis - General information.....	262
Error reaction of the axis.....	234
Event.....	209
Task	209
Event logging.....	123
Event task.....	209
Exchange drive/firmware.....	329
Exchange firmware.....	161
External event for tasks.....	209
External library.....	195

F

Firmware management.....	161
Firmware update.....	161
FlexProfile	
IndraDrive	307
FlexProfile, assign MotionProfile and axis.....	417
FlexProfile, switch and synchronize.....	408
FlexProfile editor.....	418

Index

F

Free-running task..... 209

G

General information, definition of terms and introduction..... 501

H

Hardware commissioning..... 39

Hidden libraries..... 195

I

I/O..... 222

I/O configuration..... 222

Important instructions on use..... 21

IndraDrive..... 295

IndraWorks

Interface 50

IndraWorks – startup..... 48

IndraWorks – Startup..... 48

Information on the device..... 125

Initial Values - Real Axis..... 291

Install and start IndraWorks..... 48

Install IndraWorks

Technical requirements 48

Intended

state-of-the-art 21

Intended use..... 21

Areas of use and application 21

Internal library..... 195

Interval for task..... 209

IP Address - Initial setting..... 44

IP communication..... 76

Automatic IP address generation 84

Basics 77

Configure bridge 85

Default setting 78

Firewall 86

IndraWorks commissioning dialogs 82

More notes on the use 87

Overview 76

Parameters 85

Ports 86

L

Late linking..... 195

Library..... 195

Library directory..... 195

Library management

Convert IndraLogic 1.x libraries 201

Create libraries 200

Install on the system and integrate into a project 197

Integrate external code 201

Library overview 196

L

...Library management

Library versions 199

Referenced libraries 198

Unique access to library blocks (name-space) 199

Library manager..... 195

Library menu..... 195

Container 195

Device-specific reference 195

Documentation 195

External 195

General 195

General information 195

Hidden 195

Internal 195

Namespace 195

Referenced 195

Target system-specifically 195

Unique access 195

Version 195

Library repository..... 195

Library version..... 195

Link axis - Context menus

Menu items 260

Link axis - Dialog

Settings: scaling/units 277

Link axis – dialogs

Overview 260

Link axis – General information..... 257

Load basic parameters

IndraDrive 334

Load offline parameters to the device..... 356

Log..... 123

Logger in the runtime system..... 123

M

M-0-0000, Dummy M-parameter..... 431

M-0-0001, Logical probe number..... 431

M-0-0002, Touch probe name..... 432

M-0-0005, Input selection..... 432

M-0-0006, Signal selection..... 432

M-0-0007, Control word..... 433

M-0-0009, Osci: list of all M-parameters that can be oscilloscoped..... 434

M-0-0010, List of all M parameters..... 434

M-0-0016, Project identification number..... 435

M-0-0021, Status word..... 435

M-0-0024, Command: Activation..... 436

M-0-0130, Measured value, positive edge..... 436

M-0-0131, Measured value, negative edge..... 436

M-0-0132, Difference value..... 437

M-0-0140, Time stamp, positive edge..... 437

M-0-0141, Time stamp, negative edge..... 437

M-0-0200, Dead-time compensation, positive edge..... 438

M

M-0-0201, Dead-time compensation, negative edge.....	438
M-0-0202, Starting position of the expectation window, positive edge.....	438
M-0-0203, End position of the expectation window, positive edge.....	438
M-0-0204, Starting position of the expectation window, negative edge.....	439
M-0-0205, End position of the expectation window, negative edge.....	439
M-0-0206, Maximum number of marker failures.....	439
M-0-0224, Number of marker failures, positive edge.....	439
M-0-0225, Number of marker failures, negative edge.....	440
M-0-0401, Input status.....	440
M-0-0405, Enabling.....	440
M-0-0409, Counter of measured values, positive edge.....	441
M-0-0410, Counter of measured values, negative edge.....	441
M-0-0411, Counter of difference values.....	441
Max. number of tasks.....	208
Memory alignment.....	223
Memory organization.....	223
Menu overview to operate display and keyboard	45
ML_AXISDATA_SM.....	337, 338, 340
ML_TechBase.library.....	38
ML_TechInterface.library.....	37
ML_TechMotion.library.....	38
MLs-E0A0001.....	381
MLs-E0A0002.....	381
MLs-E0A0003.....	381
MLs-E0A0004.....	381
MLs-E0A0005.....	381
MLs-E0A0010.....	381
MLs-E0A0012.....	382
MLs-E0A0013.....	382
MLs-E0A0014.....	382
MLs-E0A0015.....	382
MLs-E0A0020.....	382
MLs-E0A0021.....	382
MLs-E0A0022.....	382
MLs-E0A0023.....	383
MLs-E0A0030.....	383
MLs-E0A0100.....	383
MLs-E0C0001.....	383
MLs-E0C0002.....	383
MLs-E0C0003.....	383
MLs-E0C0004.....	384
MLs-E0C0005.....	384
MLs-E0C0006.....	384
MLs-E0C0010.....	384
MLs-E0C0014.....	384
MLs-E0F0010.....	384

M

MLs-E0F0011.....	384
MLs-E0F0012.....	385
MLs-E0F0020.....	385
MLs-E0K0001.....	385
MLs-E0K0002.....	385
MLs-E0K0003.....	385
MLs-E0K0005.....	385
MLs-E0K0006.....	386
MLs-E0K0010.....	386
MLs-E0K0012.....	386
MLs-E0K0013.....	386
MLs-E0K0020.....	386
MLs-E0K0021.....	386
MLs-E0K0022.....	386
MLs-E0K0023.....	387
MLs-E0K0024.....	387
MLs-E0K0025.....	387
MLs-E0K0026.....	387
MLs-E0K0027.....	387
MLs-E0K0028.....	387
MLs-E0K0029.....	387
MLs-E0K0030.....	387
MLs-E0K0031.....	388
MLs-E0K0032.....	388
MLs-E0K0033.....	388
MLs-E0K0034.....	388
MLs-E0K0035.....	388
MLs-E0K0036.....	388
MLs-E0K0037.....	389
MLs-E0K0038.....	389
MLs-E0K0039.....	389
MLs-E0K0040.....	389
MLs-E0K0041.....	389
MLs-E0K0042.....	389
MLs-E0K0043.....	389
MLs-E0L0012.....	390
MLs-E0L0020.....	390
MLs-E0L0021.....	390
MLs-E0S0010.....	390
MLs-E0S0012.....	390
MLs-E0S0013.....	390
MLs-E0S0014.....	391
MLs-E0S0022.....	391
MLs-I0X0001.....	391
MLs-I0X0002.....	391
MLs-W0A0020.....	391
MLs-W0A0021.....	392
MLs-W0A0022.....	392
MLs-W0A1000.....	392
MLs-W0C0020.....	393
MLs-W0F0020.....	393
MLs-W0F0021.....	393
MLs-W0F0022.....	393
MLs-W0K0001.....	393
MLs-W0L0020.....	393
MLs-W0L0022.....	394

Index

M

MLs-W0L0023.....	394
MLs-W0L0024.....	394
MLs-W0L0026.....	394
MLs-W0L0027.....	395
MLs-W0S0013.....	395
MLs-W0S0020.....	395
MLs-W0S0021.....	395
MLs-W0S0022.....	395
MLs-W0S0023.....	395
Modes for project editing... 346, 347, 348, 355, 356	
Modes to operate a real axis.....	359
Monitor task.....	201
Motion configuration validation	
Error messages	380
Motion functionality.....	345
Motion mode	345
Parameters	345
Programmable limit switch	442
Motion limit values	
Virtual axis	289
MotionProfile	
IndraDrive	300
Motion state.....	104
Motion tasks according to VDI2143.....	401
Multi-device.....	150
Brief description of the multi-device table . .	151
Configuration file to deactivate the	
Profibus DP slaves	158
Introduction	150
Multi-device functionality in detail	152
Multilingualism XLC/MLC.....	495
File management	495
File structure	495
Function	497
Node names for menu texts	499
Node names for units	498
Multi-tasking.....	201
I/O turn-around times	213
Task configuration	204
Task configuration, configuration exam- ples	205
Task editor	207
Task Execution Viewer	212
Task list/configuration	210

N

Names.....	19
Names and abbreviations.....	19
Namespace for library.....	195
Network settings via the configuration file.....	46
Notes on how to use deactivated and park- ing axes.....	360

O

OEM partition.....	36

O

Offline mode.....	346
Offline parameterization.....	347
Online mode.....	346
Operating strategy.....	53
Oscilloscope connection.....	493

P

Parameter editor.....	133
Parameter editor toolbar.....	134
Parameter group.....	135
Parameter group toolbar.....	135
Parameters - Export.....	136
Parameters - import.....	139
Parameters – Save parameters in drives.....	136
PELV.....	27
Placeholder for library.....	195
PLC log.....	123
PLC logger.....	123
PLC program	
Create	193
Pointer addressing.....	225
Pointer programming.....	223
POU	
Assign to task	209
Open from task editor	209
POUs for task.....	209
Priority	
Task	209
Processing types.....	413
Program call.....	209
Project information.....	195
Properties - Controller axis (IndraDrive).....	255
Properties - Encoder axis.....	268
Properties - link axis.....	262
Properties - Real axis.....	242
Protective extra-low voltage.....	27

R

Ready contact.....	40
Real axes, HNC	
Function package	232
Real axes, HydraulicDrive	
Firmware	232
Real axes, IndraDrive	
Firmware	231, 232
Function package	231, 232
Real axes and supported function blocks.....	229
Real axis	
Active	360
Deactivated	359
Parked	359
Real axis (IndraDrive).....	231
Real axis (IndraDrive) - Context menu	
Parameters, Load basic parameters	334
Parameters, save mode	335

R

...Real axis (IndraDrive) - Context menu	
Parameters, save parameters in the drive .	336
Real axis (IndraDrive) – Context menu	
Communication, cyclic sercos data channel	336
Parameter, drive password – Change password	335
Real axis (IndraDrive) - Dialog	
Dimensional reference - Motor encoder	283
Real axis – Context menu	
Menu items	241
Real axis - Dialogs	
Overview	238
Real axis - General information.....	231
Reference	
Applications	122
Referenced libraries.....	195
Remote access via modem.....	501
Dial-up connection between service PC and client PC	502
Dial-up connection between service PC and client PC, installation and configuration	503
Dial-up connection between service PC and client PC, possible remote maintenance tasks	503
Dial-up connection between service PC and client PC, required equipment	503
Dial-up connection between service PC and MoRoS	503
Dial-up connection between service PC and MoRoS, installation and configuration .	504
Dial-up connection between service PC and MoRoS, remote maintenance actions .	504
Dial-up connection between service PC and MoRoS, required equipment	504
General information, accessibility to the control	502
General information, connection via internet	501
General information, point-to-point connection	502
Replace hardware.....	47
Repository.....	195
Response time	
Inline I/O	64
Onboard I/O	62
RobotControl.....	445
Compiler	463
Creating a RoCo project	477
Debugger	465
Folder "Robot" - Dialog for axis lengths and coupling	450
Folder Robot – Dialog: Coordinate Offset .	453
Folder Robot – Dialog: Coupling Factors	
Belt	452

R

...RobotControl	
Folder Robot – Dialog: Jogging	454
Folder Robot – Dialog: Motion Limit Values	452
Folder Robot – Dialog: Workpiece coordinate Assignment	458
Folder Robot – Kinematic dialogs	450
General information	445
Kinematics - context menu	448
Kinematics – context menu: Diagnostics	449
Kinematics – context menu: Setup	448
Menu structure: Kinematics	446
Options, RoCo compiler	467
Options, RoCo-Editor	469
Options, RoCo online settings	469
RCL editor	459
Robot – Dialog: Safe zones folder	456
Settable options for RoCo	467
RoCo-Editor.....	469
RoCo editor, keywords.....	476
RoCo editor, page margins.....	474
RoCo-Editor, page margins – frame color "in the focus", frame color "without focus".....	475
RoCo-Editor, page margins - highlight brackets.....	475
RoCo-Editor, page margins - highlight line.....	475
RoCo-Editor, page margins - show line numbers.....	475
RoCo editor, text area.....	471
RoCo-Editor, text area - associated brackets... ..	473
RoCo-Editor, text area - cursor settings.....	472
RoCo-Editor, text area - EOL marks.....	474
RoCo-Editor, text area - foreground color, background color and font type of the editor.....	471
RoCo-Editor, text area - foreground color, background color and font type of the page margin.....	475
RoCo-Editor, text area - highlight line.....	473
RoCo-Editor, text area – Selection, Region foreground, Region background, Region style..	471
RoCo-Editor, text editor – Automatic linefeed...	470
RoCo-Editor, text editor – indent width.....	471
RoCo-Editor, text editor – Max. no. of undo.....	470
RoCo Editor, text editor – Show auto-complete upon entry.....	470
RoCo-Editor, text editor – Tabulator size.....	470

S

Safety instructions.....	18
Safety instructions for electric drives and controls.....	23
Save mode.....	335
Save parameters in drive.....	336
Save project.....	53
Scan bus configuration.....	118

Index

S

Sensitivity	
Task watchdog	209
Signal control word.....	339
Signal status word.....	337
Signatures	
Applications	122
Single-tone event.....	209
Special motion laws.....	402, 403
Structure of a project.....	93
Support	
See service hotline	513
Symbols used.....	18
Synchronous operating states.....	399
System overview.....	33
Firmware	36
Hardware	33
Software	37
SYSTEM partition.....	36

T

Target group.....	13
Target system-specific library.....	195
Task configuration	
Max. number of tasks	208
Properties	208
Task configuration properties.....	208
Task editor	
Assign POU	209
Configuration dialog	209
Max.number	208
Sequence	209
Task list.....	209
Task list/configuration.....	149
Task list/Task Execution Viewer.....	150
Task name.....	209
Task properties.....	209
Task system.....	201
System tasks	217
Toolbar of the control.....	94
Touch probe.....	324, 424
Accuracy	427
Data acquisition	427
Expectation window	430
Features	424

T

...Touch probe	
Functional description	425
Function blocks	442
Introduction	424
Overview on M-parameters	431
Transition between the modes	
Adapt the behavior of the axis adjustment dialog	355
Adjustment: command/actual configuration	348
Axis adjustment dialog	355
Axis adjustment dialog, go online	355
Mode switching	347
Turn around times	
Onboard and Fast I/Os	216
Profibus, Profinet and Inline I/Os	214
sercos III I/Os	215

U

Unintended use.....	22
Consequences, Exclusion of Liability	21
Update	
Applications list	122
Update options.....	37
USER partition.....	36

V

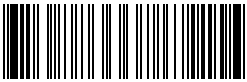
Validate Motion configuration.....	379
Vendor name.....	125
Version control.....	152
Version control of the control.....	182
Introduction	182
Special features	182
Version of a library.....	195
Virtual axes and supported function blocks.....	229
Virtual axis – Context menu	
Menu items	246
Virtual axis - Dialogs	
Overview	246
Virtual axis – General information.....	242

W

Watchdog for tasks.....	209

Notes

Bosch Rexroth AG
Electric Drives and Controls
P.O. Box 13 57
97803 Lohr, Germany
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr, Germany
Tel. +49 9352 18 0
Fax +49 9352 18 8400
www.boschrexroth.com/electrics



R911333848

DOK-MLC***-FUNC****V12-AP04-EN-P