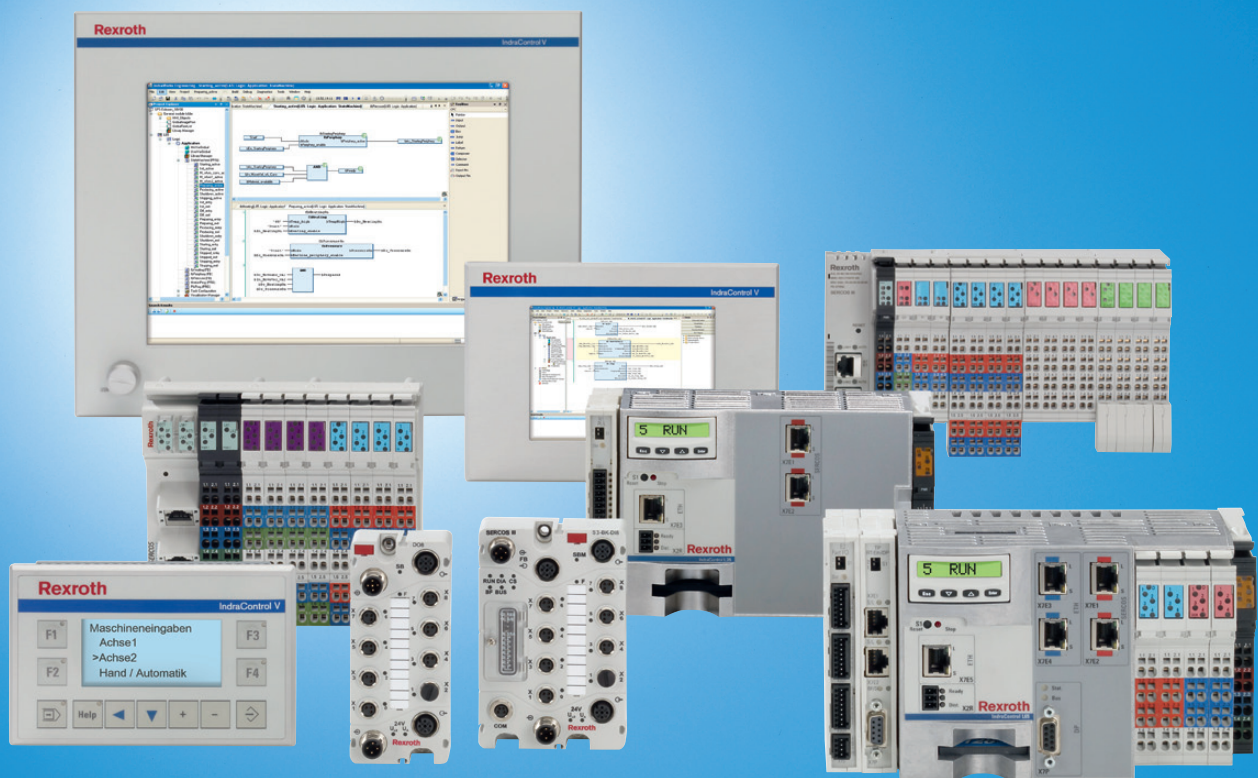


Rexroth IndraLogic XLC 13VRS

Functional Description

Application Description
R911336352

Edition 06



Title Rexroth IndraLogic
XLC 13VRS
Functional Description

Type of Documentation Application Description

Document Typecode DOK-XLC***-FUNC****V13-AP06-EN-P

Internal File Reference RS-60a011e42d89c9d90a6846a501fcc371-5-en-US-1

Change Record

Edition	Release Date	Note
Edition 01	2012-06	First edition for MLC/XLC 13VRS
Edition 02	2013-04	Revision, supplement
Edition 03	2013-06	Revision, translation
Edition 04	2013-08	Robot Control V2
Edition 05	2014-07	Boot/ CleanUp, advanced boot menu Memory alignment MPx18/MPx19 extension Case of error / error categories / SysError / BootStop menu
Edition 06	2015-02	Additions

Copyright © Bosch Rexroth AG 2015

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.

Editorial Department Automation Systems Development XLC System Development GeKue (Ka-Wa/MePe)

Table of Contents

	Page
1 About this documentation.....	15
1.1 Validity of the documentation.....	15
1.2 Documentation structure.....	15
1.3 Using safety instructions.....	16
1.3.1 Safety instructions – Structure.....	16
1.3.2 Explaining signal words and safety alert symbol.....	16
1.3.3 Symbols used.....	18
1.4 Required and supplementing documentations.....	18
1.5 Names and abbreviations.....	24
1.6 Customer feedback.....	25
2 System overview.....	27
2.1 Hardware.....	27
2.2 Firmware.....	30
2.3 Software.....	31
3 Commissioning and operation.....	33
3.1 General information.....	33
3.2 IndraLogic XLC – Basic Sequences, General Information.....	33
3.3 IndraLogic XLC L25/L45/L65 – Commissioning.....	33
3.3.1 Connecting the control.....	33
3.3.2 Operating the control display.....	35
3.3.3 Deleting control memory.....	37
3.3.4 Setting the IP address.....	38
3.3.5 Adding the Sercos bus.....	41
3.3.6 Replacing hardware.....	42
3.4 Installing and starting IndraWorks.....	42
3.5 Overview on the IndraWorks interface.....	45
3.6 Creating a project.....	46
3.7 Creating a control.....	48
3.8 Creating a function module.....	52
3.9 I/Os and field buses.....	54
3.9.1 Creating Onboard I/Os.....	54
3.9.2 Creating Inline I/Os.....	56
3.9.3 Creating Profibus I/Os.....	63
3.9.4 Creating a Profinet I/O Controller.....	63
3.9.5 Creating Sercos III I/Os.....	64
3.10 IP communication	68
3.10.1 Overview.....	68
3.10.2 Basics.....	70
3.10.3 Default setting.....	70
3.10.4 IndraWorks commissioning dialogs.....	74
3.10.5 Automatic IP address generation.....	76

Table of Contents

	Page
3.10.6	Configuring the bridge..... 77
3.10.7	Involved IP communication parameters..... 77
3.10.8	Ethernet ports and firewall..... 78
3.10.9	More information on the use of Ethernet-based interfaces..... 79
3.11	Axes..... 80
4	Control..... 81
4.1	Project structure..... 81
4.2	Toolbar of the Control "XLC/MLC Info"..... 82
4.3	Control - Nodes..... 82
4.3.1	Overview on nodes..... 82
4.3.2	Logic..... 83
	"Logic" subfolder..... 83
	"Logic" – Context menu..... 84
	Application – Toolbar..... 85
	Application – Context menu..... 86
4.3.3	Motion..... 87
	General information..... 87
	Motion – Context menu..... 88
4.3.4	Onboard I/Os..... 90
4.3.5	Inline I/Os..... 90
4.3.6	Profibus/M..... 90
4.3.7	Profinet I/O controller..... 90
4.3.8	Function modules..... 91
4.3.9	Sercos..... 91
	General information..... 91
	Sercos - Context menu..... 92
	Overview on menu items..... 92
	Setting device..... 93
	Sercos state..... 93
	Scanning bus configuration..... 93
	Configuring Sercos devices..... 93
	Sercos - Toolbar..... 93
4.4	Control – Context menu..... 94
4.4.1	Overview on menu items..... 94
4.4.2	Device editor..... 96
	Device editor - General information..... 96
	Applications 96
	Log..... 97
	PLC settings..... 98
	Information 99
4.4.3	Communication..... 100
	TCP settings..... 100
	Interfaces..... 101
	User control ID..... 101
	Overview..... 101

Table of Contents

	Page
Managing user control ID.....	102
Exchanging hardware.....	103
4.4.4 Language settings.....	103
4.4.5 Cam explorer.....	104
4.4.6 Parameters.....	109
Parameter editor.....	109
Parameter group.....	111
Finding parameters.....	113
Saving parameters in drives.....	113
Parameters - Export.....	113
Parameters - Import.....	115
4.4.7 Diagnostics.....	117
Device status.....	117
Advanced properties.....	119
Clearing errors.....	120
Error/diagnostic memory.....	120
Overview.....	120
Messages.....	120
Filters.....	121
"View" Area.....	122
"Control" Area.....	122
Context menu of the diagnostic list.....	123
Invalid parameters.....	123
Task list/configuration.....	124
Task viewer.....	124
MLPI connections.....	125
4.4.8 Multi-device.....	126
Introduction.....	126
Brief description of the multi-device table.....	127
Multi-device functionality in detail.....	128
Version control system (VCS).....	128
Difference: Multi-device disabled ⇔ enabled.....	129
Basic behavior of the multi-device table.....	129
Describing columns of the multi-device table.....	129
I/O configuration file to disable field bus slaves.....	133
Disabling slaves of different field buses.....	133
Managing the configuration file "UserDefs.cfg".....	134
4.4.9 Firmware management.....	134
4.4.10 Drive firmware management.....	136
4.4.11 Device data.....	139
Archiving.....	139
Restoring.....	143
Explorer.....	145
4.4.12 Simulation.....	147
4.4.13 Exporting.....	148
4.4.14 Importing.....	149

Table of Contents

	Page
4.4.15	Comparing/merging..... 151
	General information..... 151
	Control..... 152
	Axes..... 152
	Function modules..... 152
4.4.16	Runtime Licenses..... 153
4.4.17	Print preview..... 153
4.4.18	Printing..... 155
4.4.19	Control properties..... 157
4.5	Version control of the control..... 159
4.5.1	Introduction..... 159
4.5.2	Special Features..... 159
	Project Management..... 159
	Working with Controls..... 160
	Motion Connection with Version-controlled Project..... 160
	PLC Connection with Version-Controlled Project..... 160
	Using Multi-Device Configurations..... 160
4.6	Complete data backup..... 160
4.6.1	General information..... 160
4.6.2	Complete backup..... 160
4.6.3	Restoring a project..... 165
5	PLC programming..... 169
5.1	General information..... 169
5.2	Creating PLC programs..... 169
5.2.1	Opening "PlcProg object" 169
5.2.2	Declaring variables..... 169
5.2.3	Editing instruction..... 170
5.2.4	Compiling programs..... 170
5.2.5	Loading a PLC program to the control..... 171
5.3	Library management..... 171
5.3.1	General information..... 171
5.3.2	Library overview..... 172
5.3.3	Installing on the system and integrating into a project 173
5.3.4	Referenced libraries..... 174
5.3.5	Library versions..... 174
5.3.6	Unique access to library function blocks (namespace)..... 174
5.3.7	Creating libraries..... 175
5.3.8	Converting IndraLogic 1.x libraries..... 176
5.3.9	Integrating external code..... 177
5.4	Task system 177
5.4.1	General information..... 177
5.4.2	Node "Task configuration"..... 179
	General information..... 179
	Configuration examples..... 181
5.4.3	Task editor..... 183

Table of Contents

	Page
General information.....	183
Configuring a task	184
5.4.4 "Properties" dialog.....	186
5.4.5 Task list/configuration.....	186
5.4.6 Task viewer.....	188
5.4.7 I/O turn-around times.....	189
General information.....	189
Profibus, Profinet and Inline I/Os.....	191
Sercos III I/Os.....	191
Onboard and Fast I/Os.....	192
5.4.8 System tasks.....	193
5.4.9 CPU load of the control.....	198
5.5 I/O access.....	200
5.6 Memory organization and pointer programming.....	202
5.6.1 Memory alignment.....	202
5.6.2 Byte order.....	203
5.6.3 Pointer addressing.....	203
6 Axes.....	207
6.1 Introduction and overview.....	207
6.2 Real axis.....	208
6.2.1 General information.....	208
6.2.2 Error response of axis.....	213
6.2.3 Creating a real axis.....	213
6.2.4 Real axis – Dialogs.....	216
Overview on dialogs.....	216
6.2.5 Real axis – Context menu.....	219
Overview on menu items.....	219
Axis properties.....	220
6.3 Virtual axis.....	222
6.3.1 General Information.....	222
6.3.2 Creating a virtual axis (virtual master axis).....	222
6.3.3 Virtual axis - Dialogs.....	225
Overview on dialogs.....	225
6.3.4 Virtual axis - Context menu.....	226
Overview on menu items.....	226
6.4 Encoder axis.....	228
6.4.1 General Information.....	228
6.4.2 Creating an encoder axis (real master axis).....	228
Creating an axis in the Project Explorer.....	228
Configuring the measuring encoder.....	231
6.4.3 Encoder axis - Dialogs.....	232
Overview on dialogs.....	232
6.4.4 Encoder axis - Context menu.....	232
Overview on menu items.....	232
6.5 Dialogs.....	234

Table of Contents

	Page
6.5.1	Overview on dialogs..... 234
6.5.2	Operation mode settings..... 238
6.5.3	Motor..... 240
6.5.4	Motor temperature monitoring..... 241
6.5.5	Brake..... 242
6.5.6	Measuring systems..... 242
	Settings, Motor Encoder..... 242
	Settings of Optional Encoder..... 244
6.5.7	Settings: scaling/units..... 245
6.5.8	Scaling/units extended..... 246
6.5.9	Mechanical gear..... 248
6.5.10	Closed-loop control..... 249
6.5.11	Settings of status messages..... 250
6.5.12	Drive halt..... 251
6.5.13	Establishing dimensional reference..... 252
	Motor encoder..... 252
	Optional encoder..... 253
6.5.14	Error response..... 254
	Drive..... 254
	Power section..... 255
6.5.15	E-STOP function..... 256
6.5.16	Motion limit values..... 257
6.5.17	Initial values..... 259
6.5.18	Compensation functions/corrections..... 261
	Encoder correction..... 261
	Backlash on reversal correction..... 262
6.5.19	Drive-integrated command value generator..... 263
6.5.20	Cam table..... 263
	General information..... 263
	Master axis position offset..... 265
	Gear settings..... 265
	Cam table phase offset..... 266
	Dynamic phase offset..... 267
	Cam table selection..... 267
	Dynamic synchronization - phase synchronization..... 268
6.5.21	MotionProfile..... 269
	General information..... 269
	Master axis position offset..... 270
	Gear settings..... 271
	Cam table phase offset..... 272
	Dynamic phase offset..... 273
	MotionProfile overview..... 273
	Dynamic synchronization – Phase synchronization..... 274
6.5.22	FlexProfile..... 275
	General information..... 275
	Master axis position offset..... 277

	Page
	Gear settings..... 278
	Cam Table Phase Offset..... 278
	FlexProfile overview..... 279
	Dynamic synchronization - Phase synchronization..... 281
6.5.23	Gear with velocity synchronization..... 282
	General information..... 282
	Master axis position offset..... 282
	Gear settings..... 284
	Dynamic synchronization..... 285
6.5.24	Gear with phase synchronization..... 286
	General information..... 286
	Master axis position offset..... 288
	Gear settings..... 288
	Dynamic synchronization..... 289
6.5.25	Axis configuration..... 290
6.5.26	Drive-integrated safety engineering..... 292
6.5.27	Position switching point..... 292
6.5.28	Touch probe..... 293
6.5.29	Initial Commissioning..... 296
6.5.30	Parameters..... 297
	Loading basic parameters..... 297
	Drive password - Change password..... 298
	Save mode..... 298
	Saving parameters in the drive..... 298
6.5.31	Communication..... 299
	Cyclic Sercos data channel..... 299
	Signal status word..... 300
	Signal control word..... 302
6.5.32	Diagnostics..... 304
	Status..... 304
	IDN list of invalid operating data..... 305
	Clearing Errors..... 306
	Error/diagnostic memory..... 306
	Average value filter display..... 307
6.6	Drive as PLC device..... 308
6.6.1	General information..... 308
6.6.2	Sercos II drive as PLC device..... 309
	Sercos III drive as PLC device – Overview..... 309
	Sercos II drive as PLC device – Parameterization..... 309
	Sercos III drive as PLC device – Configuring cyclic data..... 311
	Sercos III drive as PLC device – PLC programming..... 314
6.6.3	Drive as PLC device – More field buses..... 316
6.7	PowerSupply (power supply) as PLC device..... 317
6.7.1	General information..... 317
6.7.2	Power supply (PowerSupply) as PLC device, parameterization..... 317
6.7.3	Power supply as PLC device – Configuring cyclic data..... 318

Table of Contents

	Page
6.7.4 Power supply (PowerSupply) as PLC device – PLC programming.....	320
7 Motion functionality.....	323
7.1 General information.....	323
7.2 Cycle Times.....	323
7.3 Motion Mode.....	324
7.4 Parameters.....	325
7.5 Parameterization level.....	325
7.6 Modes for project editing.....	326
7.6.1 Introduction.....	326
Offline mode.....	326
Online mode.....	326
Offline Parameterization.....	328
7.6.2 Transitions between modes.....	328
Mode switching.....	328
Adjustment of command/actual configuration.....	329
Scanning bus configuration.....	333
Axis synchronization dialog.....	335
Switching online.....	335
Adjusting the behavior of the axis synchronization dialog.....	335
7.6.3 Loading Offline Parameters to Device.....	337
7.6.4 Comparing control and project configuration.....	338
7.7 Axis modes.....	340
7.7.1 Deactivated axis.....	340
7.7.2 Parking axis.....	340
7.7.3 Notes on the use of deactivated and parking axes.....	341
7.7.4 Command value decoupling.....	341
Mode of operation.....	341
Displaying command value decoupling.....	342
7.7.5 Axis in parameterization mode.....	344
7.8 Single-axis modes.....	344
7.8.1 Overview.....	344
7.8.2 Operation modes.....	345
7.9 Operation modes for synchronous motions with electronic gear function.....	351
7.10 Boundary Conditions for an Axis Velocity Specification in the PLCopen State "Synchronized Motion".....	358
7.11 Axis commissioning.....	359
7.11.1 Overview.....	359
7.11.2 Safety functions.....	360
7.11.3 Enabling an axis.....	361
7.11.4 Velocity control.....	362
7.11.5 Positioning.....	362
7.11.6 Velocity synchronization.....	363
7.11.7 Position synchronization.....	363
7.11.8 Phase offset.....	364
7.11.9 FlexProfile.....	364

Table of Contents

	Page
7.11.10 Stop.....	365
7.12 Validating Motion configuration.....	365
7.12.1 Overview.....	365
7.12.2 Error messages.....	367
General information.....	367
Axis error messages.....	367
MLs-E0A0001: The axis name '<Name>' is not IEC-compliant.....	367
MLs-E0A0002: The axis name <Name> below the control is not unique.....	367
MLs-E0A0003: The axis name <Name> is identical to an IndraLogic keyword.....	367
MLs-E0A0005: The axis <Name> and a drive have the same name.....	368
MLs-E0A0010: The number of configured axes (<Digit1>) is higher than the maximum number of axes allowed (<Digit2>).....	368
MLs-E0A0012: The axis number <Digit> is invalid.....	368
MLs-E0A0013: The axis number <Digit> below the control is not unique.....	368
MLs-E0A0014: The axis name <Name>' is longer than permitted (maximum <Digit> char- acters)	368
The axis identification <Unique ID> below the control is not unique.....	368
MLs-E0A0020: The encoder axis <Name1>' is assigned to the real axis <Name2> This axis does not support any encoder axes.....	368
MLs-E0A0021: The encoder axis <Name> is not assigned to a real axis.....	369
MLs-E0A0023: The offline configuration of the drive <Name> does not match with the con- figured Sercos protocol.....	369
MLs-E0A0024: The firmware <Name1> of the axis <Name2> is invalid.....	369
MLs-E0A0030: For this axis, the control only supports the basic package "Closed Loop".....	369
MLs-E0A0100: A conversion error was determined causing missing internal axis data.....	369
Control error messages.....	369
MLs-E0C0001: The control name <Name> is not IEC-compliant.....	369
MLs-E0C0002: The control name <Name> in the project is not unique.....	370
MLs-E0C0003: The hardware "VEP" is not supported as control by IndraWorks.....	370
MLs-E0C0004: A control firmware of the version {VersionNumber} is not supported by this IndraWorks version.....	370
MLs-E0C0005: The device <Hardware> (<Version>) is not installed.....	370
MLs-E0C0006: The firmware release <Firmware> is not supported.....	370
MLs-E0C0010: The IP address <IP address> of the control is not unique in the project.....	370
MLs-E0C0014: The control name <Name> is longer than permitted (maximum <Digit> characters)	370
Function module error messages.....	371
MLs-E0F0010: The number of configured function modules (<Digit1>) is greater than the maximum permissible number of function modules (<Digit2>).....	371
MLs-E0F0011: More than the valid number (<Digit>) of function modules of type <Name> were configured.....	371
MLs-E0F0012: The slot <Digit> is not valid for the function module <Name>.....	371
MLs-E0F0013: The function module <Name> on the slot <Digit> has to be plugged in as last function module	371
MLs-E0F0020: The function module <Name> is not valid for the current control type.....	371
Sercos device error messages.....	371
MLs-E0S0010: The number of configured Sercos devices (<Digit1>) is greater than the maximum permissible number of Sercos devices (<Digit2>).....	371

Table of Contents

	Page
MLs-E0S0012: The Sercos address <Digit> is invalid.....	371
MLs-E0S0013: The Sercos address <Digit> below the control is not unique.....	372
MLs-E0S0014: The topology address <Digit> below the control is not unique.....	372
MLs-E0S0022: The I/O device <Name> does not match the configured Sercos protocol.....	372
Safety error messages.....	372
MLs-E0Y0010: The number of gateways "SLC-3-GS3S" (<Digit1>) is greater than the maximum permissible number (<Digit2>).....	372
MLs-E0Y0011: The number of "CIP Safety on Sercos" targets (<Digit1>) is greater than the maximum permissible number of "CIP Safety on Sercos" targets (<Digit2>).....	372
MLs-E0Y0012: The target number <Digit1> is outside the valid range (1..<Digit2>).....	373
MLs-E0Y0013: The target number <Digit> was specified multiple times.....	373
MLs-E0Y0014: "CIP Safety on Sercos" was configured for the axis, even though it is not supported by the axis.....	373
MLs-E0Y0015: The axis is assigned to the gateway "SLC-3-GS3S" which does not exist anymore. The gateway name was <Name>, the Sercos address was <Digit>.....	373
MLs-E0Y0100: "CIP Safety on Sercos with SafeLogic" was configured for the axis, but the required function module "Safety CPU" is missing.....	373
MLs-E0Y0101: "CIP Safety on Sercos with SafeLogic" was configured for the axis, but no Safety profile was selected.....	373
MLs-E0Y0102: The Safety profile <Name> selected for the axis is invalid.....	374
7.12.3 Information messages.....	374
General information.....	374
Non-object-related information messages.....	374
MLs-IOX0001: The following error <Identification> is temporarily only considered as warning.....	374
MLs-IOX0002: An internal error was detected during validation. Double-click to display details.....	374
7.12.4 Warning messages.....	374
General information.....	374
Axis warning messages.....	374
MLs-W0A0020: The configured function package of the axis <Name>" is not enabled in the corresponding drive.....	374
MLs-W0A0021: The configured axis name <Name1> does not match the axis name in the control <Name2>.....	375
MLs-W0A0022: An encoder is configured in the drive <Name>, even though the function package "Open Loop" is set.....	375
MLs-W0A0200: The configured axis is a axis different than the axis configured in the control (name: <Name>, axis number: <Digit>).....	375
MLs-W0A0201: The configured axis is not available in the control.....	375
MLs-W0A0202: An axis <Name> (axis number <Digit>) is in the control that is not in the project.....	375
Control warning messages.....	376
MLs-W0C0020: A higher device version (<Version>) is available for the control.....	376
MLs-W0C0021: The connected control has a lower firmware release (<Firmware>) as configured (<Firmware>).....	376
MLs-W0C0022: Not all IndraWorks online functions are available, as the configuration of the IndraWorks project differs from the control configuration.....	376
MLs-W0C0023: The hardware version <Value> of the connected control is syntactically incorrect.....	376

Table of Contents

	Page
MLs-W0C0200: The Motion project in the control (<Name>) does not match the Motion project in IndraWorks.....	377
Sercos device warning messages.....	377
MLs-W0S0013: Several devices with the Sercos address <Digit> are connected to the control.....	377
MLs-W0S0020: The device with the Sercos address <Digit> is disabled, even though it is connected to the control.....	377
MLs-W0S0021: The device with the Sercos address <Digit> is connected to the control, but it is not configured.....	377
MLs-W0S0023: The Motion cycle time (<Digit1>ms) is no multiple of the Sercos cycle time (<Digit2>ms).....	377
MLs-W0S0024: The device with the Sercos address <Digit> is configured in the control, but not connected.....	377
MLs-W0S0200: For the I/O device (Sercos address <Digit>), the device identification in the project (<Name1>) does not match the device identification (<Name2>) configured in the control.....	378
MLs-W0S0201: The configured I/O device (Sercos address <Digit>) is not configured in the control.....	378
MLs-W0S0202: In the control, an I/O device (device identification <Name>) is configured to the Sercos address <Digit>, but it is not in the project.....	378
MLs-W0S0203: For the I/O device (Sercos address <Digit>), the activation in the project does not match the activation in the control.....	378
Non-object-related warning messages.....	378
MLs-W0X0001: A function that is not released was enabled. This can end the warranty.....	378
Safety warning messages.....	379
W0Y0010: "CIP Safety on Sercos" was configured for the axis, but the actually connected drive does not support this functionality.....	379
W0Y0011: No "CIP Safety on Sercos" target was assigned to the gateway "SLC-3-GS3S"....	379
7.13 Electronic cams.....	379
7.13.1 General information.....	379
7.13.2 Cam solutions.....	380
7.13.3 Electronic cam.....	382
7.13.4 Use cases.....	382
7.13.5 Centralized and decentralized cam concept.....	383
7.13.6 FlexProfile – Basic functionalities	383
General information.....	383
Basic structure of synchronous operating states"FlexProfile".....	383
Profile configuration.....	384
Time master	385
Motion laws.....	385
Default motion laws.....	385
Extended motion laws.....	386
Flex step.....	387
Motion law boundary conditions.....	389
Switching and synchronizing FlexProfile	391
Selecting the coordinate system.....	391
Switching and synchronizing.....	392
Processing type.....	397

Table of Contents

	Page
Event control	398
Characteristics of the event control.....	398
Assigning MotionProfile and axis.....	401
FlexProfile – Configuration.....	402
Overview.....	402
Configuring via PLC function blocks.....	402
Configuring via IndraWorks	403
Configuring via CamBuilder.....	408
7.14 Touch probe.....	408
7.14.1 Introduction.....	408
7.14.2 Features.....	409
7.14.3 Functional description.....	409
7.14.4 Accuracy.....	411
7.14.5 Data acquisition of the touch probe.....	411
Activating measured value acquisition.....	411
Selecting measuring signal.....	412
Measured value acquisition mode.....	412
Saving measured values.....	413
Dead time compensation.....	413
Restarting or disabling measured value acquisition.....	414
Measured value status.....	415
7.14.6 Touch probe expectation window.....	415
Using the expectation window.....	415
Setting the expectation window.....	415
Marker detection.....	415
7.14.7 Overview on M-Parameters.....	416
M-0-0000, Dummy, M-parameter.....	416
M-0-0001, Logic touch probe number.....	416
M-0-0002, Touch probe name.....	416
M-0-0005, Input selection.....	416
M-0-0006, Signal selection.....	417
M-0-0007, Control word.....	417
M-0-0009, Osci: list of all M-parameters that can be oscilloscoped.....	419
M-0-0010, List of all M-parameters.....	419
M-0-0016, Project identification number.....	419
M-0-0021, Status word.....	419
M-0-0024, Command: Activation.....	420
M-0-0130, Measured value, positive edge.....	421
M-0-0131, Measured value, negative edge.....	421
M-0-0132, Difference value.....	421
M-0-0133, Time difference.....	422
M-0-0140, Time stamp, positive edge.....	422
M-0-0141, Time stamp, negative edge.....	423
M-0-0200, Dead-time compensation, positive edge.....	423
M-0-0201, Dead-time compensation, negative edge.....	423
M-0-0202, Starting position of the expectation window, positive edge.....	423

Table of Contents

	Page
M-0-0203, End position of the expectation window, positive edge.....	423
M-0-0204, Starting position of the expectation window, negative edge.....	424
M-0-0205, End position of the expectation window, negative edge.....	424
M-0-0206, Maximum number of marker failures.....	424
M-0-0224, Number of marker failures, positive edge.....	424
M-0-0225, Number of marker failures, negative edge.....	425
M-0-0401, Input status.....	425
M-0-0405, Enabling.....	425
M-0-0409, Counter of measured values, positive edge.....	426
M-0-0410, Counter of measured values, negative edge.....	426
M-0-0411, Counter of difference values.....	426
M-0-1000, Command: Load basic touch probe parameters.....	427
M-0-2100, List of all backup operation M-parameters.....	427
7.14.8 Function blocks for touch probes.....	427
7.15 Programmable limit switch.....	428
8 CamBuilder.....	431
8.1 CamBuilder functionality.....	431
9 Oscilloscope.....	433
9.1 Oscilloscope functionality.....	433
10 Error cases.....	435
10.1 Introduction and overview.....	435
10.2 Error categories.....	435
10.2.1 Exception in IEC application.....	435
10.2.2 SysError.....	437
10.2.3 Spontaneous control reboot.....	438
10.3 Procedure in case of SysError.....	438
10.4 BOOTSTOP Menu.....	439
10.5 Core Dump.....	441
11 User-defined languages for parameter names, units, diagnostics and menu texts... 443	443
11.1 General information.....	443
11.2 File management.....	443
11.3 File structure.....	443
11.4 Function.....	444
11.5 Node names for units.....	446
11.6 Node names for menu texts (display texts).....	447
12 Remote access via modem.....	449
12.1 General information.....	449
12.1.1 Definition of terms and introduction.....	449
12.1.2 Connection via internet.....	449

Table of Contents

	Page
12.1.3 Point-to-point connection.....	449
12.1.4 Accessibility to the control.....	450
12.2 Dial-up connection between service PC and client PC.....	450
12.2.1 General information.....	450
12.2.2 Required equipment.....	450
12.2.3 Installation and configuration.....	451
12.2.4 Possible remote maintenance tasks.....	451
12.3 Dial-up connection between service PC and MoRoS.....	451
12.3.1 General information.....	451
12.3.2 Required equipment.....	452
12.3.3 Remote maintenance tasks.....	452
12.3.4 Installation and configuration.....	452
General information.....	452
Initial MoRoS configuration.....	452
Network configuration.....	454
MoRoS network configuration.....	454
Network configuration of the controls.....	456
Configuration service PC.....	456
13 Service and support.....	461
Index.....	463

1 About this documentation

1.1 Validity of the documentation

Overview on target groups and product phases

In the following illustration, the framed activities, product phases and target groups refer to the present documentation.

Example: In the product phase "Mounting (assembly/installation)", the "mechanic/electrician" can execute the activity "unpack, mount and install" using this documentation.

Required qualification: Individual who is able to assess the tasks assigned and identify possible safety risks owing to qualification in the subject, knowledge and experience. The individual should also be familiar with the standards and regulations.

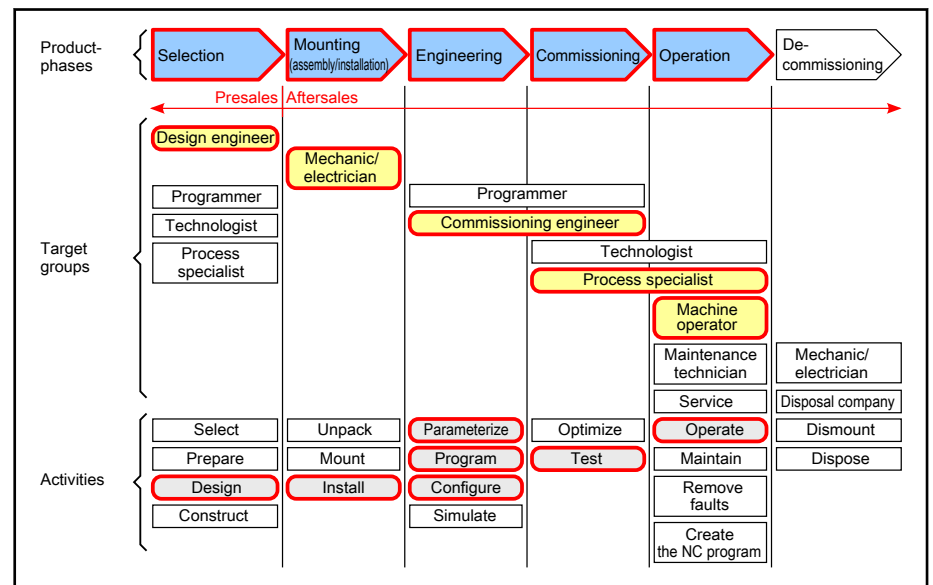


Fig. 1-1: Assigning this documentation to the target groups, product phases and target group activities

Purpose This documentation describes the following IndraLogic XLC elements 13VRS:

- Wizards
- Context menus
- Dialogs
- Frequently required sequences
- Device configuration
- Function modules
- Functionalities

1.2 Documentation structure

The [chapter 2 "System overview" on page 27](#) contains a system overview of the IndraLogic XLC.

The overview is structured as follows:

- hardware
- Firmware
- Software

About this documentation

The [chapter 3 "Commissioning and operation" on page 33](#) and [chapter 4 "Control" on page 81](#) contain descriptions on the commissioning and operation as well as on the IndraWorks user interface.

The [chapter 5 "PLC programming" on page 169](#) provides an overview on the PLC programming.

The [chapter 6 "Axes" on page 207](#) describes the functionalities of the different axis types.

The [chapter 7 "Motion functionality" on page 323](#) describes the Motion functionality.

The [chapter 8 "CamBuilder" on page 431](#) and [chapter 9 "Oscilloscope" on page 433](#) contain document specifications describing the respective function.

The [chapter 10 "Error cases" on page 435](#) describes how to react if an error is reported.

The [chapter 11 "User-defined languages for parameter names, units, diagnostics and menu texts" on page 443](#) describes the use of user-defined languages.

The [chapter 12 "Remote access via modem" on page 449](#) describes the remote access to an IndraLogic XLC.

For information on the customer service helpdesk, refer to [chapter 13 "Service and support" on page 461](#).

1.3 Using safety instructions

1.3.1 Safety instructions – Structure

The safety instructions are structured as follows:

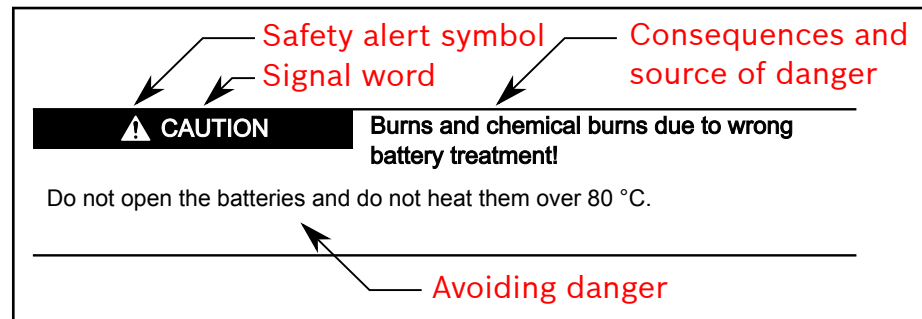


Fig. 1-2: Safety instructions – structure

1.3.2 Explaining signal words and safety alert symbol

The safety instructions in this documentation contain specific signal words (danger, warning, caution, notice) and, if necessary, a safety alert symbol (according to ANSI Z535.6-2006).

The signal word is used to draw attention to the safety instruction and also provides information on the severity of the hazard.

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

[About this documentation](#)

** WARNING**

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

** CAUTION**

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

NOTICE

In the event of non-compliance with this safety instruction, material damage can occur.

About this documentation

1.3.3 Symbols used

Pointers are displayed as follows:



This is a note.

Tips are displayed as follows:



This is a tip.

1.4 Required and supplementing documentations

Documentation titles with type codes and parts numbers

IndraWorks		MLC	XLC
/36/	Rexroth IndraWorks 13VRS Software Installation DOK-IWORKS-SOFTINS*V13-CORS-EN-P, R911336880 This documentation describes the IndraWorks installation.	X	X
/5/	Rexroth IndraWorks 13VRS Engineering DOK-IWORKS-ENGINEE*V13-APRS-EN-P, R911336870 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 13VRS Functional Description DOK-MLC***-FUNC****V13-APRS-EN-P, R911336295 This documentation describes wizards, context menus, dialogs, control commissioning, device configuration and functionalities of the IndraMotion MLC.	X	
/20/	Rexroth IndraLogic XLC 13VRS Functional Description DOK-XLC***-FUNC****V13-APRS-EN-P, R911336352 This documentation describes wizards, context menus, dialogs, control commissioning, device configuration and functionalities of the IndraLogic XLC.		X
/7/	Rexroth IndraWorks 13VRS CamBuilder DOK-IWORKS-CAMBUIL*V13-APRS-EN-P, R911336291 This documentation describes the basic principles and operation of the CamBuilder, the cam editing tool.	X	X
/37/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Automation Interface DOK-XLCMLC-AUT*INT*V13-APRS-EN-P, R911336356 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-APRS-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

IndraWorks		MLC	XLC
/29/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Project Conversion DOK-XLCMLC-PROCONV*V13-APRS-EN-P, R911336366 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 13VRS Commissioning DOK-MLC***-STARTUP*V13-CORS-EN-P, R911336308 This documentation describes the steps to commission and service the IndraMotion MLC system. It includes checklists for frequent tasks and a detailed description of the steps.	X	

Tab. 1-1: XCL/MLC documentation overview

About this documentation

Motion		MLC	XLC
/23/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Libraries DOK-XLCMLC-FUNLIB**V13-LIRS-EN-P, R911336360 This documentation describes the function blocks, functions and data types of the RIL_Common-Types, 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 13VRS Generic Application Template DOK-XLCMLC-TF*GAT**V13-APRS-EN-P, R911336370 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 13VRS RCL Programming Instruction DOK-MLC***-RCL*PRO*V13-APRS-EN-P, R911336297 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	
/32/	Rexroth IndraMotion MLC 13VRS Robot Control V2 DOK-MLC***-ROCO***V13-RERS-EN-P, R911341035 The documentation provides information about the Robot Control V2. The focus of this documentation is on PLCopen programming. The program structure, variables, functions and motion commands as well as the required system parameters are described. Furthermore, the documentation contains the description of the ML_Robot library.	X	
/21/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Parameters DOK-XLCMLC-PARAM***V13-RERS-EN-P, R911336364 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**-FKRS-EN-P, R911328670 This documentation describes all functional properties of the IndraDrive firmware in the variants MPH-07, MPB-07, MPD-07 and MPC-07.		
/11/	Rexroth IndraDrive MPx-17 Functions DOK-INDRV*-MP*-17VRS**-APRS-EN-P, R911331236 This documentation describes all functional properties of the IndraDrive firmware in the variants MPB-17, MPM-17, MPC-17 and MPE-17.		

Tab. 1-2: XCL/MLC documentation overview

About this documentation

Field Buses		MLC	XLC
/4/	Rexroth IndraWorks 13VRS Field Buses DOK-IWORKS-FB*****V13-APRS-EN-P, R911336872 This documentation describes the supported field buses and their diagnostic function blocks.	X	X
/4a/	Rexroth IndraWorks 13VRS Field Buses Libraries DOK-IWORKS-FB*LIB**V13-LIRS-EN-P, R911337857 This manual describes the field bus libraries: RIL_ProfibusDP_02, RIL_ProfibusDP Slave, RIL_Profine-tIO, RIL_ProfinetIODevice, RIL_EtherNetIPAdapter, RIL_MappingList, RIL_SERCOSIII, RIL_Inline including their diagnostics and error reactions of the function blocks.	X	X
/63/	SERCOS System Manual for I/O Devices DOK-CONTRL-ILS3*****-APRS-EN-P, R911333512 This documentation describes the configuration, parameterization, commissioning and diagnostics of I/O devices with a SERCOS interface.	X	X

Tab. 1-3: XCL/MLC documentation overview

HMI		MLC	XLC
/8/	Rexroth IndraWorks 13VRS HMI DOK-IWORKS-HMI*****V13-APRS-EN-P, R911336874 This documentation describes the functions, configuration and operation of the user interfaces IndraWorks HMI Engineering and IndraWorks HMI Operation.	X	X
/6/	Rexroth IndraWorks 13VRS WinStudio DOK-IWORKS-WINSTUD*V13-APRS-EN-P, R911336882 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 13VRS HMI Connection DOK-XLCMLC-HMI*****V13-APRS-EN-P, R911336362 This documentation describes the visualization systems supported by the IndraLogic XLC and IndraMotion MLC and their connection.	X	X

Tab. 1-4: XCL/MLC documentation overview

PLC		MLC	XLC
/3/	Rexroth IndraWorks 13VRS IndraLogic 2G Programming Instruction DOK-IWORKS-IL2GPRO*V13-APRS-EN-P, R911336876 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 13VRS Basic Libraries IndraLogic 2G DOK-IL*2G*-BASLIB**V13-LIRS-EN-P, R911336285 This documentation describes the system-comprehensive PLC libraries.	X	X

Tab. 1-5: XCL/MLC documentation overview

About this documentation

Technology		MLC	XLC
/30/	Rexroth IndraMotion MLC 13VRS Technology Libraries DOK-MLC***-TF*LIB**V13-LIRS-EN-P, R911336324 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 13VRS RegisterControl (Library) DOK-MLC***-REGI*CO*V13-LIRS-EN-P, R911336306 This documentation describes the inputs and outputs of the individual function blocks and provides notes on their use.	X	
/62/	Rexroth IndraMotion MLC 13VRS RegisterControl (Application Manual) DOK-MLC***-REGI*CO*V13-APRS-EN-P, R911336304 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 13VRS Winder Function Application DOK-MLC***-TF*WIND*V13-APRS-EN-P, R911336326 This application-related system documentation describes the application of the winder technology functions.	X	

Tab. 1-6: XCL/MLC documentation overview

Hydraulics		MLC	XLC
/40/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Sequential Programming DOK-XLCMLC-SEQPROG*V13-LIRS-EN-P, R911336368 This documentation includes the library for the sequential programming.	X	X
/41/	Rexroth IndraMotion MLC 13VRS MH_TechHydrBase DOK-MLC***-TF*HBASEV13-LIRS-EN-P, R911336312 No purpose defined.	X	
/42/	Rexroth IndraMotion MLC 13VRS MH_TechHydrMotion DOK-MLC***-TF*HMOT*V13-LIRS-EN-P, R911336316 No purpose defined.	X	
/43/	Rexroth IndraMotion MLC 13VRS MH_HydrControl DOK-MLC***-TFHCLIB*V13-LIRS-EN-P, R911336328 No purpose defined.	X	
/44/	Rexroth IndraMotion MLC 13VRS MH_SynchControl DOK-MLC***-TFHSLIB*V13-LIRS-EN-P, R911336330 No purpose defined.	X	

Tab. 1-7: XCL/MLC documentation overview

About this documentation

Hardware		MLC	XLC
/1/	Rexroth IndraControl L45/L65/L85 Control DOK-CONTRL-ICL45L65L85-PRRS-EN-P, R911332116 This documentation describes the IndraControl L45/L65/L85 controls.	X	X
/2/	Rexroth IndraControl L25 DOK-CONTRL-IC*L25****-PRRS-EN-P, R911328474 This documentation describes the IndraControl L25 controls.	X	X
/24/	Rexroth IndraControl Lxx 13VRS Function Modules DOK-CONTRL-FM*LXX**V13-APRS-EN-P, R911336283 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**-PARS-EN-P, R911297317 This documentation describes all parameters implemented in the firmware for drive controllers of the IndraDrive family. It supports the parameterization of the drive controllers.		
/13/	Rexroth IndraDrive MPx-02 to MPx-07 and HMV DOK-INDRV*-GEN-**VRS**-WARS-EN-P, R911297319 This documentation describes all diagnostics implemented in the following firmwares: • Drive controller firmwares from MPx-02 to MPx-08 and • firmwares of the supply devices of the type "HMV". It supports the operating crew as well as the programmer at troubleshooting.		
/35/	Rexroth IndraDrive Drive Controller Control Sections CSB01, CSH01, CDB01 DOK-INDRV*-CSH*****-PRRS-EN-P, R911295012 This documentation is a project planning manual to select and use the control sections CSB01, CDB01 and CSH01 for drive controllers of the product families Rexroth IndraDrive M and Rexroth IndraDrive C.		

Tab. 1-8: XCL/MLC documentation overview

Diagnostics and service		MLC	XLC
/26/	Rexroth IndraWorks 13VRS IndraMotion Service Tool DOK-IWORKS-IMST****V13-RERS-EN-P, R911337707 This documentation describes the IndraMotion Service Tool (IMST). It is a web-based diagnostic tool to access the control systems IndraMotion MLC L25, L45 or 65 as well as the MLP via an Ethernet high-speed connection. The IMST allows OEMs, end users and service engineers to access and remotely diagnose a system. A PC with Internet Explorer version 6 or 7 is required.	X	X
/22/	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Diagnostics DOK-XLCMLC-DIAG****V13-RERS-EN-P, R911336358 This documentation includes all control diagnostics implemented in the control systems IndraLogic XLC and IndraMotion MLC.	X	X

Tab. 1-9: XCL/MLC documentation overview

About this documentation

System Overview		MLC	XLC
/48/	Rexroth IndraMotion for Printing 13VRS System Overview DOK-IM*PR*-SYSTEM**V13-PRRS-EN-P, R911336289 This documentation describes the product IndraMotion for Printing. It introduces the control systems, drive systems and I/O systems as well as the commissioning and programming.	X	
/48/	Rexroth IndraMotion for Packaging 13VRS System Overview DOK-IM*PA*-SYSTEM**V13-PRRS-EN-P, R911336287 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 13VRS System Overview DOK-MLC***-SYSTEM**V13-PRRS-EN-P, R911336310 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 13VRS System Overview DOK-XLC***-SYSTEM**V13-PRRS-EN-P, R911336354 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

Tab. 1-10: XCL/MLC documentation overview

First Steps		MLC	XLC
/25/	Rexroth IndraMotion MLC 13VRS, First Steps DOK-MLC***-F*STEP**V13-CORS-EN-P, R911336293 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 13VRS, First Steps DOK-XLC***-F*STEP**V13-CORS-EN-P, R911336350 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

Tab. 1-11: XCL/MLC documentation overview

1.5 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. The Imc interface includes control signals and parameters for the diagnostics and administration of the control
IndraLogic XLC	Compact PLC system with integrated Motion functionality
IndraWorks Engineering Framework	Project planning and commissioning tool of Bosch Rexroth
IndraDrive	Drive controller
Wizard	Wizard guiding the user through several dialogs for an ergonomic data input

About this documentation

Term	Explanation
Function block	Provides application-specific functionalities
IEC	International Electrotechnical Commission
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
Master communication	Master communication

Tab. 1-12: Names and abbreviations used

1.6 Customer feedback

Customer requests, comments or suggestions for improvement are of great importance to us. Please email your feedback on the documentations to Feedback.Documentation@boschrexroth.de. Directly insert comments in the electronic PDF document and send the PDF file to Bosch Rexroth.

2 System overview

2.1 Hardware

Control hardware The IndraLogic XLC system consists of the control hardware of the IndraControl series of and of a corresponding firmware.

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

The control systems contain the following components:

IndraLogic XLC L25:

- The main features of an embedded control hardware IndraControl L25 in the lower performance segment are:

- Ethernet 10/100 MBit
- Firmware on a 1 GB Compact Flash Card
- Connection for Inline I/O terminals (possible I/O extensions)

One of the following interfaces can be selected additionally:

- Onboard Sercos III interface (Ethernet)
- Onboard Profibus DP master/slave, Profinet/Ethernet IP adapter

IndraLogic XLC 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 or 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)

IndraLogic XLC 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 or 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)

The following axis types are supported:

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

System overview

Technical data Comparison of technical data of the IndraLogic XLC hardware:

IndraControl	CML25.1		CML45.1		CML65.1	
	PN	3N	NP	3P	NP	3P
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					
Removable storage (Compact Flash card)	1 GB					
Ready contact	⑤		⑤		⑤	
Interfaces						
Ethernet 10/100 MBit	⑤		⑤		⑤	
Sercos III	–	⑤	–	⑤	–	⑤
Profinet-I/O	○	–	⑤	⑤	⑤	⑤
Profibus DP (Master/Slave)	⑤	–	⑤	⑤	⑤	⑤
Ethernet IP						
Master (scanner)	–	–	–	–	–	–
Slave (adapter)	⑤	–	⑤	⑤	⑤	⑤
USB	–		–		–	
Inputs/outputs	–		8DI+8DO		8DI+8DO	
I/O extension (Inline terminals)	Up to 64 bytes					
Function modules	⑤ Up to two modules can be connected		⑤ Up to four modules can be connected			
Other						
Max. number of axes	16		32		64	
Operating system	VxWorks					
Display	Diagnostics Display					

- ⑤ Default function
○ Option
– Not available

Tab. 2-1: Technical data of the IndraLogic XLC

Communication interfaces in comparison:

		Indra-Control L25	Indra-Control L45	Indra-Control L65
Interfaces				
Sercos III	Real-time Ethernet bus	○	○	○
	Max. number of Sercos devices (drives + I/O)	48	64	99
	Max. number of drives	16	32	64
	Max. number of I/Os in case of max. number of axes	32	32	35
Sercos II (via function module) (only IndraMotion MLC)	Real-time Motion bus	–	○	○
Profibus DP	Master	–	⑤	⑤
	Slave	–	⑤	⑤
Profinet I/O	Controller (master)	○	⑤	⑤
	Device (slave)	○	⑤	⑤
Standard EtherNet	EtherNet TCP/UDP/IP	⑤	⑤	⑤
		X7E3	X7E5	X7E5
Ethernet IP	Scanner (master)	–	–	–
	Adapter (slave)	○	⑤	⑤

⑤ Default function
○ Option
– Not available



For further information on the IndraLogic XLC system, refer to "Rexroth IndraLogic XLC13VRS System Overview" (see [chapter 1.4 "Required and supplementing documentations"](#) on page 18).

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:

- 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
- Real-time Ethernet (CFL01.1-TP); Real-time Ethernet und Profibus DP master or slave functionality

System overview



For the configuration of the function modules, refer to [chapter 3.8 "Creating a function module" on page 52](#).

For more information on function modules, refer to "Rexroth IndraControl Lxx 13VRS Function Modules" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

I/O periphery

The following bus systems are provided for the I/O extension of the IndraLogic XLC:

- Inline, Profibus*, Sercos III*, Profinet*

* - IndraLogic XLC L25 either has Sercos III or Profibus and Profibus.

The configuration of this hardware is described in

- Inline I/O - [chapter 3.9.2 "Creating Inline I/Os" on page 56](#)
- Profibus I/O - [chapter 3.9.3 "Creating Profibus I/Os" on page 63](#)
- Sercos III I/O - [chapter 3.9.5 "Creating Sercos III I/Os" on page 64](#)
- Profinet I/O - [chapter 3.9.4 "Creating a Profinet I/O Controller" on page 63](#)

Visualization devices

A variety of HMI devices are available for visualization.

The IndraLogic XLC L25/L45/L65 support the following devices:

- IndraControl VCP - protocol: BRC symbolic via
 - Ethernet
 - Profibus
- IndraControl VEP - Ethernet via WinStudio to configure the 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).

2.2 Firmware

CF card

To operate the control IndraLogic XLC, a corresponding Compact Flash card with the system firmware (Runtime) IndraLogic XLC 13VRS 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 overview

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 5.2.5 "Loading a PLC program to the control" on page 171)

Tab. 2-2: 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

Tab. 2-3: Memory distribution/Memory requirement of the IndraLogic XLC

Update options

Within one version, the system firmware can be updated via a download function in IndraWorks (refer to [chapter 4.4.9 "Firmware management"](#) on page 134) (new release).

2.3 Software

IndraWorks

The "IndraWorks Engineering" software is used to commission and configure the IndraLogic XLC. The software 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 screens 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

System overview

- The AxisInterface facilitates the access to axes. The AxisInterface 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^{compact} is available for the target system IndraLogic XLC:

- Technology-based function blocks extend the basic functionality of the IndraLogic XLC 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)

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



For further documentations on the IndraLogic XLC system, refer to [chapter 1.4 "Required and supplementing documentations"](#) on [page 18](#).

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

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

For the OSCAT library, refer to www.oscat.com.



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.

3 Commissioning and operation

3.1 General information

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

For information on the PLC programming with IndraLogic, refer to "Rexroth IndraWorks 13VRS IndraLogic 2G Programming Instruction" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

3.2 IndraLogic XLC – Basic Sequences, General Information

Working with IndraWorks and an IndraLogic XLC requires a functioning hardware with compatible firmware components.

Also refer to "Rexroth IndraMotion XLC 13VRS, First Steps" in [chapter 1.4 "Required and supplementing documentations" on page 18](#).

Thus, the following steps are recommended:

- Hardware commissioning
 - [chapter 3.3.1 "Connecting the control" on page 33](#)
 - [chapter 3.3.3 "Deleting control memory" on page 37](#) if required
 - [chapter 3.3.4 "Setting the IP address" on page 38](#)
 - [chapter 3.3.5 "Adding the Sercos bus" on page 41](#)

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

First, the offline activities:

- [chapter 3.4 "Installing and starting IndraWorks" on page 42](#)
- [chapter 3.6 "Creating a project" on page 46](#)
- [chapter 3.7 "Creating a control" on page 48](#) and if required
 - [chapter 3.9.1 "Creating Onboard I/Os" on page 54](#)
 - [chapter 3.9.2 "Creating Inline I/Os" on page 56](#)
 - [chapter 3.9.3 "Creating Profibus I/Os" on page 63](#)
 - [chapter 3.9.5 "Creating Sercos III I/Os" on page 64](#)
 - [chapter 3.9.4 "Creating a Profinet I/O Controller" on page 63](#)
 - [chapter 3.8 "Creating a function module" on page 52](#)

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.

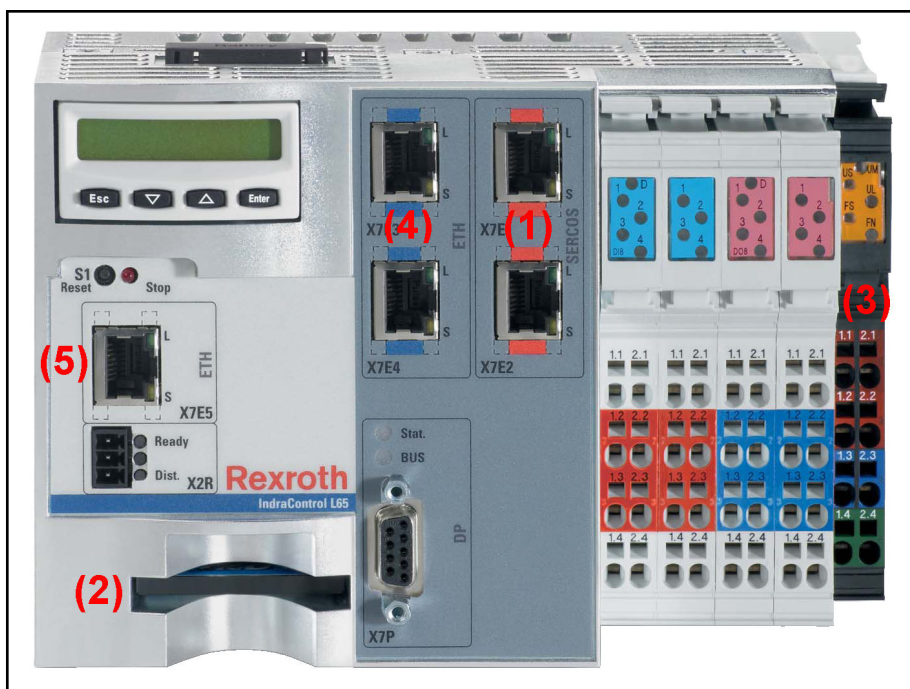
3.3 IndraLogic XLC L25/L45/L65 – Commissioning

3.3.1 Connecting the control

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

Commissioning and operation

Connections



- ① Sercos III ports
- ② CF card
- ③ Power supply
- ④ Connections for Ethernet IP, Profinet, Standard Ethernet (only IndraLogic XLC L45/L65)
- ⑤ Standard Ethernet and Ready contact connection

Fig. 3-1: Example: CML65.1 hardware for IndraLogic XLC applications



Fig. 3-2: Example: CML25.1 hardware for IndraLogic XLC applications

Ready contact behavior

The Ready contact is opened by the software under the following conditions:

- Switching to CP0
- Fatal system error (F9)

The Ready contact is closed under the following conditions:

- 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).

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 ports X7E1 and X7E2 ① with an Ethernet cable (elimination of possible errors in the Sercos bus)
Omit this step for controls without Sercos interface
- Insert the CF card with the IndraLogic XLC firmware into ②
- Connect network cable with connection to the workstation at ⑤
- Provide supply voltage at ③

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

3.3.2 Operating the control display

The control display indicates the control status, the network setting configuration and the single command triggering.

The control status is displayed after startup. If there is no error, the PLC status (STOP or RUN) and Sercos bus status (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 go to 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 with <Esc> and the status display becomes active again.
Arrow down	Shows the next entry in the menu level.
Arrow up	Shows the previous entry of the menu level.

Commissioning and operation

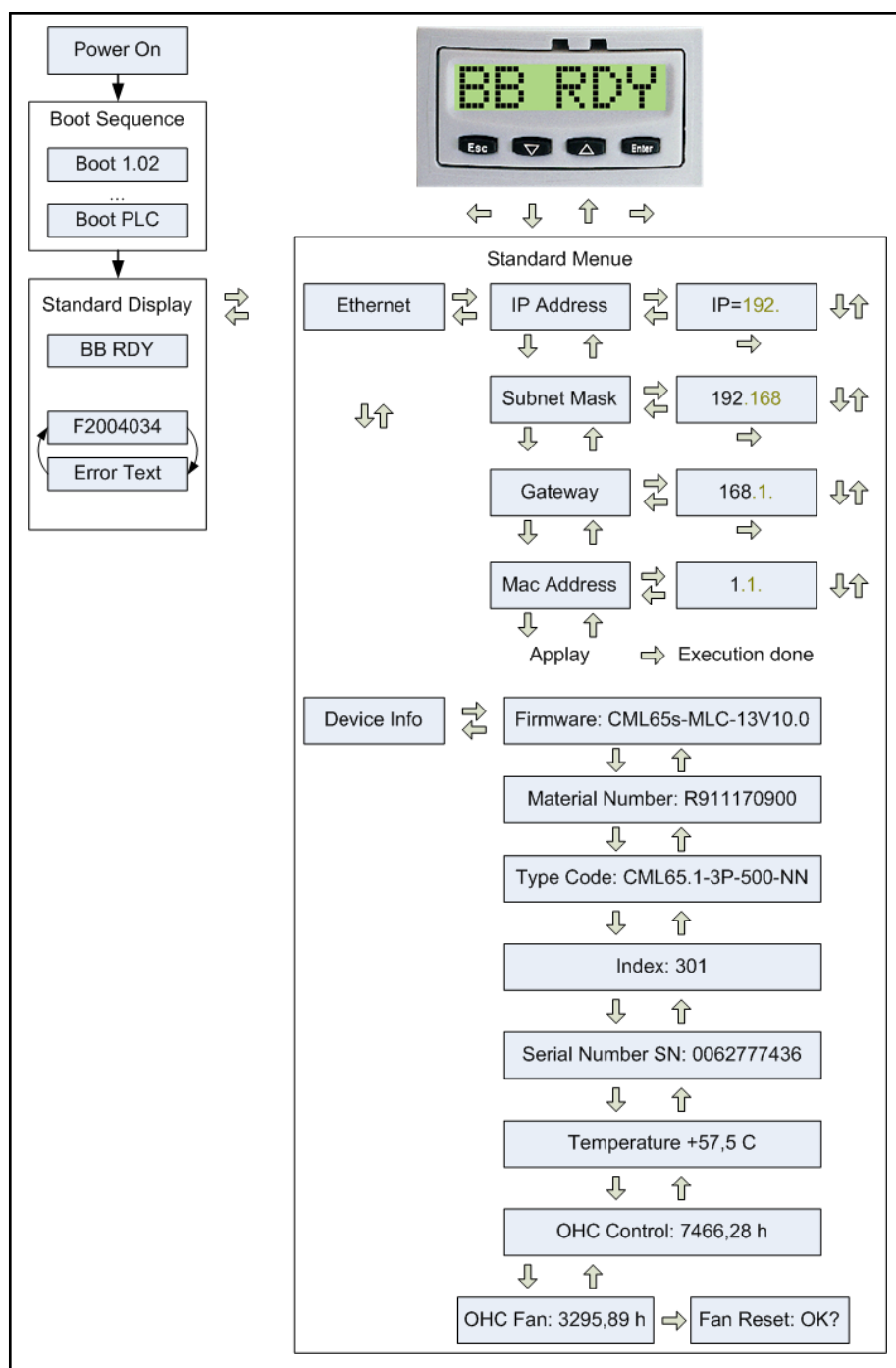


Fig. 3-3: Diagnostics and operation of the IndraLogic XLC display

Press <Esc> and <Enter> simultaneously on 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.

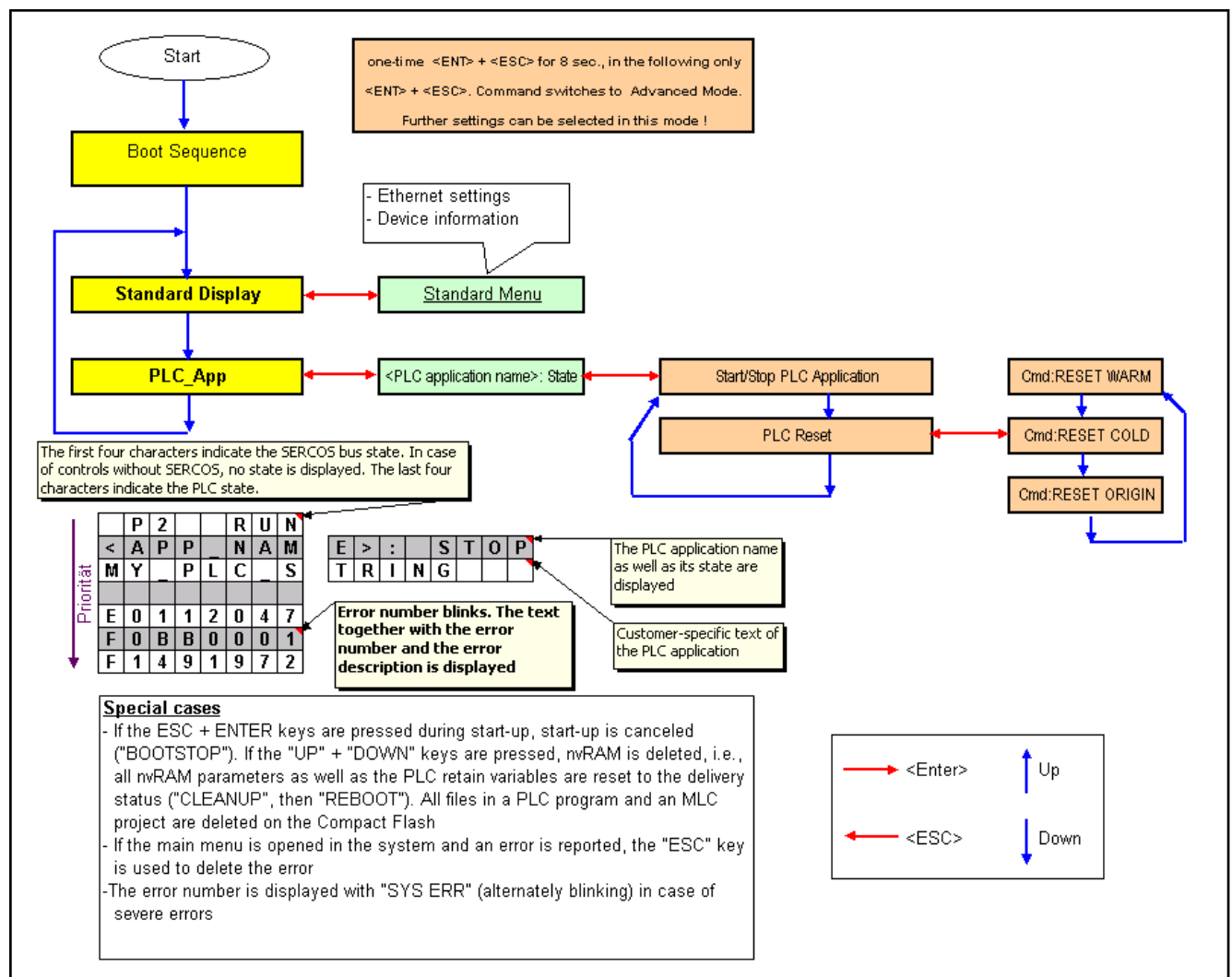


Fig. 3-4: Menu in advanced (extended) mode

3.3.3 Deleting control memory

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.



Fig. 3-5: Rebooting the control

Commissioning and operation

1. After switching on the control, press both the outer buttons <ESC> and <Enter> below the display and keep them pressed until **BOOTSTOP** is displayed.
2. Press both the inner buttons.
CLEANUP? is displayed and the buttons can be released.
3. Press <Enter>.
CLEANUP, SUCCESS, REBOOT, StartUp are subsequently shown in the display
The control runs up to **BB RDP**, the memory is cleared, the Motion components are ready for operation and the PLC components in ready for operation. No valid PLC application is loaded.

3.3.4 Setting the IP address

IP address – Initial setting Before the control can be connected to the programming device via Ethernet (e.g. notebook), provide the control with an IP address valid for the network.

State upon delivery The following addresses are set on the control upon delivery:

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

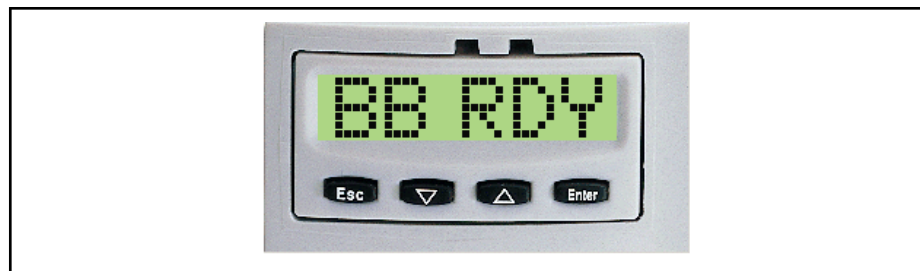
Tab. 3-1: Initial values

Set the IP address and subnet mask of a control via the four keys of the control display (see fig. 3-6 "Display with four operating keys (standard display)" on page 38). 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 the **BB RDY** state, retrieve and set the IP address and the subnet address and determine the current firmware release.



- 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. 3-6: Display with four operating keys (standard display)

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

Commissioning and operation

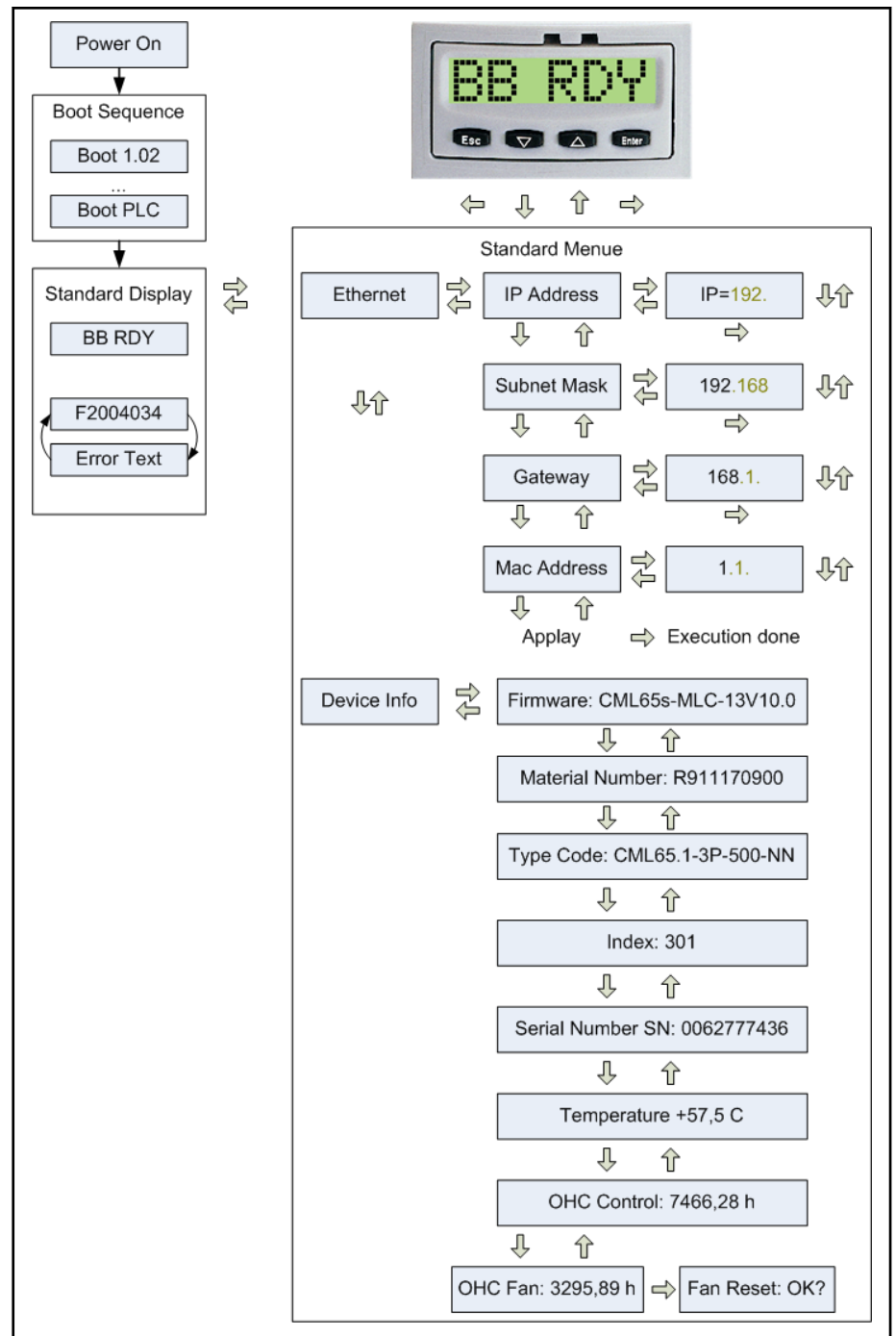


Fig. 3-7: Standard menu overview to operate display and keyboard

1. Move with <Enter> from the standard display **BB RDY** to the **ETHERNET** display.
2. Press <Enter> again to go to the **IP address**.
3. Press <ENTER> again to change the preset address (change the numerical value with <DOWN> or <UP>; continue 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.

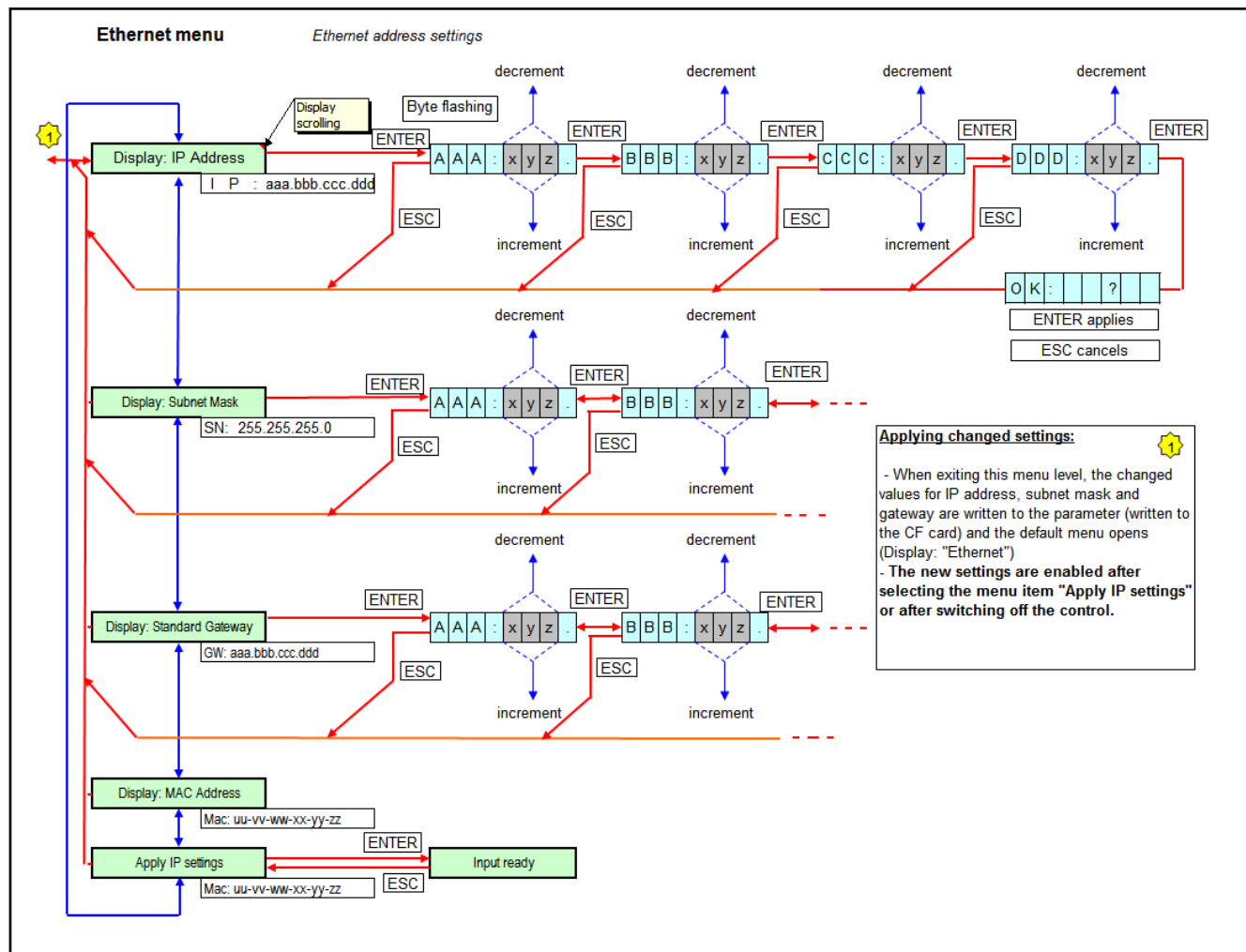


Fig. 3-8: Ethernet menu



An address change is applied only after the "Apply IP Settings" menu item was enabled or the voltage was switched off and back 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 supposed to correspond to the requirements for future use.

Behavior in case of IP address conflict

The same IP address is not to be provided to two devices in one network. If this is the case, the control responds as follows:

1. If, when booting the control or after reconfiguring the IP address, it is determined that the same IP address is already used, the control stops the communication on the engineering port. A corresponding error message is displayed.

Commissioning and operation

2. If the control detects at runtime that its own IP address is used by another device, it waits for 10 seconds whether the second device stops the communication. A corresponding warning is shown on the display of the control during this period. If the communication is not stopped after 10 seconds, the control stops its communication via the engineering port. A corresponding error message is displayed.

If the control stops the communication via the engineering port, it is not reachable via Ethernet until the next restart or until a new IP address is specified via the display. In this case, there is no communication between IndraWorks and the control.

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 has to be in ASCII format and the file name has to be "Config.txt".

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 desired values in that template.

Do not change the spelling of the keywords! The syntax of the values has to comply with the conventions.

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

Network parameters upon delivery

```
[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) may not contain leading zeroes.

Cross-over connection

A cross-over cable provides a special coupling between an IndraLogic XLC and a PC. In this case, adjust the IP address of the PC and its subnet mask to the IndraLogic XLC 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 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	Automatically	255.255.255.128
Standard gateway	Neutral	Obtain	Not set

Tab. 3-2: Setting when connecting via cross-over

3.3.5 Adding the Sercos bus

This section is only relevant for controls with onboard Sercos III port.

The bridge suggested in section "Connections" on page 34 was used instead of the Sercos bus and is now replaced by the bus.

The following prerequisites are required:

Commissioning and operation

- 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 used once
- the wiring is correct

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

After switching the control off and back on, **BB Stop** should be reached for the control and without power on **bb** for the drives.

3.3.6 Replacing hardware

Target Replacing the IndraLogic XLC hardware. Transferring project data and firmware from one control hardware to another using a CF card.

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 the new control. Apply any present function modules and the Inline bus.
2. Insert the CF card of the old control into the new control.
3. Switch on the control.

After switch-on, the control hardware detects the new CF 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".

3.4 Installing and starting IndraWorks

Installing "IndraWorks Engineering"

Before the initial use, install the IndraWorks program system of the IndraWorks Engineering Suite.

For more information on the installation, refer to the manual "Rexroth IndraWorks 13VRS Software Installation" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

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 when installing from the network drive, there is an installation manual with information on possible service packages.

IndraWorks – Startup

After , the initial start of IndraWorks Engineering (or after **View ► Startup**), the following screen is displayed:

Commissioning and operation

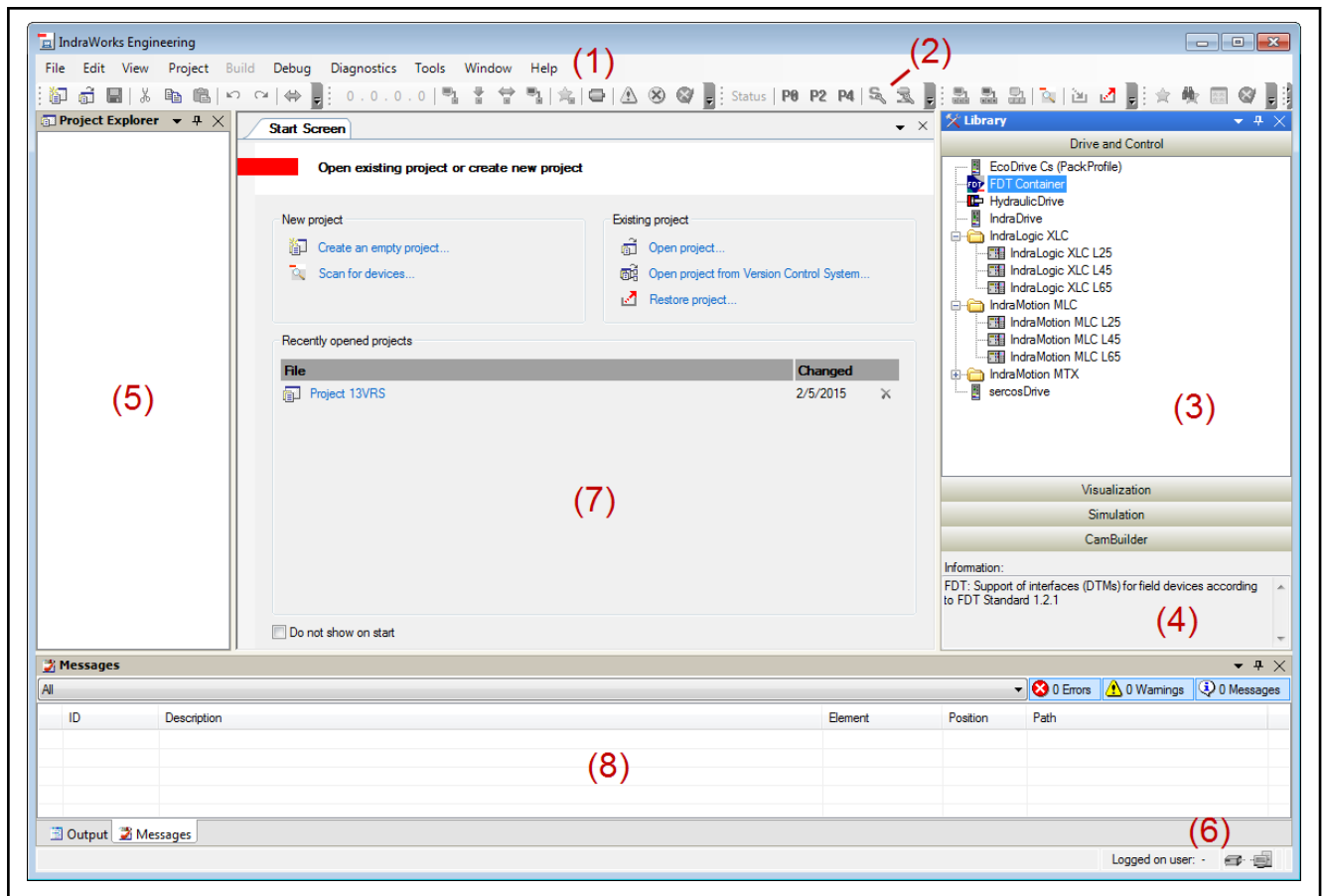


Fig. 3-9: Engineering Suite – Start screen after initial start

The figure illustrates the menu ① of the Engineering Suite. The menu items can be supplemented or changed 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 Engineering Suite. Its elements are stored in the following folders:

- **Drive and Control:**
 - EcoDrive Cs (PackProfile)
 - FDT Container
 - HydraulicDrive
 - IndraDrive
 - IndraLogic XLC
 - IndraLogic XLC L25
 - IndraLogic XLC L45
 - IndraLogic XLC L65
 - IndraMotion MLC
 - IndraMotion MLC L25
 - IndraMotion MLC L45
 - IndraMotion MLC L65

Commissioning and operation

- SercosDrive
 - etc.
- **Visualization:**
 - OEM data
 - VCP/VCH
 - VEH/VEP
 - VPx/VSx
- **Simulation**
- **CamBuilder:**
 - Cam Pool
 - Cam
 - Preview
 - Preview of the superimposed cams (X/Y)

Right-click on the element for the help on each element.

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

- **Periphery**
- **Function modules**
- **PLC objects**

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

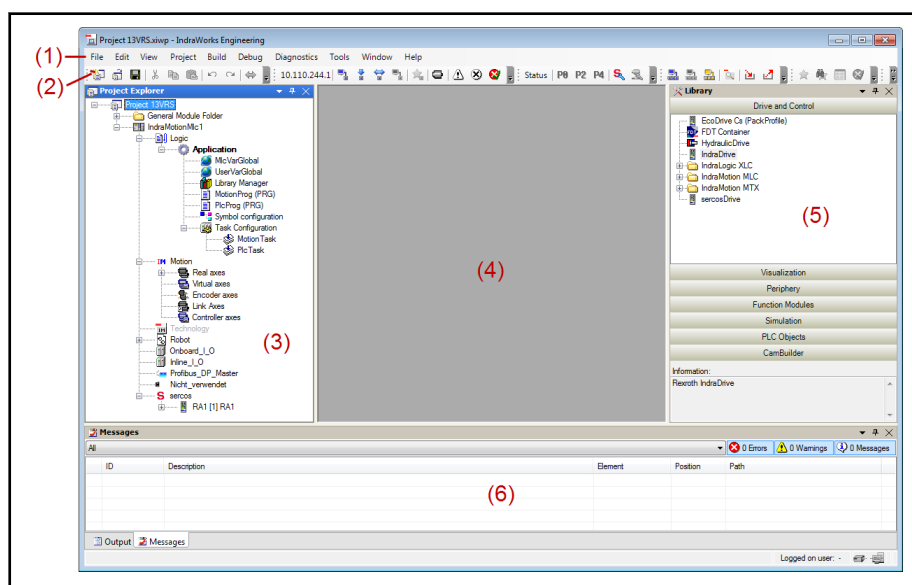
Area ⑤ is reserved for the Project Explorer. In the Project Explorer, the library elements are arranged according to the preferences of the user. Working with the Project Explorer is described in the following chapters.

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

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

Messages about the current project are displayed in the "Messages" ⑧ window.

3.5 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. 3-10: IndraWorks interface with default settings

Menu bar ① The menu bar ① and menu items are freely configurable. 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. Keep the mouse button pressed to store the individual menu items in any menu of the menu bar. When the "Customize" dialog is open, right-click to open a context menu for each menu and menu item in the menu bar. Rename or delete (among other functions) entries using the context menu.

Toolbar ② The toolbar is ② freely configurable. 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. Keep the mouse button pressed to store the individual icons in any toolbar. When the "Customize" dialog is open, right-click to open a context menu for each icon in the toolbar. 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.

Commissioning and operation

- Device libraries** ⑤ The devices available for an IndraWorks project are stored in the device libraries. The devices are classified into different groups:
- Drive and Control
 - Visualization
 - Periphery
 - Function modules
- etc.
- Output window** ⑥ Status and diagnostic information on the current project are output in the output window.
- Windows** Freely move and dock the individual windows, 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 (refer to the following figure).

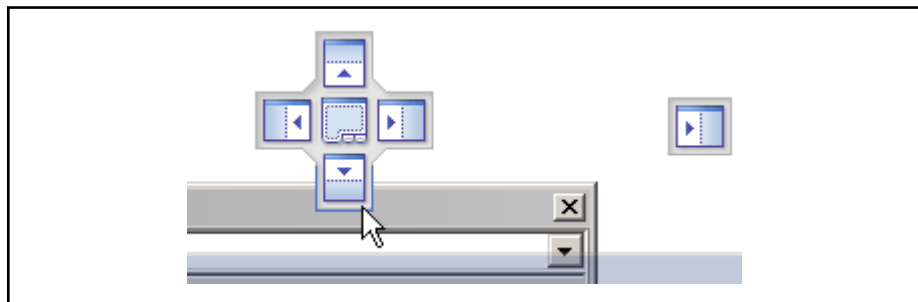


Fig. 3-11: Docking arrows

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

- Resetting interface layout** The standard user interface can be reset via the **Window ► Reset window layout** menu.



For more information on the user interface and handling of IndraWorks, refer to the documentation "Rexroth IndraWorks Engineering 13VRS" (see [chapter 1.4 "Required and supplementing documentations"](#) on page 18).

3.6 Creating a project

A new project is created via the "[IndraWorks – Startup](#)" on page 42 or via the menu items **File ► New ► Project**.

When IndraWorks projects are created, an ID is given in 13VRS and higher. This version ID indicates the minimum IndraWorks installation before editing the project. For more information, refer to the manual "Rexroth IndraWorks 13VRS Engineering", chapter "Working with IndraWorks ⇒ Compatibility mode", see [chapter 1.4 "Required and supplementing documentations"](#) on page 18.

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

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

The following fonts are set by default:

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

Commissioning and operation

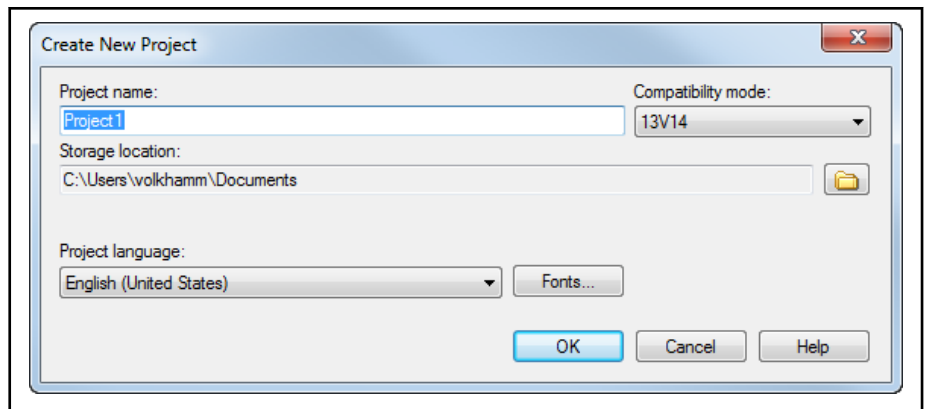


Fig. 3-12: Creating a new project – Specifying the name



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 of the project

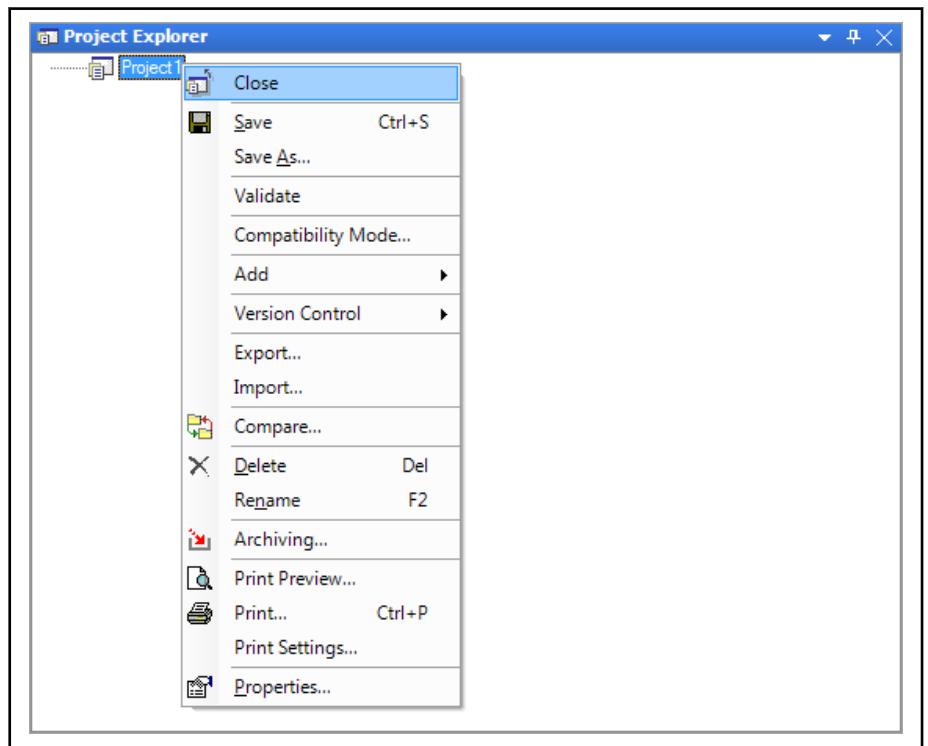


Fig. 3-13: Context menu of an empty project

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** - Current project state is stored 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 7.12 "Validating Motion configuration" on page 365](#)

Commissioning and operation

- **Add** – Allows to add a new folder, new file or new element to the project
- **Version Control** – Allows to record changes in documents or files, see [chapter 4.5 "Version control of the control" on page 159](#)
- **Export...** - Saves elements from the current project to a file [chapter 4.4.13 "Exporting" on page 148](#)
- **Import...** - Imports elements from a file into the current project, see [chapter 4.4.14 "Importing" on page 149](#)
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging" on page 151](#)
- **Delete** - Deletes the created project
- **Rename** – Allows to rename the project
- **Archive** - Current project state is stored in the file system or on the device as a password-protected data record (optional), see [chapter 4.6 "Complete data backup" on page 160](#)
- **Print Preview...** - Provides a preview of project and device data to be printed
- **Print...** - Printing project and device data
- **Print Settings** - Selects font and size for printing
- **Properties** - displays project name and path



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 controlled by the Suite. However, the user can trigger the saving process themselves at this place.

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

Refer to "Rexroth IndraWorks 13VRS CamBuilder" [chapter 1.4 "Required and supplementing documentations" on page 18](#).

Operating strategy

There are the following 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 [chapter 3.7 "Creating a control" on page 48](#)).
- Right-click to define the element in the Project Explorer at the desired position (this option is used with logic elements, e.g. virtual axis, see also [chapter 6.3.2 "Creating a virtual axis \(virtual master axis\)" on page 222](#)).

3.7 Creating a control

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

Creating control via drag&drop

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 the element are shown in the information window.

Commissioning and operation

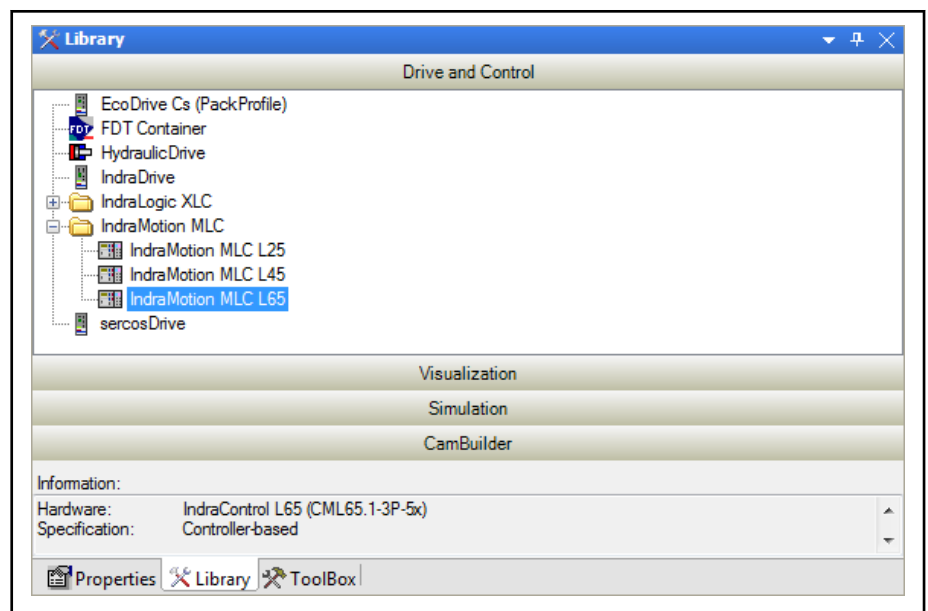


Fig. 3-14: Creating a control (as shown for the IndraMotion MLC L65) - Dragging from the library



Fig. 3-15: Creating a control - Dropping it in the Project Explorer

Creating a control via the context menu

Right-click on the project node to open the context menu of the project. Select the control via the menu item "Add".

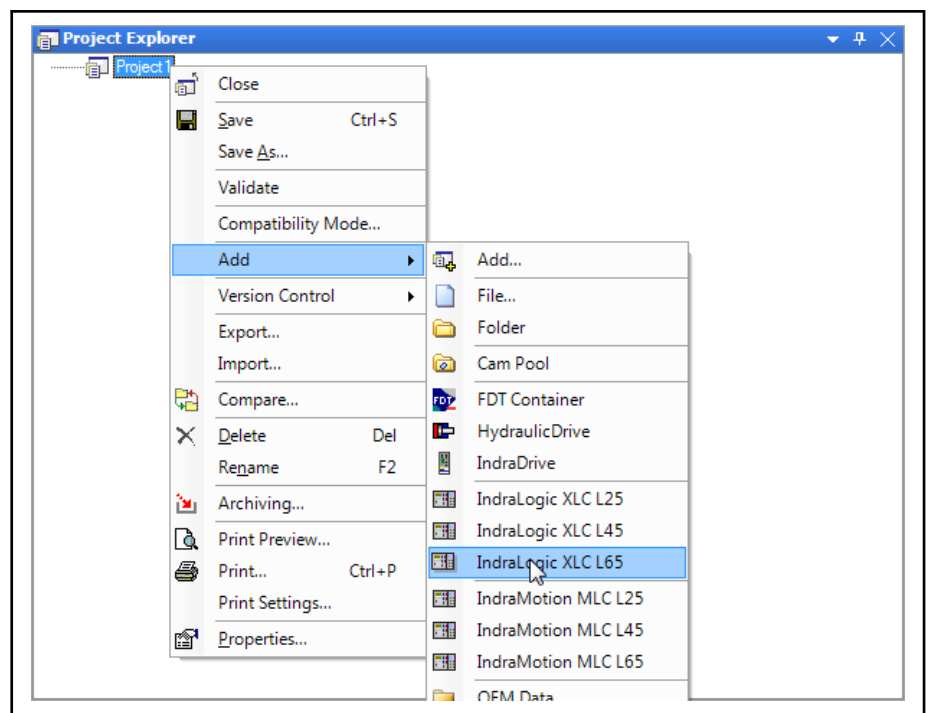


Fig. 3-16: Creating an IndraLogic XLC in the project

Creating a control by copying

A control can also be added via "copy and paste". A control that already exists in the project is copied and pasted at the project node.

Commissioning and operation

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

Fig. 3-17: Creating a control - General Settings

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

Apply or change the preset values. A comment is optional and subsequently displayed as tooltip in the Project Explorer.

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

Fig. 3-18: Creating a control (shown for the IndraMotion MLC L65) - Configuration

Device type The "Device Type" shows the system and the hardware of the control to be created. If there are several hardware variants for one system, select the respective hardware variant. The control hardware is written on the type plate of the control.

Commissioning and operation

The following hardware types are available for the IndraLogic XLC:

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

Tab. 3-3: IndraLogic XLC hardware variants

The selected hardware platform can be subsequently modified via the "Properties" dialog of the device (open via context menu of the device). See also [chapter 4.4.19 "Control properties" on page 157](#).

Firmware version	The firmware version generally corresponds to the IndraWorks Suite version.
Firmware release	The firmware release indicates the firmware state within the version. The firmware release can be subsequently updated. If the project was loaded in the compatibility mode, this compatibility mode has to be set to the current IndraWorks version before if necessary. For more information, refer to the manual "Rexroth IndraWorks "13VRS Engineering, chapter "Working with IndraWorks ⇒ Compatibility mode", see chapter 1.4 "Required and supplementing documentations" on page 18.
IP address	The IP address of the control "has to" be set in the Configuration step. Otherwise, there is no communication with the control. The IP address corresponds to the value set in the chapter 3.3.4 "Setting the IP address" on page 38 section.
PLC gateway	The PLC gateway is generally located on the same computer on which IndraWorks is running. Generally, this setting can thus left unchanged.
PLC communication	From version 13VRS, it can be chosen between two PLC communication types. TCP is recommended due to a better firewall handling and no required broadcast (no broadcast towers). Due to the protocol, it is slower than UDP. UDP is required (mandatory) for SafeLogic due to internal routing functions.
Connection test	First, the connection test checks whether a control with this IP address (ping) exists and responds with the device name of the control and the current user name (author). Both can differ from the device name and the author listed in the previous figure. In the second step, the PLC communication is checked.
Secure online mode	The "Secure Online Mode" switch enables additional requests to start and stop the PLC and to force values in the PLC monitoring. Due to security reasons, the "Secure Online Mode" is enabled by default.
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 function modules 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.

Commissioning and operation

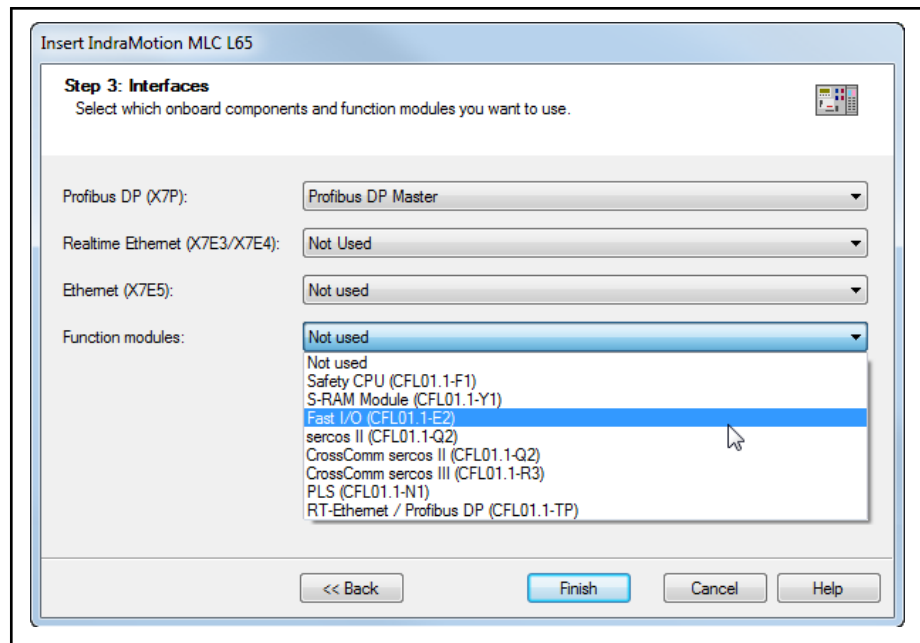


Fig. 3-19: Creating a control (shown for the IndraMotion MLC L65) - Specifying the function modules connected, here "Fast I/O (CFL01.1-E2)"



A function module can also be entered into the IndraLogic XLC Lxx even if the function module 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.

Click on **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 4.1 "Project structure" on page 81](#).

Context menu

The project structure is explained in [chapter 4.4 "Control – Context menu" on page 94](#).

3.8 Creating a function module



Function modules are only available for Lxx variants in case of IndraLogic XLC controls.

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

Specifying function module when configuring 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 added to the right location in the Project Explorer.

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

Commissioning and operation

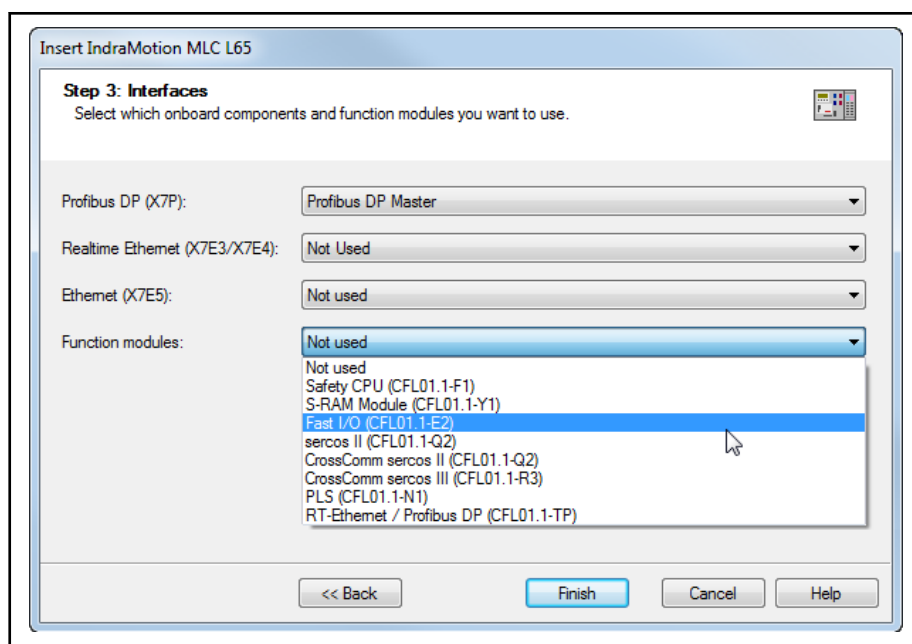


Fig. 3-20: Inserting a Fast I/O function module using a wizard (for an IndraMotion MLC for example)

Inserting a module subsequently
to an already configured control

Function modules to be subsequently connected can be selected via the "Properties" dialog of the control, via the context menu item **Add** of the control or via drag&drop from the library and stored in the Project Explorer on the control.

Drag&drop from the device library:

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

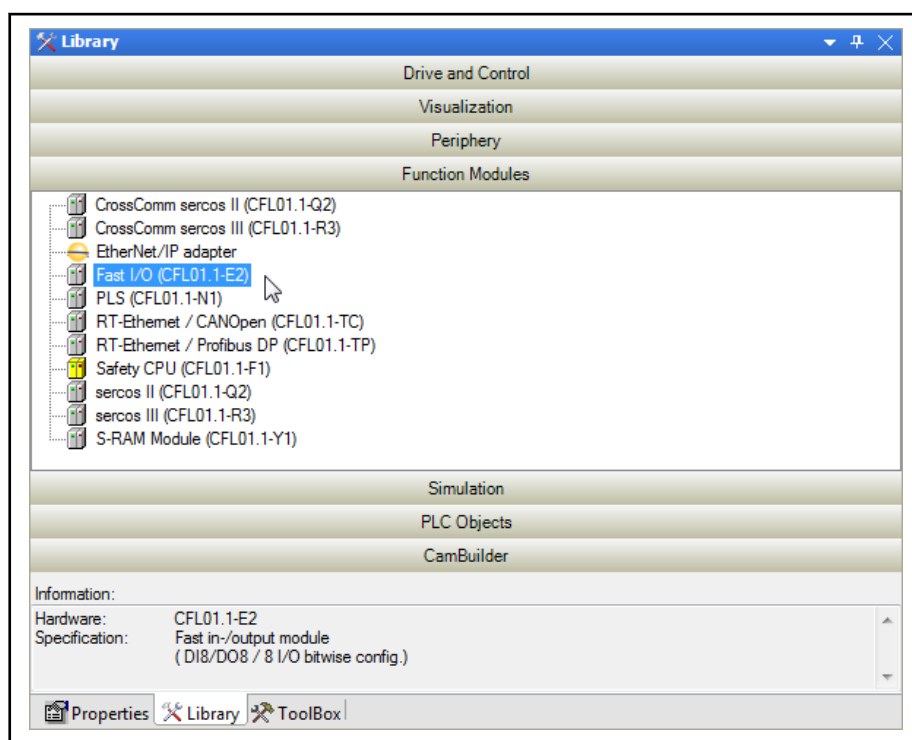


Fig. 3-21: Dragging a module from the library

.... and "drop" it on the control or above the Sercos folder.

Commissioning and operation

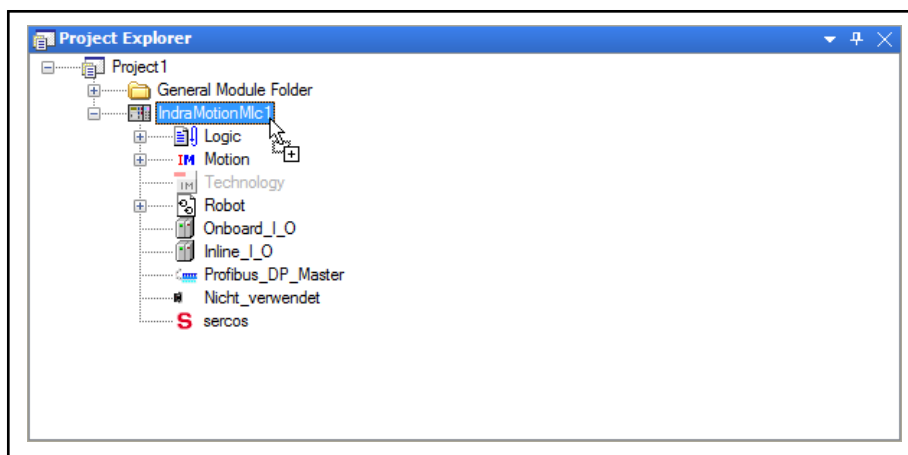


Fig. 3-22: Inserting a module



A subsequently connected module is only detected upon the next control startup.

Configuring the module

Each module has to be subsequently configured via corresponding dialogs.

The function modules currently available for the IndraLogic XLC, as well as their functionality and configuration, are described in "Rexroth IndraControl Lxx 13VRS Function Modules", see [chapter 1.4 "Required and supplementing documentations"](#) on page 18.

3.9 I/Os and field buses

3.9.1 Creating Onboard I/Os

The IndraLogic XLC L45 and IndraLogic XLC L65 provide the user with 1 byte (fast) of inputs/outputs. There are no onboard I/Os available on the IndraLogic XLC 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.

Commissioning and operation

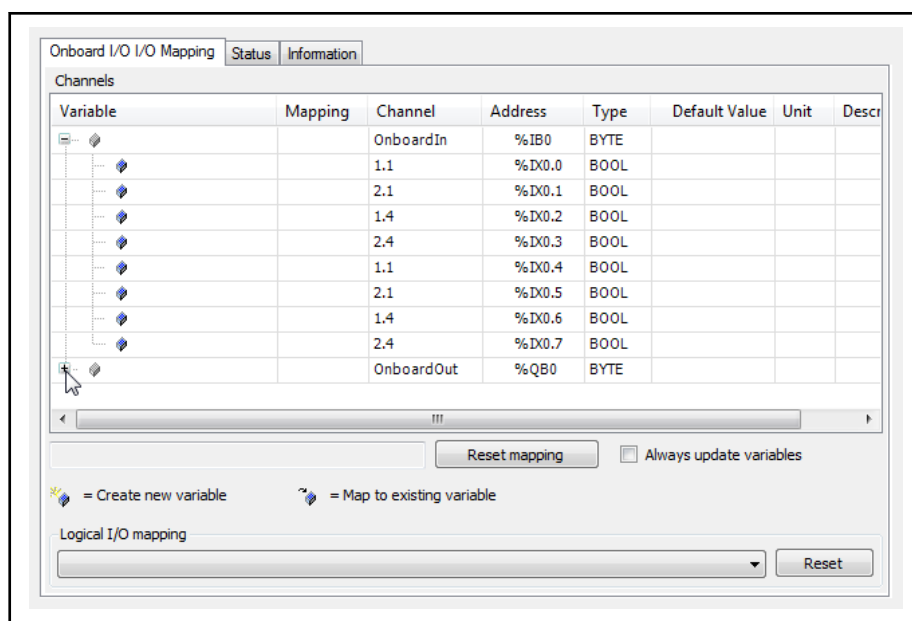


Fig. 3-23: Creating local I/O – Declaring inputs and outputs

The starting addresses for the input and output bytes are assigned automatically, but can be changed subsequently (see [fig. 3-31 "Labeling fixedly specified address" on page 60](#)).



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.

"Onboard I/O Mapping" 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 symbol 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 into the **Mapping** column. Click on the icon and the variable is automatically **declared internally as global variable** and the icon changes to . After this point in time, the variable is globally available in the application.
- Channel:** Symbolic channel name
- Address:** I/O address. Enter the 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 labeled 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:** Preassignment of the variable (TRUE or FALSE)

Commissioning and operation

- **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.

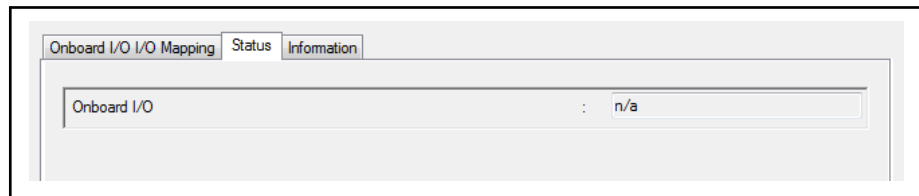


Fig. 3-24: Onboard status workspace

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

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

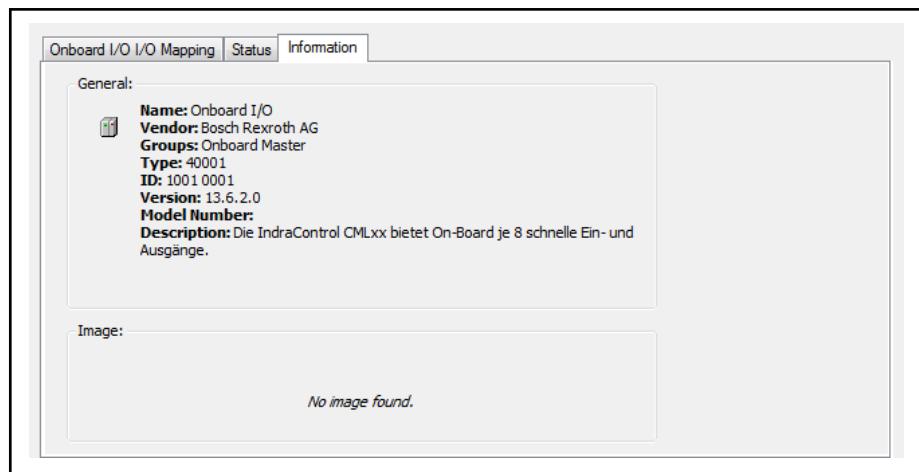


Fig. 3-25: Onboard information workspace

This subdialog of the device dialog (Device Editor) displays some general information from the device description file on the device currently selected in the device tree.

The following information is displayed:

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

3.9.2 Creating Inline I/Os

Inline modules are connected at the right of the IndraLogic XLC control.

Commissioning and operation

For details, refer to "Rexroth IndraControl L45/L65/L85 Control" in [chapter 1.4 "Required and supplementing documentations"](#) on page 18.



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



Check that 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.

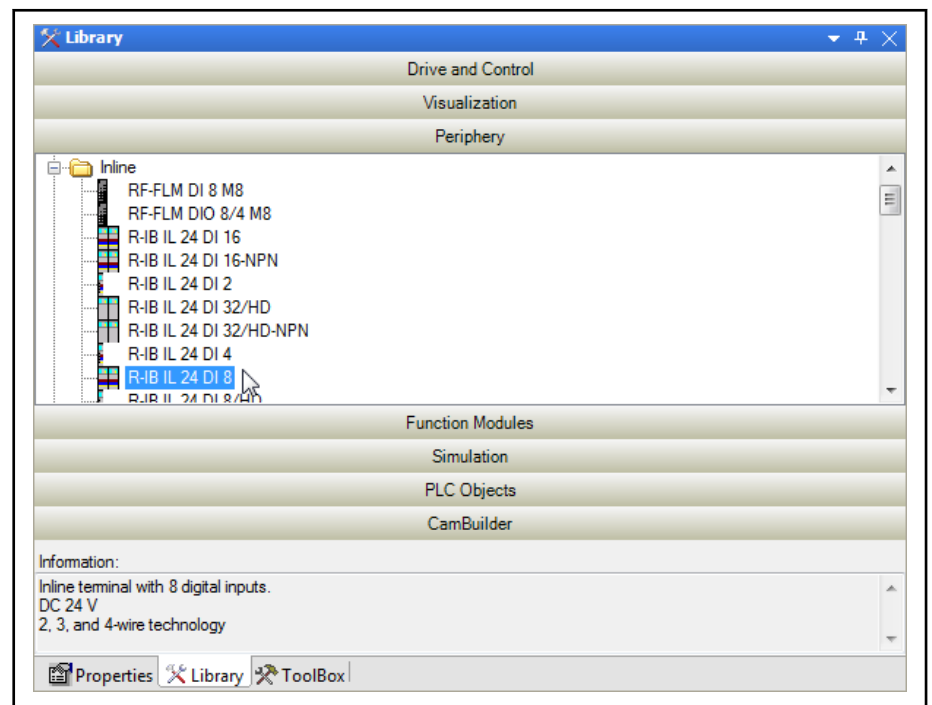


Fig. 3-26: Creating Inline I/Os – Selecting and dragging the module from the library

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



Observe the assembly sequence during creation!

Commissioning and operation

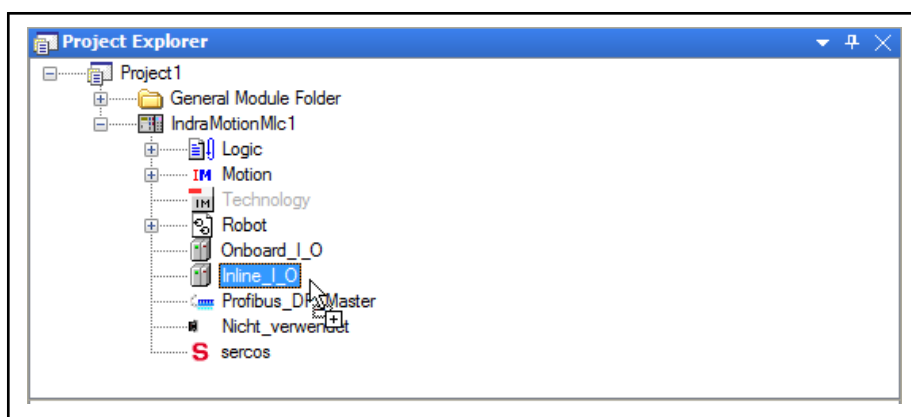


Fig. 3-27: Creating Inline I/Os – Dropping to the Project Explorer

or

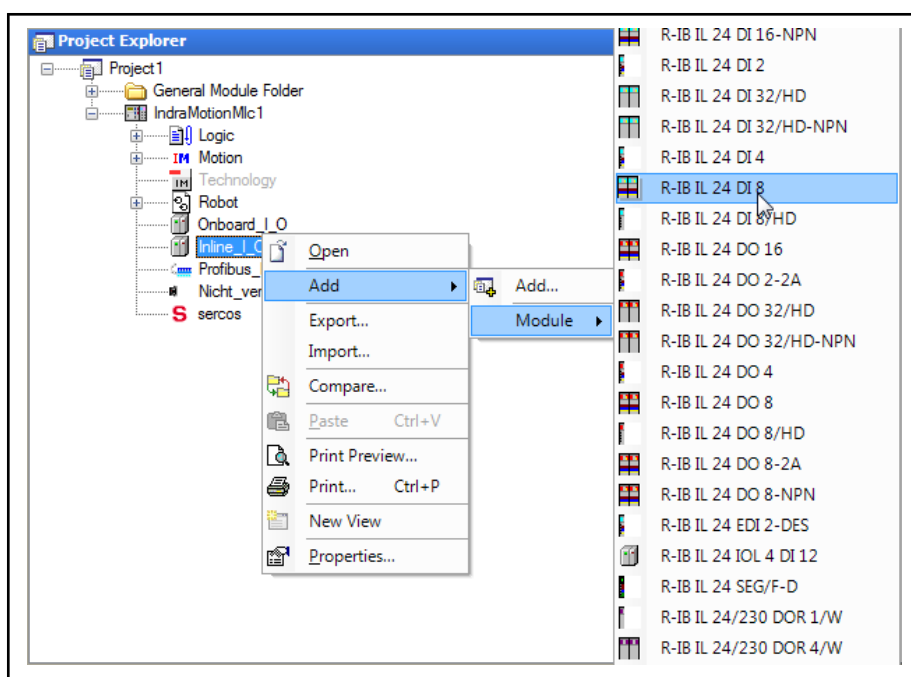


Fig. 3-28: Creating Inline I/Os – Selection via context menu

Settings

Make the required module settings for the respective module.

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

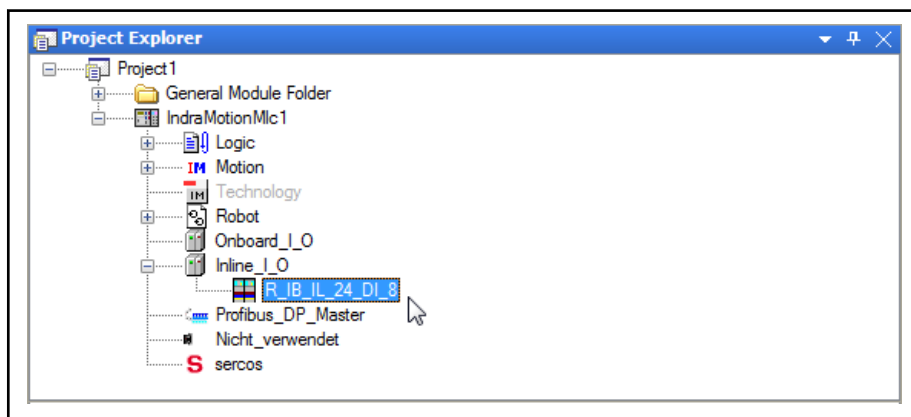


Fig. 3-29: Inline I/O module in the Project Explorer

Commissioning and operation

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

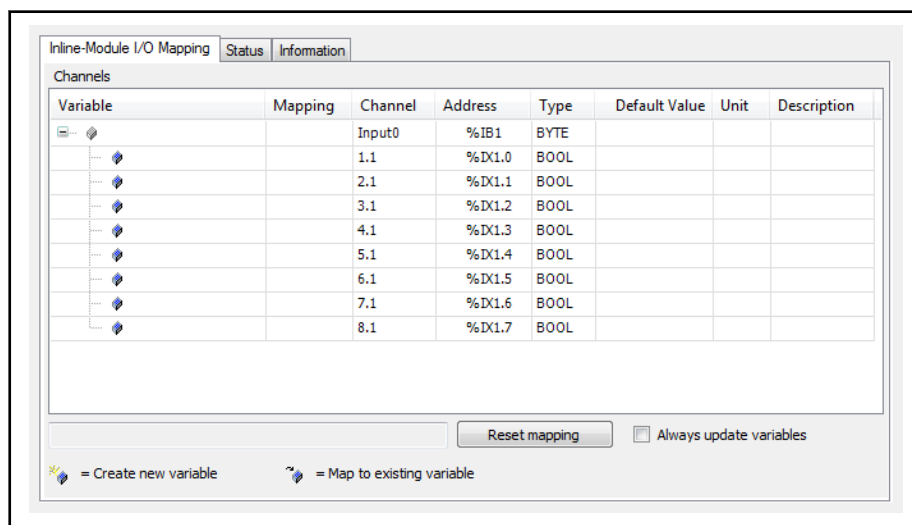


Fig. 3-30: "Inline Module I/O Mapping" workspace

"Inline Module I/O Mapping" tab

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

- Variable:** This column shows the outputs and inputs of the module (in this example). Click on the plus or minus symbol 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 into the Mapping column. Click on the icon and the variable is automatically **declared internally as global variable** and the icon changes to . After this point in time, the variable is globally available in the application.
- Channel:** Symbolic channel name
- Address:** I/O address. Enter the I/O address as byte address (e.g. %IB100). Entries in italics are for display purposes only and cannot be edited.

Manually and fixedly specified addresses are labeled with this symbol



Fixedly specified addresses do not change when inserting additional I/O modules.

Commissioning and operation

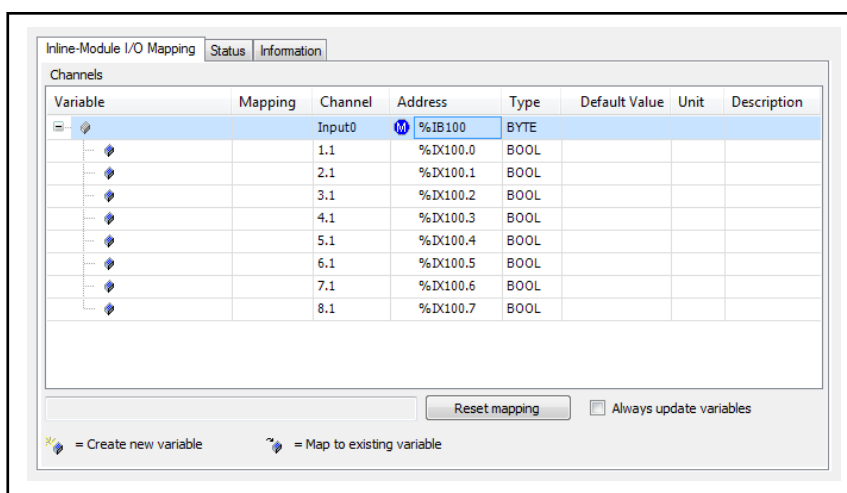


Fig. 3-31: Labeling fixedly specified address

- **Type:** Byte addresses are labeled 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:** Preassignment 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. 3-32: Inline status workspace

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

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

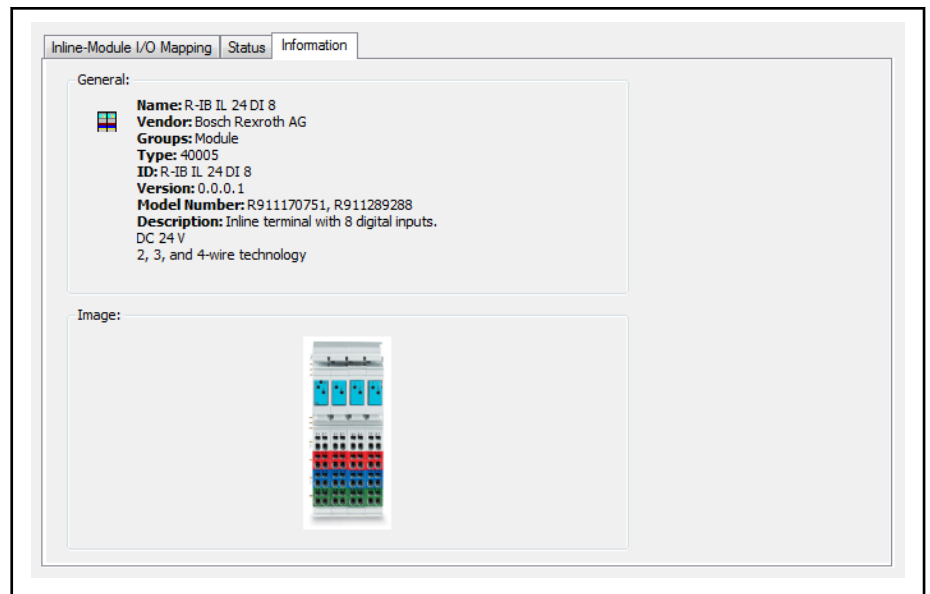


Fig. 3-33: Inline information workspace

This subdialog of the device dialog (Device Editor) displays some general information from the device description file on the device currently selected in the device tree.

The following information is displayed:

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

Enabling/disabling modules via the Project Explorer

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

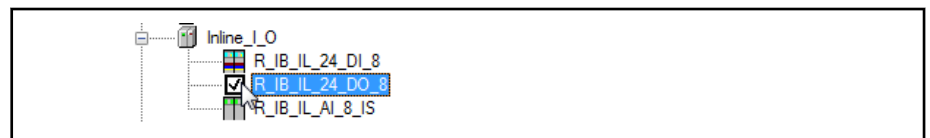


Fig. 3-34: Selecting/deselecting an Inline module

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.



Fig. 3-35: Enabled state of an Inline module

Commissioning and operation



Fig. 3-36: Disabled state of an Inline module

Deactivating modules via "User-Defs.cfg"

Use this function to disable Inline I/O modules configured in IndraWorks. Disable them by the definition in a stored file located on the Compact Flash card of the control. This impedes diagnostic messages for modules that are configured, but do not exist.

Enabling the function

The function is enabled by its existence and a valid file content "User-Defs.cfg", stored in the partition "\ata0b\" (USER) of the Compact Flash card. Another path or file name can be defined by making entries in the "cfg file" of a system.

Example:

```
[IOSpecialDefs]
IODeactFilePath=MyFilePath
IODeactFileName=MyFileName
```

Point in time of module deactivation

The "UserDefs.cfg" is analyzed and the entered modules are disabled at this point in time and in the context of the internal event "PrepareUpdateConfiguration". Evaluation takes place in case of the following actions:

- Startup after power on
- Reset (warm)
- Reset (cold)
- Load program after reset (origin)

Structure of the file "UserDefs.cfg"

The file is structured in a single or in multiple existing sections and the following contained entries:

- **Sections**

The denomination of a section is written in "[" and "]". A section ends with the beginning of the next section or with the end of the file.

The name consists of a symbol "InlineIO.Master." and the constant index "0" (due to the compatibility to the Profibus slave deactivation).

Example: [InlineIO.Master.0]

Special function: Due to the definition of character "X" for the index instead of a character, this section is ignored in the evaluation.

Several sections can be defined for the master index "0". All their entries are assigned to the Inline master. Together with the "x function" described above, different module groups can be handled easily

- **Module entry**

An entry to disable an Inline module consists of the keyword "ModuleDeactPosi", the assignment character "=" and the 1- or 2-digit decimal specification of the module position.

Example: ModuleDeactPosi = 3

The valid value range is 1...64. "0" is also valid, but not effective

- **Comments**

A semicolon (";") is defined as comment character. The text is not analyzed from the position of the semicolon up to the end of the line.

Example of the file "UserDefs.cfg"

```
;----- Master.0 -----  
  
[InlineIO.Master.0]  
  
ModuleDeactPosi = 3  
ModuleDeactPosi = 7  
ModuleDeactPosi = 15  
ModuleDeactPosi = 16  
  
;-----
```

3.9.3 Creating Profibus I/Os

The Profibus node is located below the Profibus I/O node. By default, this node is configured as master. Via the context menu or when creating a control (see Profibus DP configuration in [chapter 3.7 "Creating a control" on page 48](#)), it can, however, also be reconfigured as slave. Thus, I/Os created under a slave can be activated by a master of another control. However, the rest of this description is limited to the configuration and creation of a master and its slaves.

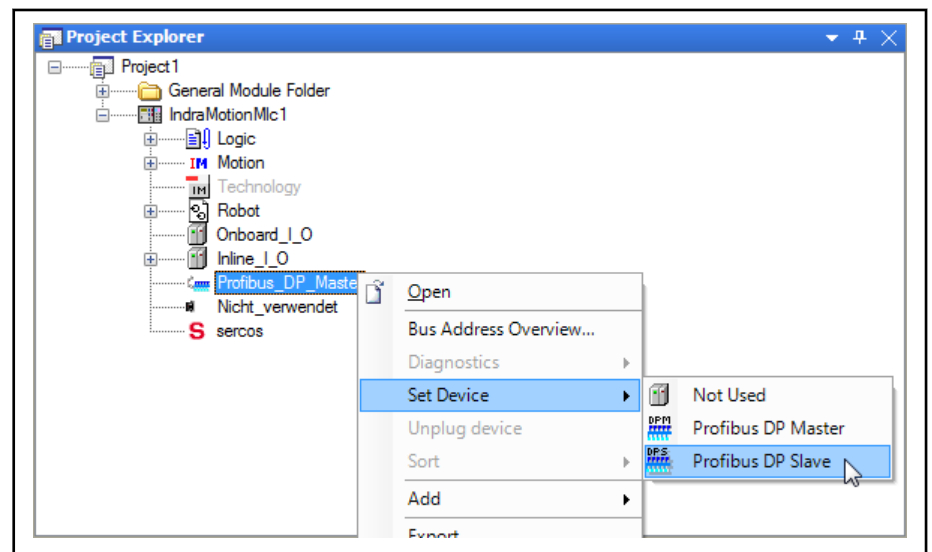


Fig. 3-37: Reconfiguring Profibus node as slave

Profibus DP master and slave configuration

The configuration of the Profibus DP master and slave is described in "Rexroth IndraLogic XLC IndraMotion MLC 13VRS Field Buses" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

3.9.4 Creating a Profinet I/O 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 I/O is based on the Profibus DP function model, but uses **Ethernet TCP/IP** for cyclic communication between the central control and the decen-

Commissioning and operation

tralized field devices. Profinet I/O 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 I/O Devices") are described by the manufacturer in so-called "device description files", which can be loaded using the device database after the device description files 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.

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

Configuring the Profinet I/O controller and the I/O devices

The configuration of the Profibus I/O controller and the I/O devices is described in "Rexroth IndraLogic XLC IndraMotion MLC 13VRS Field Buses" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

3.9.5 Creating Sercos III I/Os

The Sercos III master has the following functional characteristics:

- Support of up to 99 Sercos III I/O devices (32 in the case of L25)
- 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.

Configuring Sercos III master und slaves

The configuration of the Sercos III master and slave is described in "Rexroth IndraLogic XLC IndraMotion MLC 13VRS Field Buses" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Replacing hardware 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.

IndraLogic XLC 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.

Commissioning and operation

5. The **Remote Address Assignment** button is used to set the Sercos address of the devices.
6. Clear error and restart the PLC.

Option 2: With the IndraMotion Service Tool.

IndraLogic XLC 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 the found Sercos device are displayed according to the topology address. The topology addresses are output according to their wiring order, starting from X7E1 of the IndraLogic XLC (also for ring wiring). For devices connected to X7E2 of the IndraLogic XLC, 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;
```


Commissioning and operation

```

    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 sercos III devices * 2 (2 bytes per device)
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
    // Confirmation 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 here fixedly 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 IndraLogic XLC (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. Following successful writing of the configuration, delete the error and switch to the operating mode. Both can be executed via the `ImcCtrl` structure of the IMC interface.

Variant 3 b): Retrieving 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 SIII 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 if exchange of I/O or drive...
    // The user has to acknowledge this by bConfirmReadSercosConfig

IF bConfirmReadSercosConfig = TRUE THEN
    // Confirmation 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
```

Commissioning and operation

```

// of an exchange of an I/O or drive...
// The user has to acknowledge this by bConfirmWriteSercosConfig

IF bConfirmWriteSercosConfig = TRUE AND
// Confirmation 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
// 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.

Following successful writing of the configuration, delete the error and switch to the operating mode. Both can be executed via the ImcCtrl structure of the IMC interface.

3.10 IP communication

3.10.1 Overview



The settings for the IP communication with IndraLogic XLC are only available with a Sercos interface on the Lxx variants.

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 a fast backup/restoration of the drives of the control via Ethernet are possible. Direct communication of the control with a superordinate HMI or Enterprise Resource Planning System (ERP) is also 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 con-

Commissioning and operation

figuration and capabilities of the control system to integrate it into an existing network.

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 I/Os of the system
3. The onboard **Realtime Ethernet** interface is a configurable multi-protocol interface to exchange data with the periphery or a third-party head control. This interface can be configured and supports multiple Realtime Ethernet protocols such as Ethernet IP, Profinet, Standard TCP/IP (see documentation "Rexroth IndraWorks 13VRS Field Buses" in the chapter "TCP/IP and UDP communication via field bus interface", see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).
4. Up to 64 controls can communicate in one 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.



C2C functionality via the optional function module **CrossComm Sercos III (CFL01.1-R3)** is only available for IndraMotion MLC.

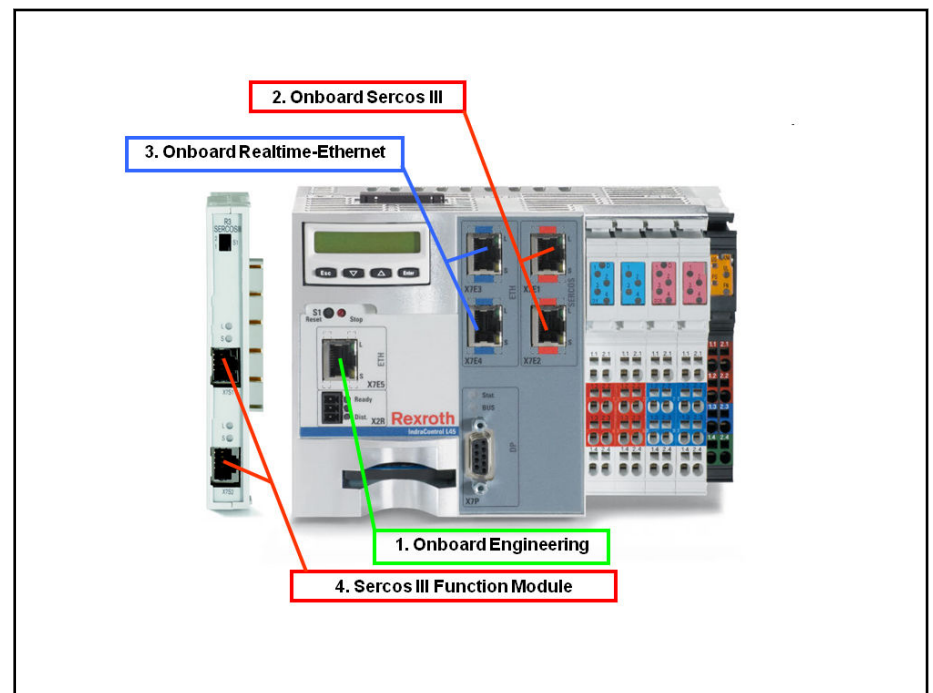


Fig. 3-38: Ethernet interfaces

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

- **Connectivity:** Each free Ethernet interface provides the option of participating in 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 device

Commissioning and operation

- **Flexible:** Special requirements and topologies can be implemented by means of manual IP settings

3.10.2 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. The bridge forwards the received data telegram using the MAC address. Bridges work on layer 2 (backup/data link layer) of the OSI reference model.

- Advantage: Simple design or implementation, no maintenance
- Disadvantage: Possible loops can interfere with the network.

Router/routing

Routing refers to an IP forwarding procedure. A router consists of several network interfaces, which also logically have 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 accordingly (packages are routed). Forwarded packages either arrive in a directly connected target network (also target subnetworks) known to the router 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 significant
- Disadvantage: More technical effort than for bridging, greater configuration effort

3.10.3 Default setting

The IP default values contain the following settings:

- All Sercos III interfaces of the control are part of the **bridge** and thus provided with the same 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 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
 - Control number (C-0-0031)

Automatically generated IP address and subnet mask

	IP address	Subnet mask
Configuration C-0-0031 = 0 (default)		
All Sercos III interfaces of the control/bridge	172.31.254.254	255.255.0.0

	IP address	Subnet mask
All Sercos III drives and I/Os	172.31.254.<SercosAddress>	255.255.0.0
Configuration 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>.<SercosAddress>	255.255.0.0

Tab. 3-4: Automatically generated IP address and subnet mask

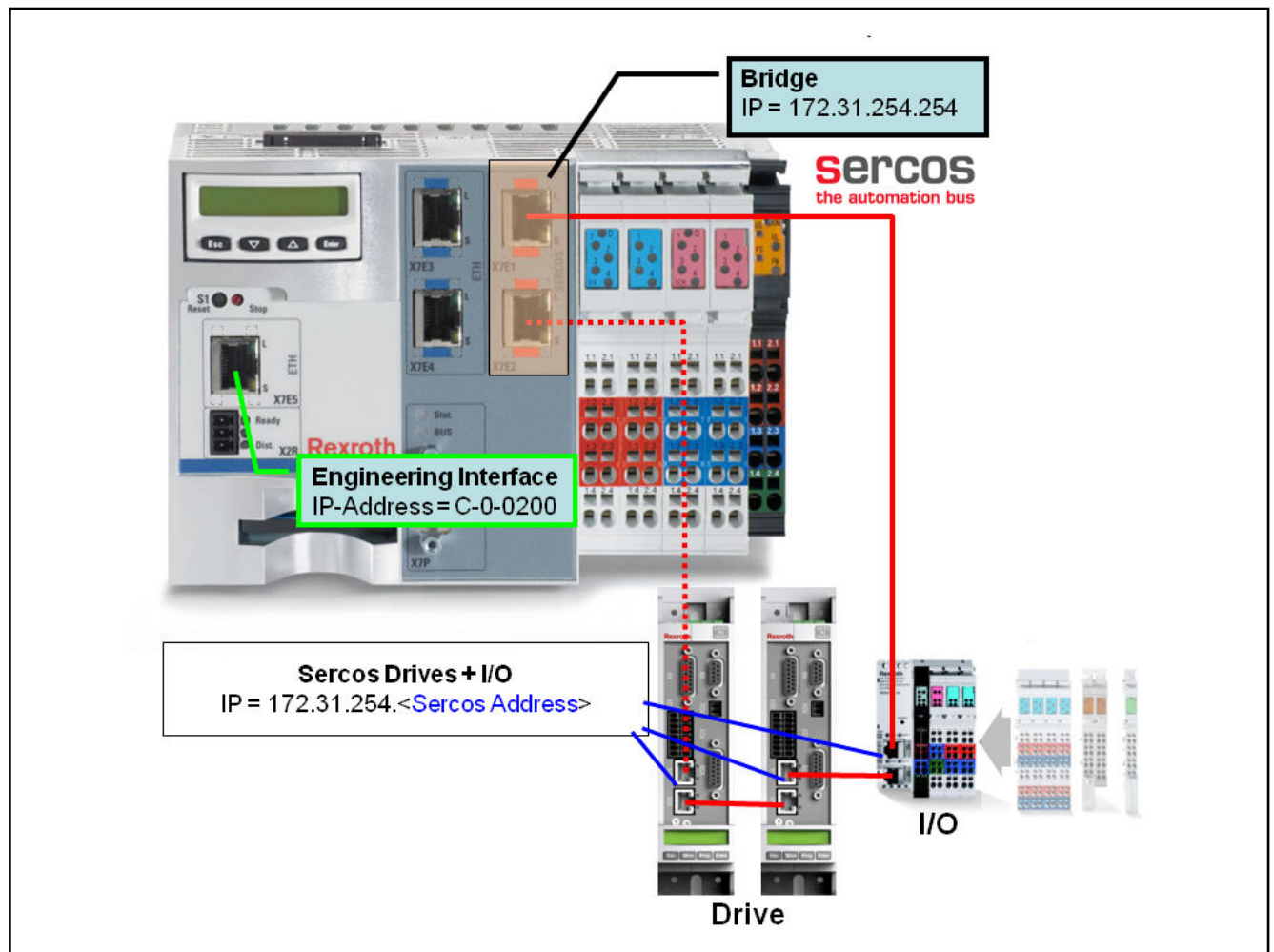


Fig. 3-39: IP settings with default configuration and C-0-0031=0

Commissioning and operation

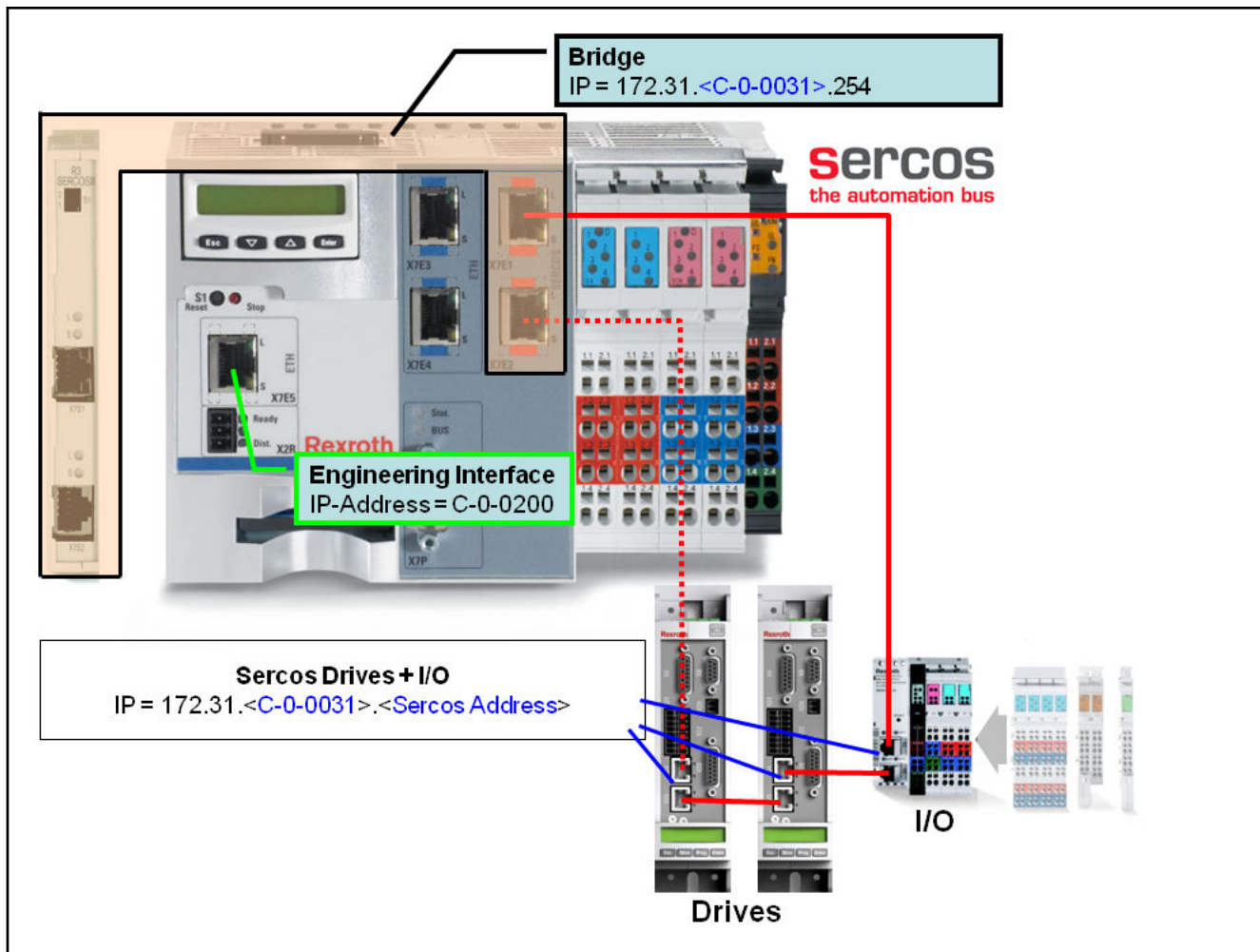


Fig. 3-40: IP settings with default configuration and C-0-0031 > 0

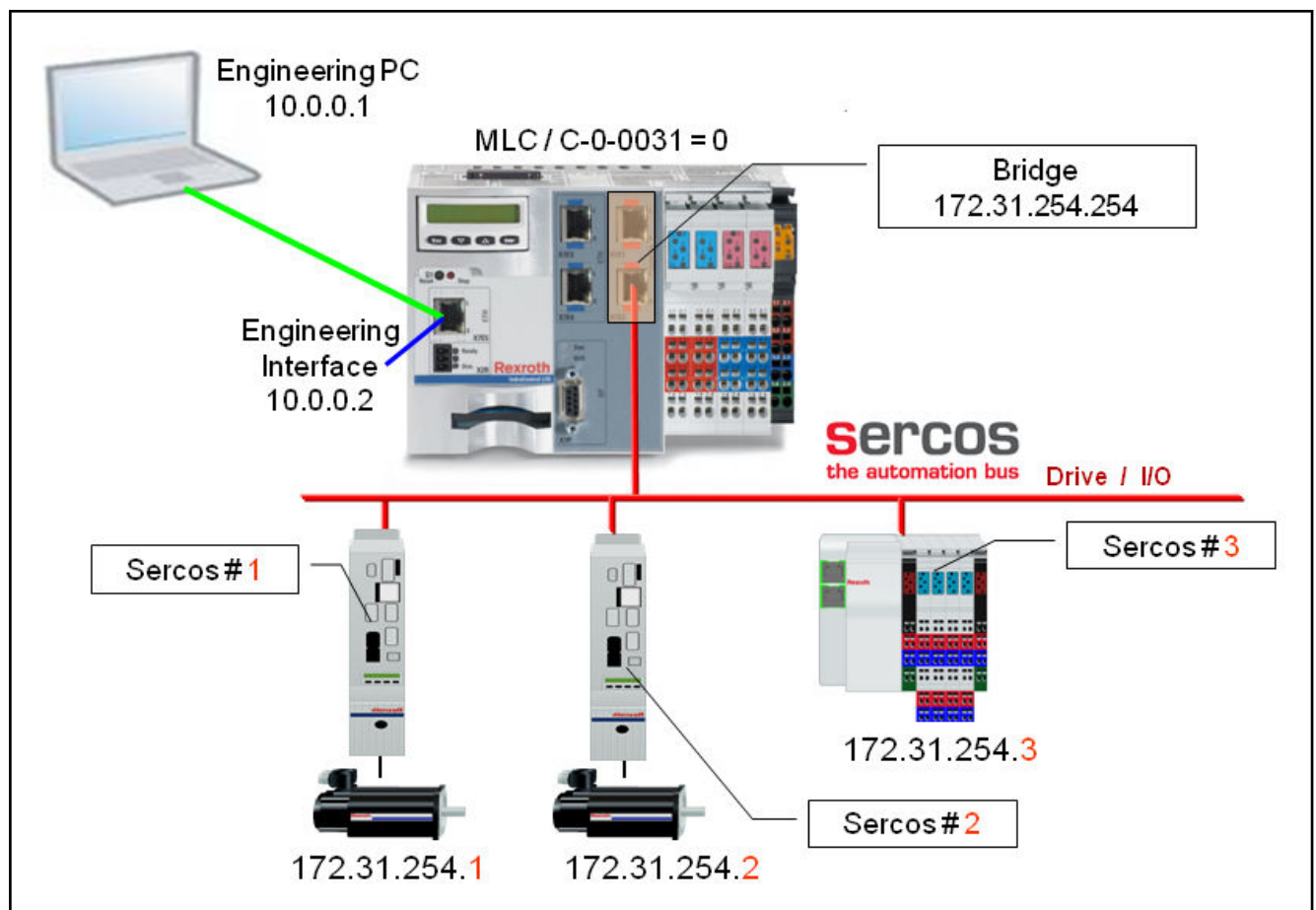


Fig. 3-41: Example: Minimum topology (default configuration)

Commissioning and operation

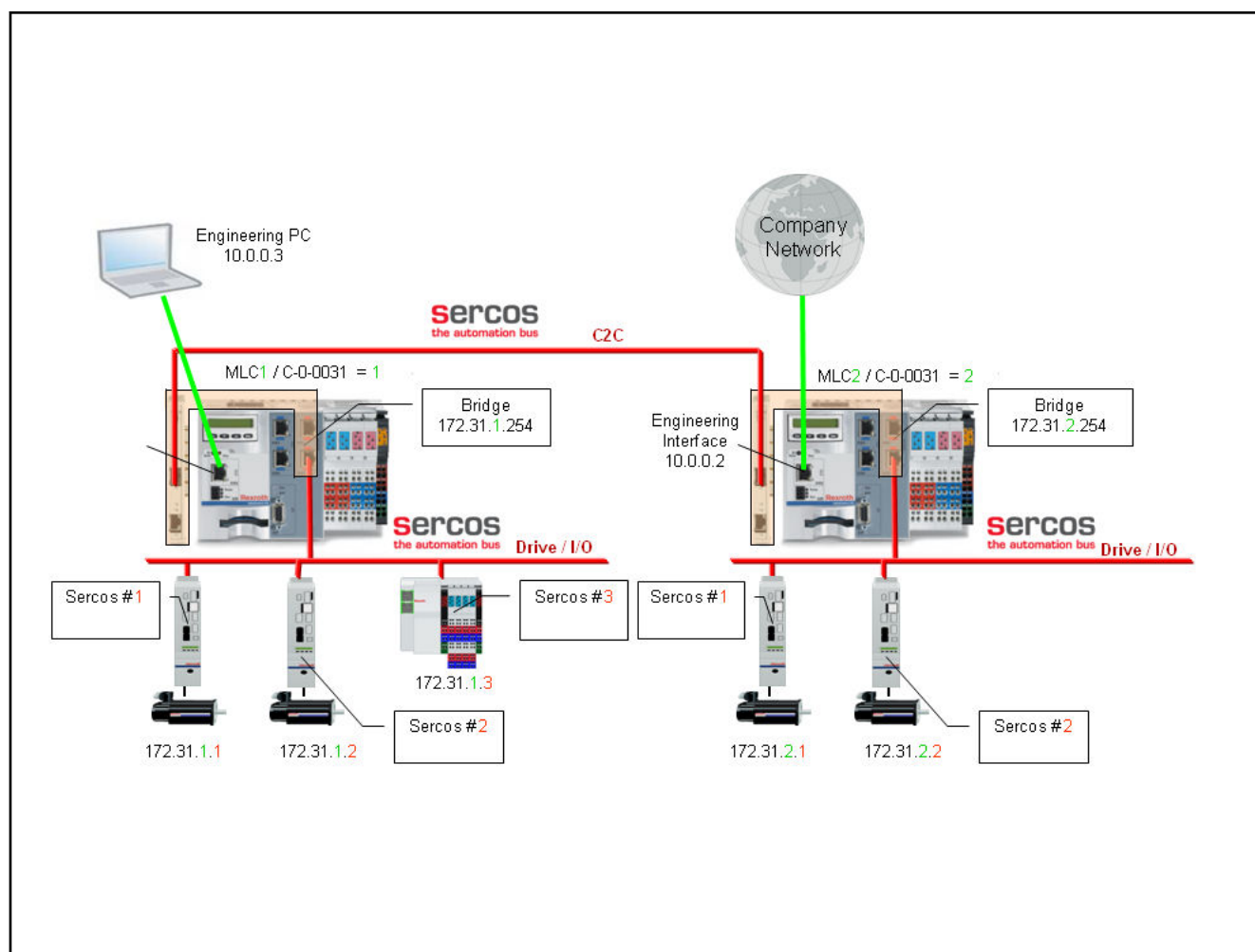


Fig. 3-42: Example: Bridging with the Sercos III function module (default configuration)



With the Engineering PC and via the IP channel of the Sercos III C2C connection, even the MLC2 control can be reached. 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

3.10.4 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.

Commissioning and operation

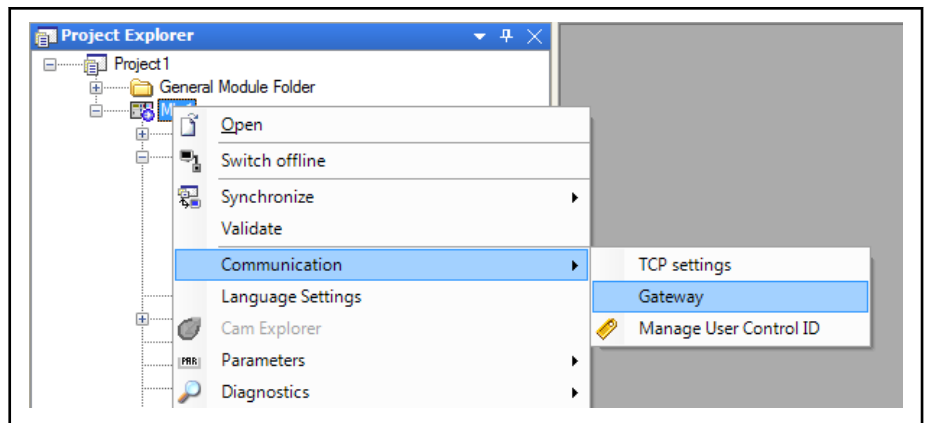


Fig. 3-43: Context menu to commission the Ethernet-based interfaces

Enable or disable the automatic address generation and the assignment to the bridge for the respective interfaces. Press **Apply Command Configuration** to apply the settings. Furthermore, load and enable the default IP settings (see [chapter 3.10.3 "Default setting" on page 70](#)) via the **Load Basic Network Parameters** button.

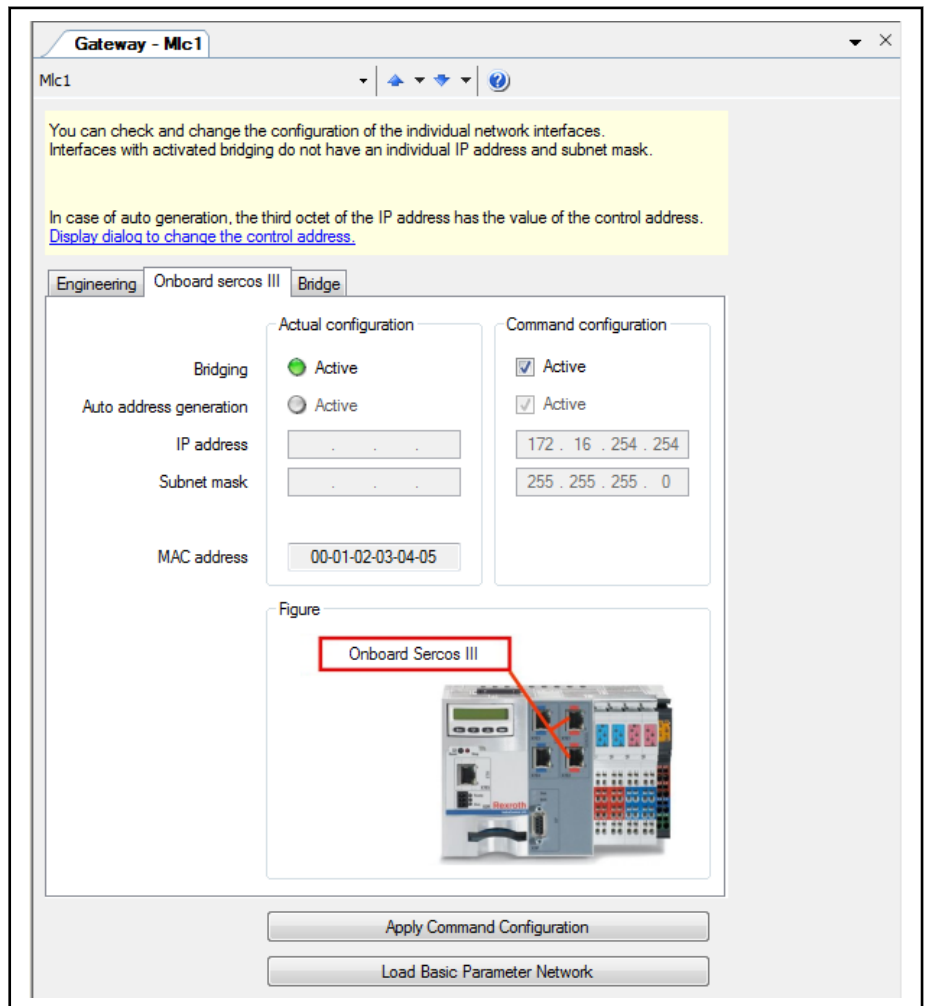


Fig. 3-44: Commissioning dialog to set the IP communication (in this case, bridging and automatic address generation enabled)

Commissioning and operation

**Onboard Realtime Ethernet
Special features**

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.

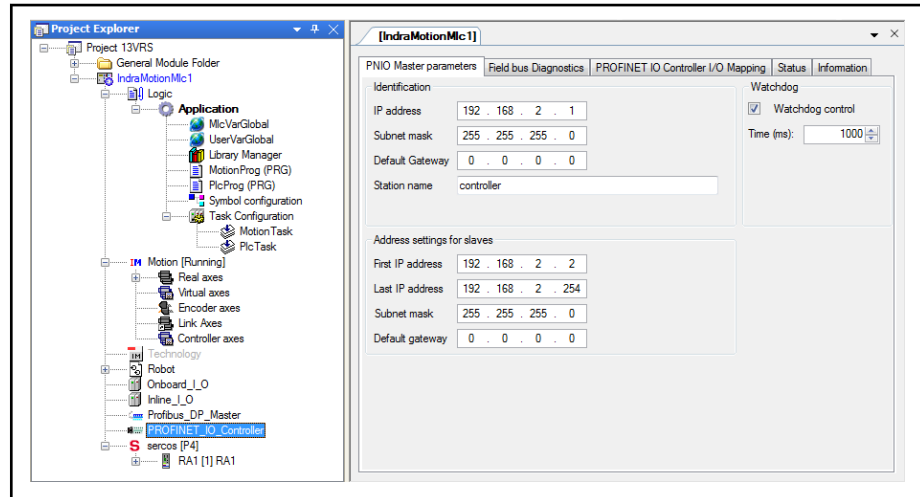


Fig. 3-45: Onboard Realtime Ethernet PROFINET_IO_Controller dialog



If the **automatic address generation** or the **bridging** is activated 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**.

3.10.5 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
Onboard Realtime 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 link)	C2C master (C-0-0701 Bit1 = TRUE)	172.16. 0.254	255.240.255.0
	C2C slave (C-0-0701 Bit1 = FALSE)	172.16.0.<C-0-0031>	255.240.0.0
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

Tab. 3-5: Automatic IP address generation of the control interfaces



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

	C-0-0031 Control address	C-0-0235 bit 0 = 1 Onboard Sercos III in bridge	C-0-0235 Bit 0 = 0 Onboard Sercos III not in bridge
Sercos III drive or I/O address	0	172.31.254.<SercosAddress>	172.16.254.<SercosAddress>
	otherwise	172.31.<C-0-0031>.<SercosAddress>	172.16.<C-0-00031>.<SercosAddress>

Tab. 3-6: Automatic IP address generation of sercos III drives and Sercos III I/Os

3.10.6 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 (for 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 for other interfaces.

Furthermore, the onboard Engineering interface and the Realtime Ethernet interface can be included into 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.

3.10.7 Involved IP communication parameters

IndraWorks commissioning dialogs (see [chapter 3.10.4 "IndraWorks commissioning dialogs" on page 74](#)) are available for the typical use cases using the parameters described below. 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 IP configuration**. 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 Realtime 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
- 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 via the command **C-0-1024 Command: Load basic parameter of IP configuration**. Furthermore, the default value of the IP address generation is written to the command value parameter (C-0-0235) and then automatically applied.

Commissioning and operation

- The parameter **C-0-0219 Status of network configuration** indicates bit by bit for the individual interfaces 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 hardware-side.

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		
Onboard Realtime Ethernet	Actual configuration	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 configuration	C-0-0230	C-0-0231		

Tab. 3-7: Overview on the most important IP communication settings

3.10.8 Ethernet ports and firewall

The control and PC communicate via Ethernet. Several protocols are used depending on the use case. Firewalls - blocking the communication on some Ethernet ports - are installed on most of the PCs by default. An enabled firewall is not critical in most use cases. If communication problems occur between control and PC, the firmware settings have to be checked.

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

Commissioning and operation

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
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
TCP 11740, 11741	Incoming	PLC	IndraWorks

Tab. 3-8: 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.

3.10.9 More information 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	Rexroth IndraMotion MLC/MLP 10VRS, MLD/MPx 06VRS OPC Communication DOK-IM*ML*-OPC*COM*V10-AWxx-EN-P
ModbusTCP	Data exchange of the control with third-party HMIs. The control is the ModbusTCP server.	Rexroth IndraWorks 13VRS Basic Libraries IndraLogic 2G DOK-IL*2G*-BASLIB**V13-LIxx-EN-P
Network variables	Control-comprehensive data exchange of PLC variables via broadcast and UDP	Rexroth IndraWorks 13VRS IndraLogic 2G PLC Programming System DOK-CTRL-IL2GPRO*V13-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 exchange master axis positions for example	Rexroth IndraControl Lxx 13VRS Function Modules, DOK-CTRL-FM*LXX**V13-APxx-EN-P

xx

Edition

Tab. 3-9:

Further information on the use of Ethernet-based interfaces

Commissioning and operation

3.11 Axes

To create the different axis types, refer to [chapter 6 "Axes" on page 207](#).

4 Control

4.1 Project structure

An IndraLogic XLC project is characterized by a control instance (e.g. Xlc1).

The control instance consists of its context menu and several nodes: a Logic, a Motion and a Sercos node**; Onboard I/O*, Inline I/O-, Profibus/M*, Profinet_IO-Controller folder* are also available Function module folders (e.g. Fast I/O (CFL01.1-E2)) can be added to the control.

* - only IndraLogic XLC L45/L65

** - not IndraLogic XLC without Sercos

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" node is provided with the following subfolders in which the individual axis instances can be created.

- Real axes
- Virtual axes
- Encoder axes

Context menu and folder differ depending on their operating state (offline/online parameterization). For the offline state, refer to [chapter 3 "Commissioning and operation"](#) on page 33.

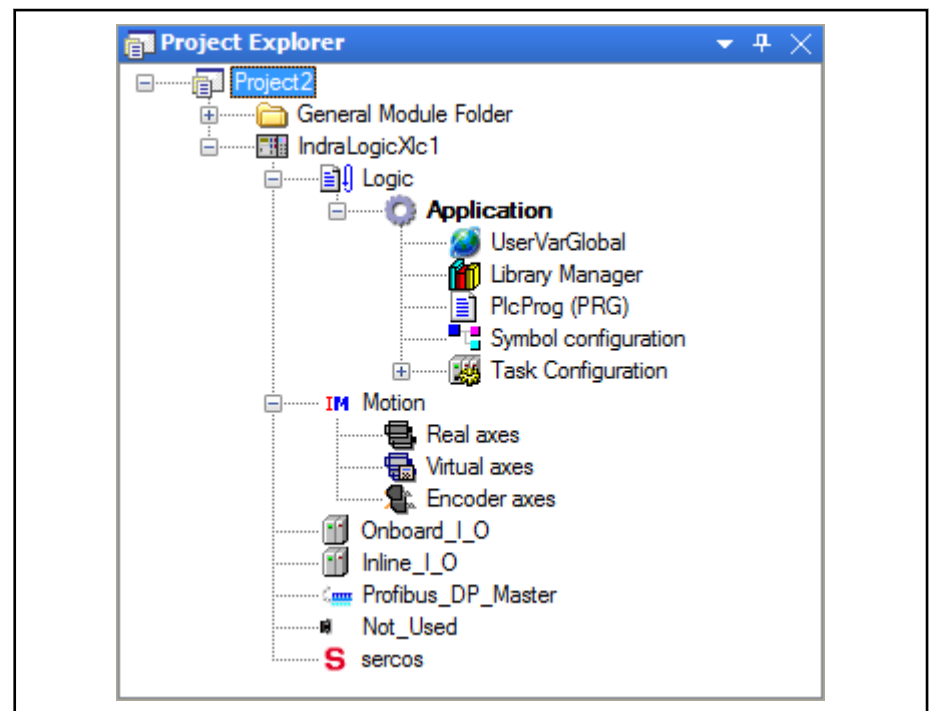


Fig. 4-1: Example: Tree structure of an IndraLogic XLC project

Control

4.2 Toolbar of the Control "XLC/MLC Info"

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



Fig. 4-2: Toolbar "XLC/MLC Info"

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

- IP address of the control
- Switching device online
- Load motion configuration from PC to control.
- Compare PC motion configuration with parameter file
- Starting parameter editor of the control
- Axis status: Displays whether axes rotate. Click to open the "Device Status" dialog (see [chapter "Device status" on page 117](#))
- Warning: This field displays whether a warning is present. Click to open the "Error Memory" dialog (see [chapter "Error/diagnostic memory" on page 120](#))
- Error: This field displays whether an error is present. Click to open the "Error Memory" dialog (see [chapter "Error/diagnostic memory" on page 120](#))
- Clear error: Click to execute the "Clear Error" command in the control and in all connected Sercos drives

4.3 Control - Nodes

4.3.1 Overview on nodes

The nodes below the control result from their configuration:

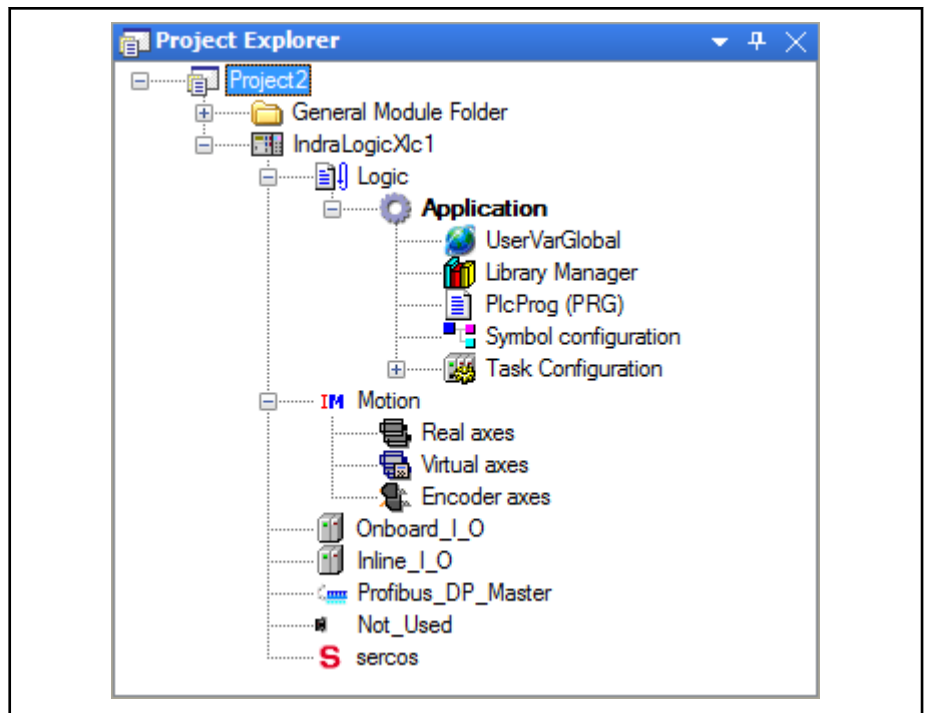


Fig. 4-3: Example: Node of the IndraLogic XLC L65 control

- [chapter 4.3.2 "Logic" on page 83](#), transition to PLC programming
- [chapter 4.3.3 "Motion" on page 87](#), basic principle of motion programming
- Options to connect and configure the input/output periphery
 - [chapter 4.3.4 "Onboard I/Os" on page 90](#), 1 byte fast inputs and outputs each*
 - [chapter 4.3.5 "Inline I/Os" on page 90](#), configurable Inline modules
- [chapter 4.3.6 "Profibus/M" on page 90](#), field bus to connect Inline modules, HMI devices, etc.
- [chapter 4.3.7 "Profinet I/O controller" on page 90](#), field bus to connect Profinet modules
- Function modules
- [chapter 4.3.9 "Sercos" on page 91](#), field bus to operate standard drives

* - only IndraLogic XLC L45/L65

4.3.2 Logic

"Logic" subfolder

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

Control

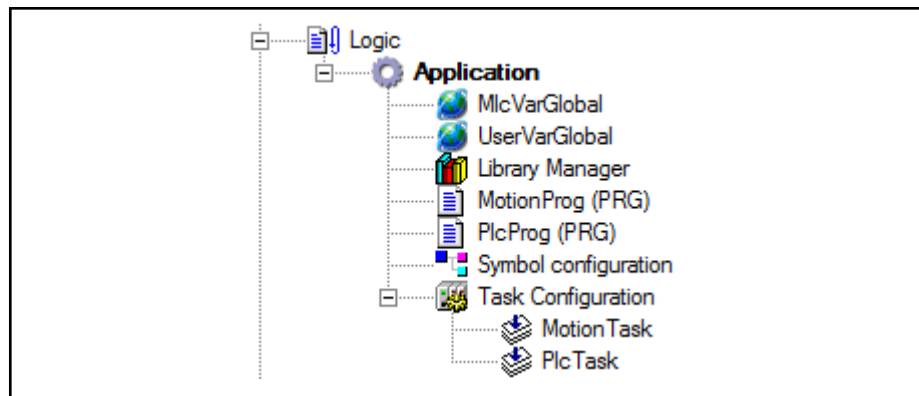


Fig. 4-4: "Logic" node and its subfolders

After the control was applied to the Project Explorer, two program files are automatically created below the "Application" folder and assigned to the two following 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 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 [chapter 5.3 "Library management" on page 171](#)).

The two preconfigured tasks are integrated into the overall task system of the control (see [chapter 5.4 "Task system" on page 177](#)).

"Logic" – Context menu

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

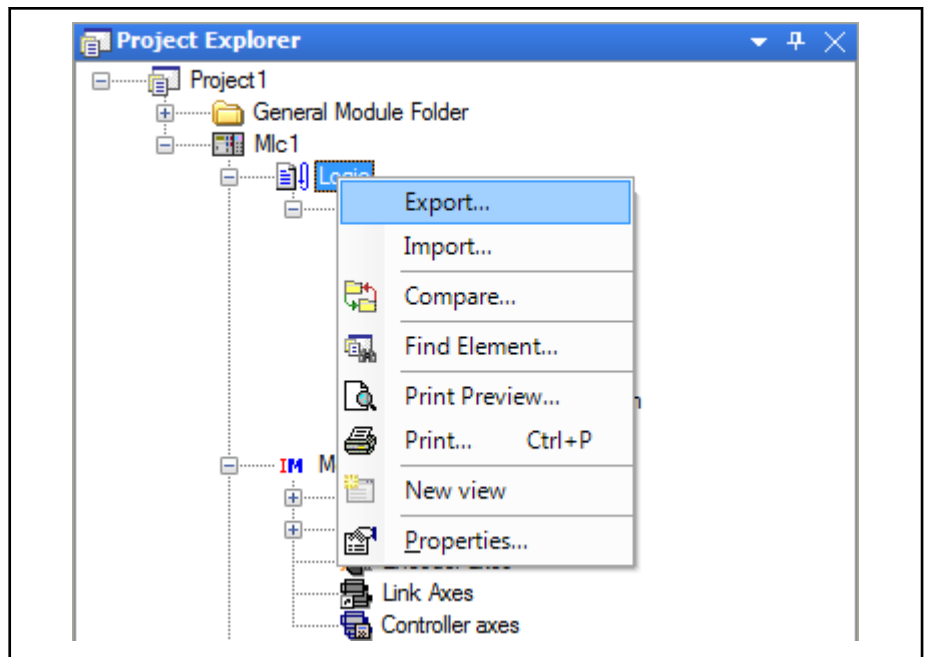


Fig. 4-5: Example: Context menu of the IndraMotion MLC "Logic" node

- **Export...** - Saves the "Logic" elements to a file
- **Import...** - Imports "Logic" elements from an export file on the PC. The "Logic" elements to be imported can be selected before the import
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging"](#) on page 151
- **Find Element...** - Searches 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 comprehensive, large volumes of paper can be generated! Use the print option of the individual PLC editors.

- **New View** - Displays the node in individual view
- **Properties...** - Displays and changes the properties. The following properties can be modified:
 - Exclude from build
 - External implementation
 - Enable system call
 - Specify compiler definitions

Application – Toolbar



Fig. 4-6: IndraLogic toolbar

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

- **Build Application** - Compiles the currently active PLC application

Control

- **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 the 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 context menu "Set Active Application"

- **Stop** - Stops PLC

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 context menu "Set Active Application"

Application – Context menu

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

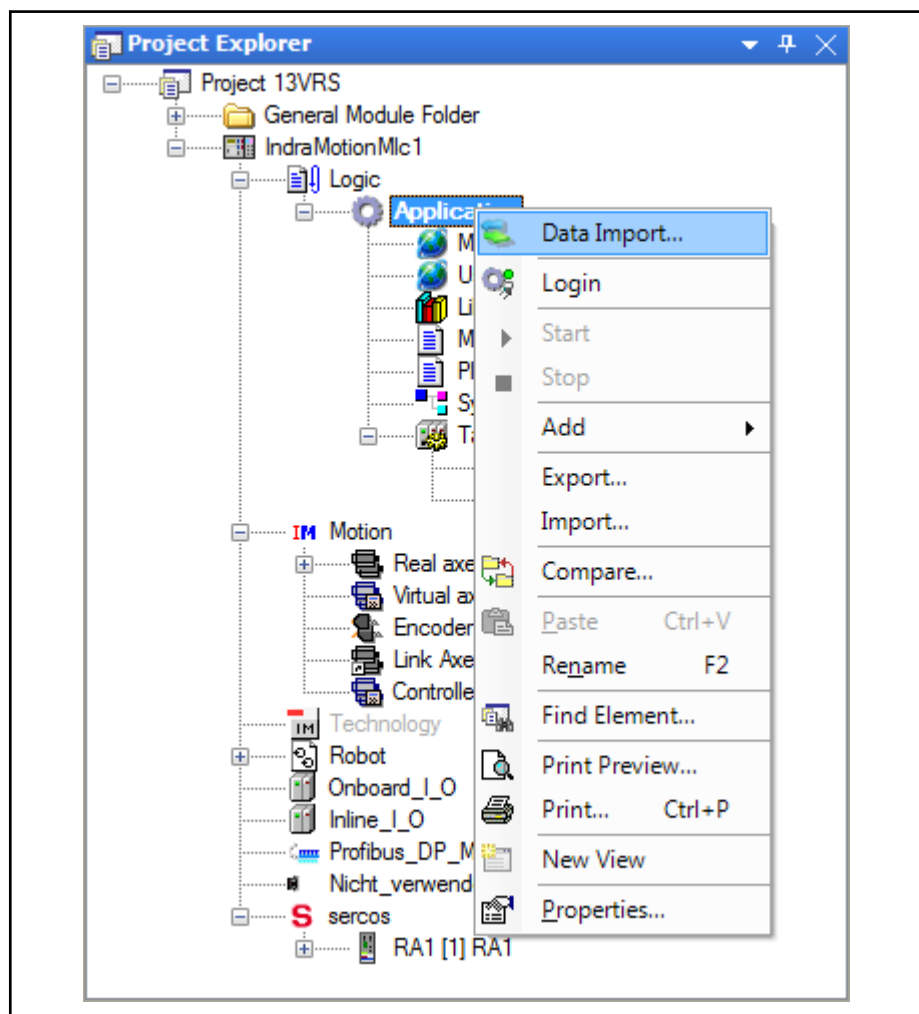


Fig. 4-7: Context menu of the "Application" subfolder

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

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



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

- **Start** - Starts the 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 context menu "Set Active Application"
- **Stop** - Stops the PLC
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 context menu "Set Active Application"
- **Add** - Adds objects, POU's, folders, etc.
- **Export...** - Saves "Application" elements to a file
- **Import...** - Imports "Application" elements from an export file on the PC. The "Application" elements to be imported can be selected before the import
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging" on page 151](#)
- **Rename** - Renames the node
- **Find Element...** - Searches elements below the node
- **Print Preview...** - Preview of the "Application" components selected for printing
- **Print...** - Prints the selected "Application" components
- **Add** - Adds PLC-specific objects
- **New View** - Displays the node in individual view
- **Properties...** - Displays and changes properties

4.3.3 Motion

General information



The "Motion" node is only available on controls with Sercos interface.

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

- Real axes based on IndraDrive, SercosDrive or EcoDriveCS
- 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

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

Control

At first, the folders are empty.

A real axis is created via **drag&drop** or **copy&paste** of the respective library element (see also [chapter 6.2.3 "Creating a real axis" on page 213](#)).

A virtual axis is created via the "Add" context menu item (see also [chapter 6.3.2 "Creating a virtual axis \(virtual master axis\)" on page 222](#)).

An encoder axis is created via the "Add" context menu item (see also [chapter 6.4.2 "Creating an encoder axis \(real master axis\)" on page 228](#)).

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

- Real axis - [chapter 6.2.4 "Real axis – Dialogs" on page 216](#)
- Virtual axis - [chapter 6.3.3 "Virtual axis - Dialogs" on page 225](#)
- Encoder axis - [chapter 6.4.3 "Encoder axis - Dialogs" on page 232](#)

In **online mode**, the operating state of the "Motion" calculation is displayed at the "Motion" node. Currently, the operating state 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")
- **"Checking"**: Parameterization verification active. If the parameterization is incorrect, the Motion calculation can possibly not be started (respective Sercos phase "P3")
- **"Running"**: Operating mode: Motion calculation active (respective Sercos phase "P4")

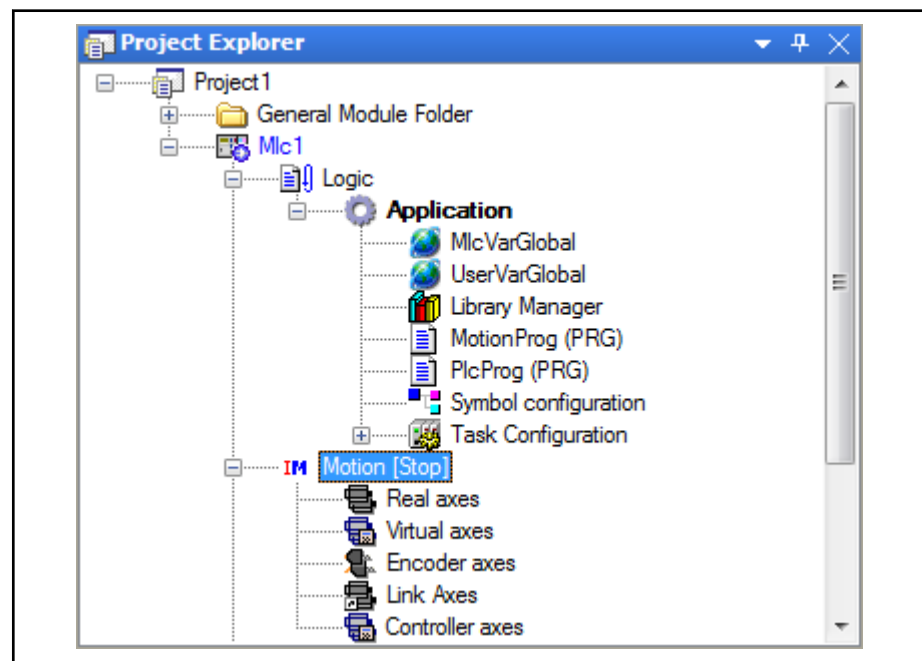


Fig. 4-8: "Motion" node in online mode:

Motion – Context menu

For commands on the complete Motion functionality (that is on all axes), go to the context menu of the "Motion" node.

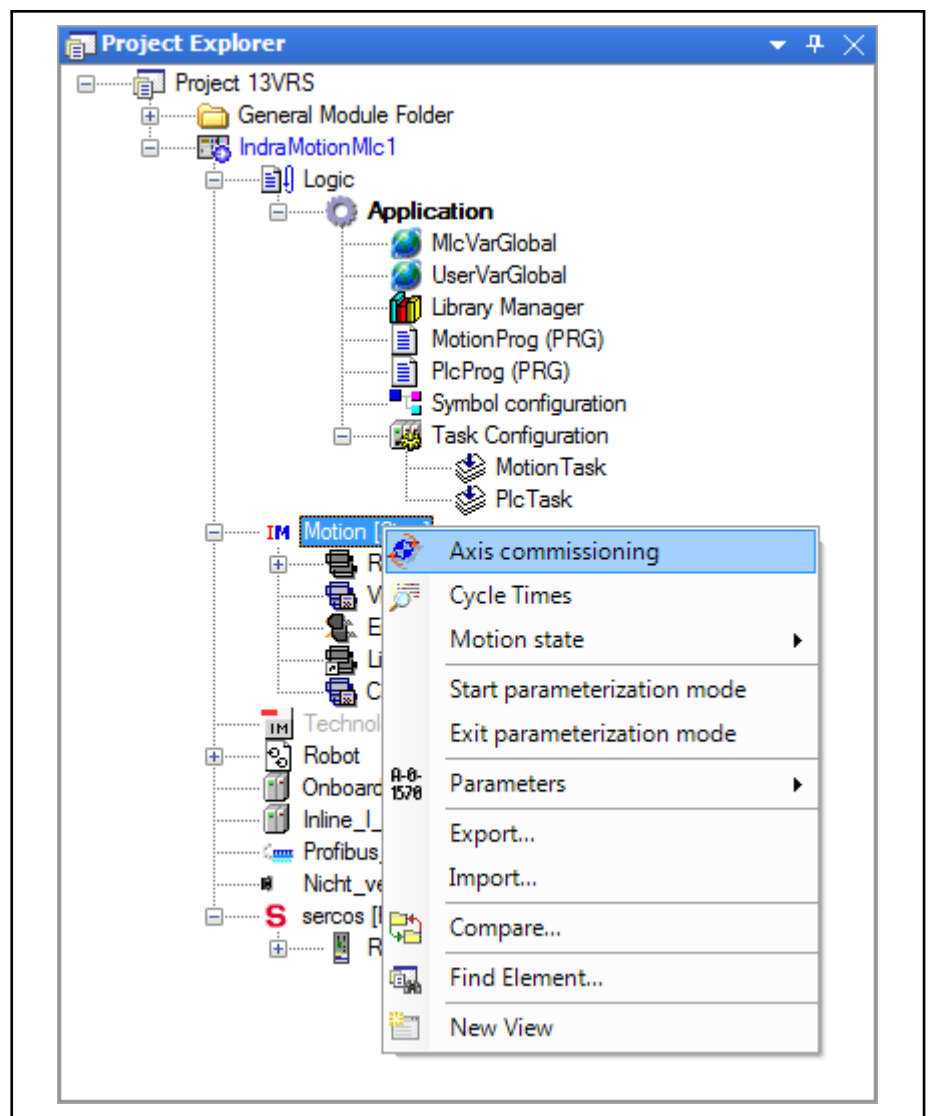


Fig. 4-9: Context menu of the Motion node

Axis commissioning

Use this command to move axes without creating a PLC program before (see [chapter 7.11 "Axis commissioning" on page 359](#)).

Cycle times

Use this command to see the current behavior of the task management and to set its timing (see [chapter 7.2 "Cycle Times" on page 323](#)).

Motion state

Use the following commands to control the operating state of the Motion calculation:

- **"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"

Control

Each axis creates its own environment:

- For the contents of the context menus of the different axis types, refer to:
 - [chapter 6.2.5 "Real axis – Context menu" on page 219](#)
 - [chapter 6.3.4 "Virtual axis - Context menu" on page 226](#)
 - [chapter 6.4.4 "Encoder axis - Context menu" on page 232](#)

To go to the individual software tools, use the context menus and dialogs.

Start parameterization mode

Use this command to switch all axes to the parameterization mode. The Sercos bus remains in phase 4. The parameterization mode allows to reparameterize axes while the PLC is running and still controls all Sercos I/O devices. For more information on the parameterization mode, refer to [chapter 7.7.5 "Axis in parameterization mode" on page 344](#). If one of the axes cannot be switched to the parameterization mode, the "Error/diagnostic memory" dialog is displayed (see [chapter "Error/diagnostic memory" on page 120](#)).

Exit parameterization mode

Use this command to exit the parameterization mode of all axes currently in the parameterization mode. This command can be started in the phases 2 and 4. For more information on the parameterization mode, refer to [chapter 7.7.5 "Axis in parameterization mode" on page 344](#). If the parameterization mode cannot be exited for one of the axis, the "Invalid Parameters" dialog is displayed (see [chapter "Invalid parameters" on page 123](#)).

4.3.4 Onboard I/Os



This node is only available for an IndraLogic XLC L45/L65.

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

- Onboard - outputs (also refer to [chapter 3.9.1 "Creating Onboard I/Os" on page 54](#))

The individual options are described in detail in "Rexroth IndraWorks 13VRS Field Buses", chapter "Configuring Onboard I/Os (dialogs)" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

4.3.5 Inline I/Os

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

- Inline modules (refer to [chapter 3.9.2 "Creating Inline I/Os" on page 56](#))

The individual options are described in detail in "Rexroth IndraWorks 13VRS Field Buses", chapter "Configuring Inline I/Os (dialogs)" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

4.3.6 Profibus/M

The individual options are described in detail in [chapter 3.9.3 "Creating Profibus I/Os" on page 63](#)

4.3.7 Profinet I/O controller

If a control is created, but no controller is selected for the Realtime Ethernet 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.

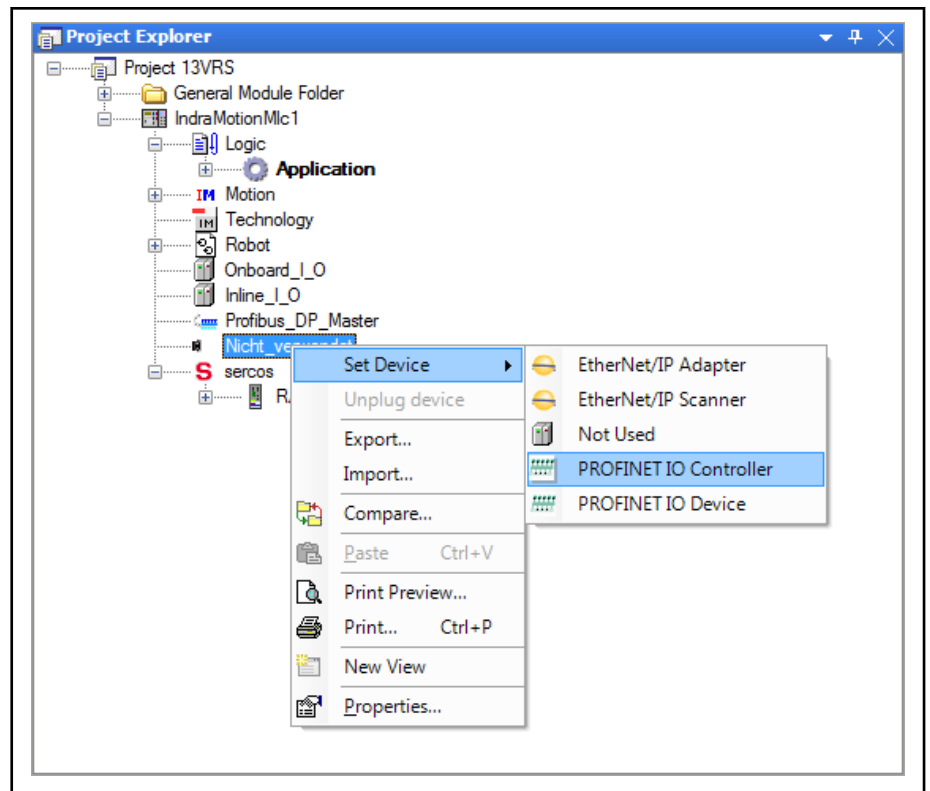


Fig. 4-10: Creating Profinet I/O controller

The individual options are described in detail in [chapter 3.9.4 "Creating a Profinet I/O Controller"](#) on page 63

4.3.8 Function modules

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

The individual function modules and their configurations are described in detail in "Rexroth IndraControl Lxx 13VRS Function Modules" (see [chapter 1.4 "Required and supplementing documentations"](#) on page 18).

4.3.9 Sercos

General information

The Sercos node is only available if the control is provided with a Sercos interface. If it is still not required, the node can be labeled as "Not Used".

This node represents the Sercos III master communication.

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

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 an example, see [chapter 6.2.3 "Creating a real axis"](#) on page 213). 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:

Control

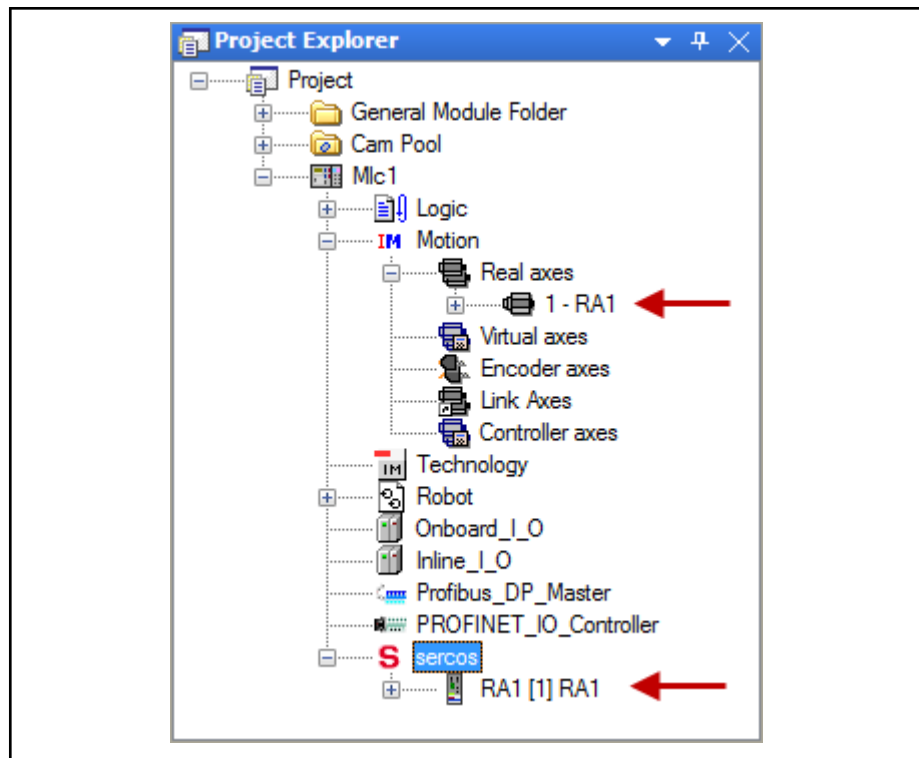


Fig. 4-11: Creating a real axis RA1 - Overview in the Project Explorer



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 user settings made in the drive dialogs 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

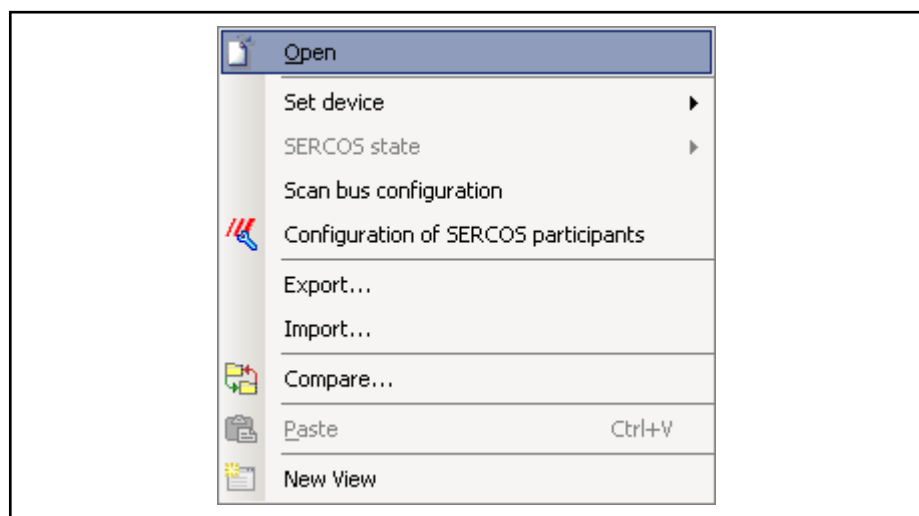


Fig. 4-12: Context menu of the Sercos node

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

- [chapter "Setting device" on page 93](#)
- [chapter "Sercos state" on page 93](#)
- [chapter "Scanning bus configuration" on page 93](#)
- [chapter "Configuring Sercos devices" on page 93](#)

Setting device

The menu item "Set Device" can enable and disable the Sercos interface of IndraLogic XLC controls with Sercos. The interface can have two configurations:

- Sercos master
- Not used

If the interface is configured as "Not Used", all configured Sercos devices are deleted from the configuration.



Changes in this configuration require the download of the PLC program and the Motion configuration. Additionally, a restart is required.

Sercos state

The "Sercos state" menu item allows switching to the target phase of the control (see also [chapter 7.3 "Motion Mode" on page 324](#)).

Scanning bus configuration

The "Scan Bus Configuration" menu item determines the devices connected to the Sercos bus (see also [chapter "Scanning bus configuration" on page 333](#)).

Configuring Sercos devices

The "Configuration of Sercos Devices" menu item opens a display of all Sercos devices (see also [chapter "Adjustment of command/actual configuration" on page 329](#)).

Sercos - Toolbar



This toolbar is only available for the variants with Sercos interface.

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



Fig. 4-13: Sercos toolbar

- Display of the current phase ①
- Phase switching to P0 (only online) ②
- Phase switching to P2 (only online) ③
- Phase switching to P4 (only online) ④
- Open the dialog "Configuration of Sercos Devices" (always available) ⑤

Control

- Start of the "Configuration Drive Bus" dialog (only online if a Sercos II master communication is configured) ©*
- * - only for IndraMotion MLC controls

4.4 Control – Context menu

4.4.1 Overview on menu items

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

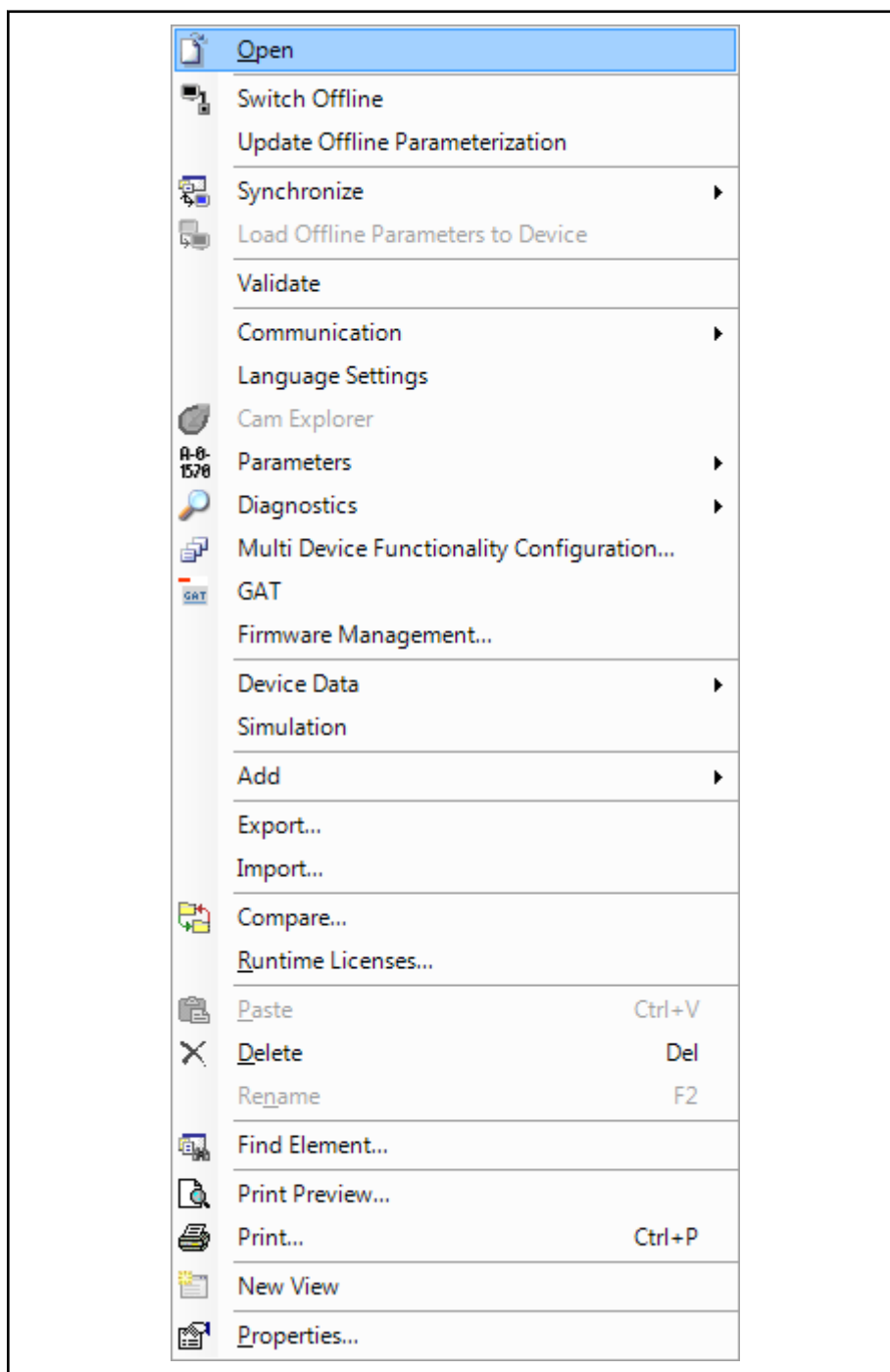


Fig. 4-14: Context menu of a control in online mode

- **Open**, see [chapter 4.4.2 "Device editor" on page 96](#)

Control

- **Switch Offline**, see [chapter 7.6 "Modes for project editing" on page 326](#)
- **Update Offline Parameterization**, see [chapter "Offline Parameterization" on page 328](#)
- **Load Offline Parameters to Device**, see [chapter 7.6.3 "Loading Offline Parameters to Device" on page 337](#)
- **Validate**, see [chapter 7.12 "Validating Motion configuration" on page 365](#)
- **Communication** - Communication parameter setting, refer to [chapter 4.4.3 "Communication" on page 100](#) or control IP communication setting or [chapter 3.10 "IP communication" on page 68](#)
- **Language Settings** - Language setting of the control, see [chapter 4.4.4 "Language settings" on page 103](#)
- **Cam Explorer**, See [chapter 4.4.5 "Cam explorer" on page 104](#)
- **Parameters**, see [chapter 4.4.6 "Parameters" on page 109](#)
- **Diagnostics** - Contains the subitems
 - [chapter "Device status" on page 117](#)
 - [chapter "Advanced properties" on page 119](#)
 - [chapter "Clearing errors" on page 120](#) - Deletes all control and drive errors
 - [chapter "Error/diagnostic memory" on page 120](#) - Provides an overview on the current diagnostic state of the control and its drives
 - [chapter "Invalid parameters" on page 123](#) - Displays the parameter "Diagnostics (C-0-0626)" and lists all invalid parameters
 - [chapter 5.4.5 "Task list/configuration" on page 186](#) - Displays the present tasks and their priorities
 - [chapter 5.4.6 "Task viewer" on page 188](#) - Displays the present tasks and their priorities
- **Multi-Device Configuration...** - Allows the use of an application and its parameters for several controls, see [chapter 4.4.8 "Multi-device" on page 126](#)
- **GAT - Generic Application Template** is a structured template to add and edit PLC program code (refer to GAT Generic Application Template)
- **Firmware Management...** - Loads and updates the control firmware, see [chapter 4.4.9 "Firmware management" on page 134](#)
- **Device Data*** - Saves and restores device data or files and folders from PC to control and vice versa, see [chapter 4.4.11 "Device data" on page 139](#)
- **Simulation** - Simulates the control without real hardware, see [chapter 4.4.12 "Simulation" on page 147](#)
- **Export...** - Saves elements from the current project to a file [chapter 4.4.13 "Exporting" on page 148](#)
- **Add** Adds new elements below the control
- **Import...** - Imports elements from a file into the current project, see [chapter 4.4.14 "Importing" on page 149](#)
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging" on page 151](#)

Control

- **Runtime Licenses** - This dialog outputs all licenses used by the PLC program at runtime, see [chapter 4.4.16 "Runtime Licenses" on page 153](#)
- **Delete** - Deletes the control and its subfolders. **Only available offline**
- **Rename** - Renames the control. **Only available offline**
- **Find Element...** - Searches for elements in the project
- **Print Preview...** - Provides a preview of project and device data, see [chapter 4.4.17 "Print preview" on page 153](#)
- **Print...** - Provides a printout of the current project and device data, see [chapter 4.4.18 "Printing" on page 155](#)
- **New View** - Displays node in individual view
- **Properties...** - Displays and modifies control properties [chapter 4.4.19 "Control properties" on page 157](#)

4.4.2 Device editor

Device editor - General information

The device editor provides dialogs to configure a control managed in the Project Explorer.

By default, double-click on the control to open the **Devices** 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:

- [chapter "Applications " on page 96](#): List of the applications on the control
- [chapter "Log" on page 97](#): Displays the PLC log file
- [chapter "PLC settings" on page 98](#)
- [chapter "Information " on page 99](#): General information on the device (name, vendor, version, etc.)

Applications

This tab displays and, if required, deletes applications currently located on the control.

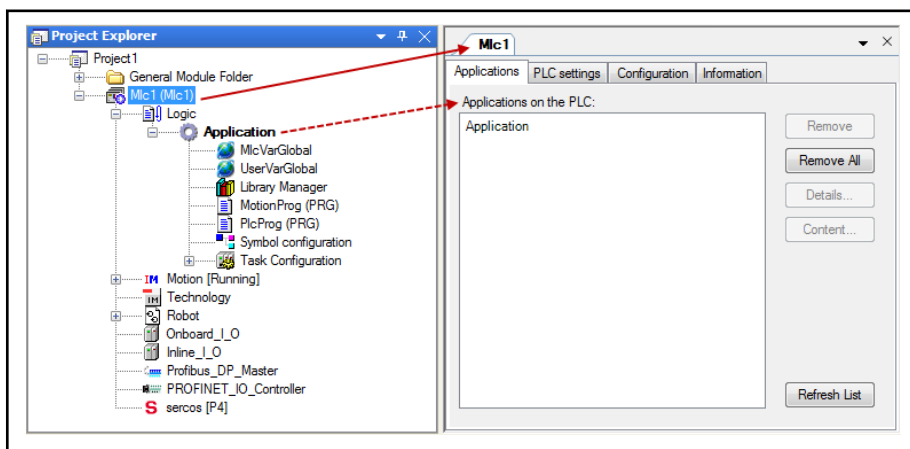


Fig. 4-15: Device dialog, Application

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

Control

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 to those available in the project. If the list does 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 to those in the function blocks with the same name in the application 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

The log (logger) is switched off by default. Enable the log via **Tools ► Options** in the IndraWorks main menu (refer to the following figure).

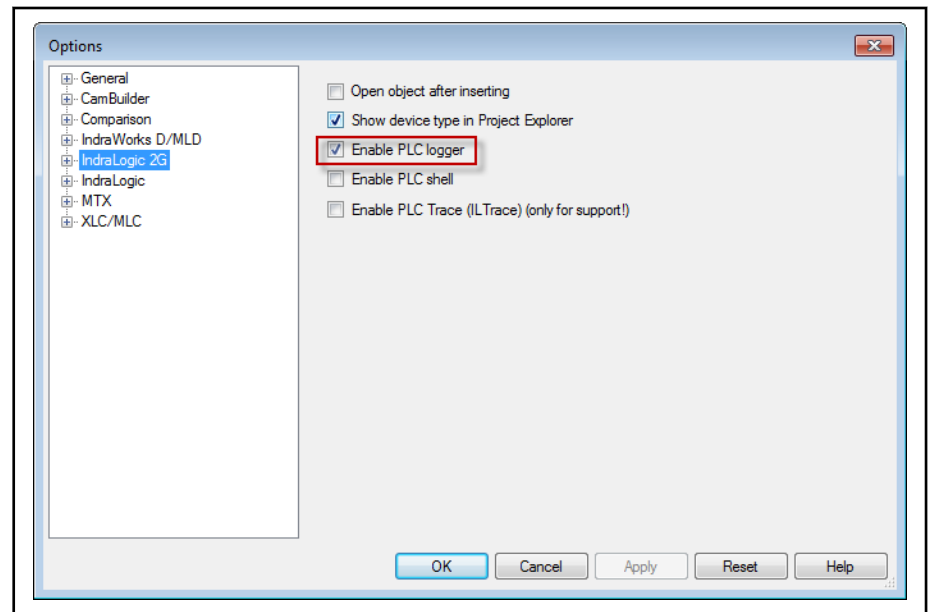


Fig. 4-16: Enable PLC logger

The events of the control are recorded via the log. These are:

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

Control

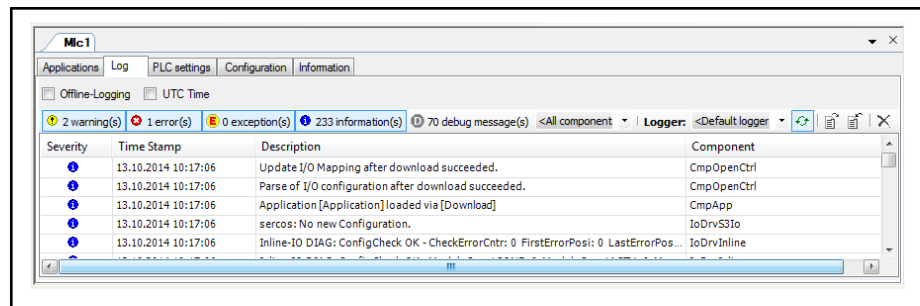


Fig. 4-17: Device editor, log

A log 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 shown or hidden via the corresponding button in the bar above the list. The respective number of log entries is displayed on the button for the respective category.

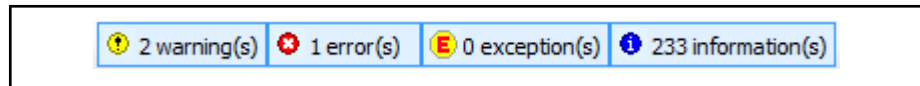


Fig. 4-18: Severity of event

Time Stamp: Date and time, e.g. "16.11.2010 14:00".

Description: Describing the event, e.g. "Sercos III I/O: SetParamP4 OK".

Component: Name of the respective component.

Com Name (component name): An individual component can be selected in a drop-down list to display only related log 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 via .

The content of the list can be exported to an XML file. Press  to open the standard dialog to save a file. The file filter is set to "xml-files" (*.xml)". The log list is stored in the selected directory with the file name specified 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 (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  icon.

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



For further diagnostic options, refer to [chapter 4.4.7 "Diagnostics" on page 117](#).

PLC settings

This tab specifies the control behavior in the "Stop" state.

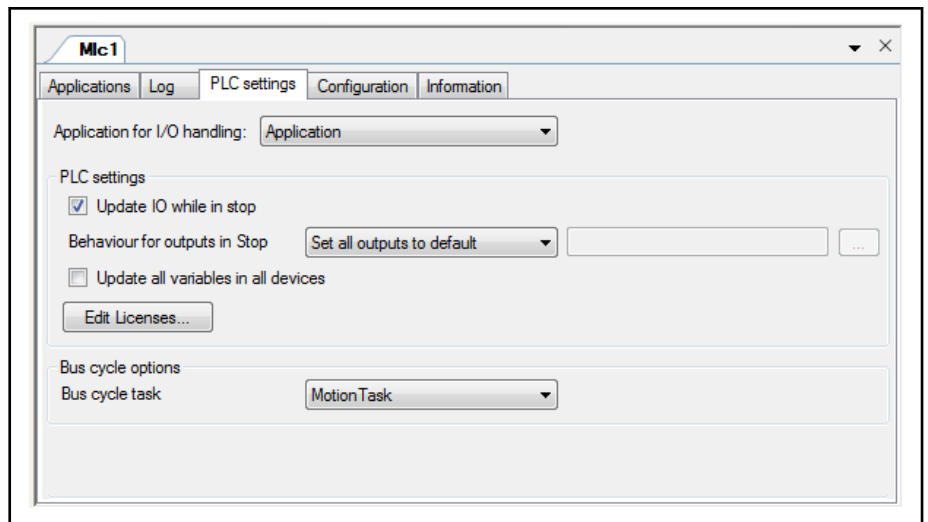


Fig. 4-19: Device editor, PLC settings

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

PLC settings: **Update I/O 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 the "Stop" state:

- **Retain 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 and it is executed when the control goes into the "Stop" state. Click on ... 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 drop-down list provides all tasks defined in the task configuration of the active application (e.g. "MotionTask", "PlcTask" etc.).



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

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. This setting applies to all I/O devices. However, this can be overwritten locally by every I/O device (see [chapter "Profibus, Profinet and Inline I/Os" on page 191](#)).

Information

This tab of the **Devices** dialog displays some general information from the device description file on the currently selected device in the Project Explorer.

The following information is displayed:

- Name
- Vendor
- Categories

Control

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

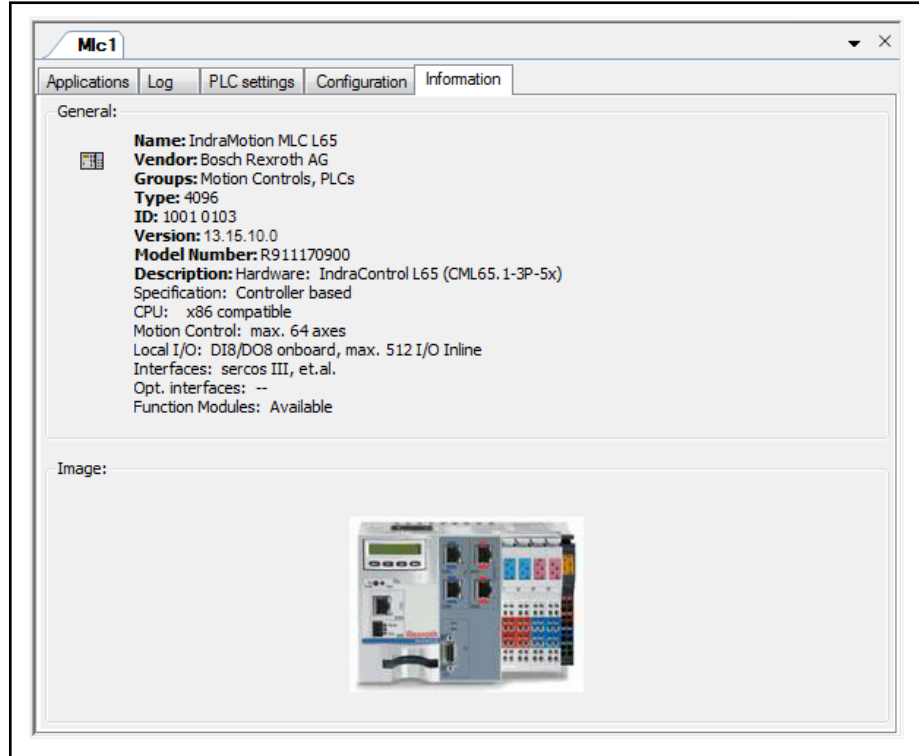


Fig. 4-20: Device editor, Information

4.4.3 Communication

TCP settings

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

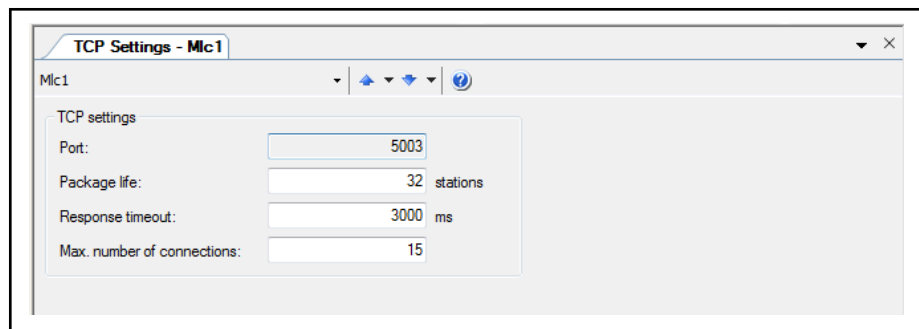


Fig. 4-21: TCP settings

Port: Specifying the port for the TCP communication.

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

Response timeout: "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 is aborted.

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

Use this menu item to make the settings for the IP communication. For a detailed description, refer to [chapter 3.10 "IP communication" on page 68](#).

User control ID

Overview

A user control ID can be written to the control. If a control is provided with this ID, enter it when connecting for the first time (login, switch online,...) to an IndraWorks project session.

Example

Example: A machine is provided with two controls: S1 and S2. The control S1 is provided with the (random) user control ID S1 and the control S2 is also provided with the (random) ID S2. Both controls can also obtain almost any other ID. When logging into the control S1 with IndraWorks, IndraWorks prompts to enter the control ID into a dialog. If the ID is correct, it is logged in. If the ID is incorrect, the complete action is canceled. If the entered ID is valid, it is not prompted anymore by IndraWorks at next login or when switching online within a running IndraWorks project session. Only after closing and opening the IndraWorks project again, entering the ID upon the next connection establishment is required.



Fig. 4-22: Dialog to enter the user control ID requested by IndraWorks (in this example, it is the ID "S2")

Important

Control

It is important that the user control ID is only saved on the control and that it is not confidential. The ID is not part of the IndraWorks project. Thus, it is not project property and not under version control. The user control ID must not be confused (with a password for example). The user control ID is used to impede a connection in IndraWorks such as, for example, a login with a wrong control. Therefore, IndraWorks prompts to enter the user control ID before establishing the connection.

This is only possible if there is a user control ID on the control. A brief explanation on how to create, rename and delete a user control ID is given in the following.

Managing user control ID

Creating an ID

To create a user control ID on a control, establish a connection with IndraWorks. Thus, either switch online or log in. Subsequently, open the dialog "Manage User Control ID" in the context menu of the control (right-click on the "Control" node) and go to the menu item "Communication".

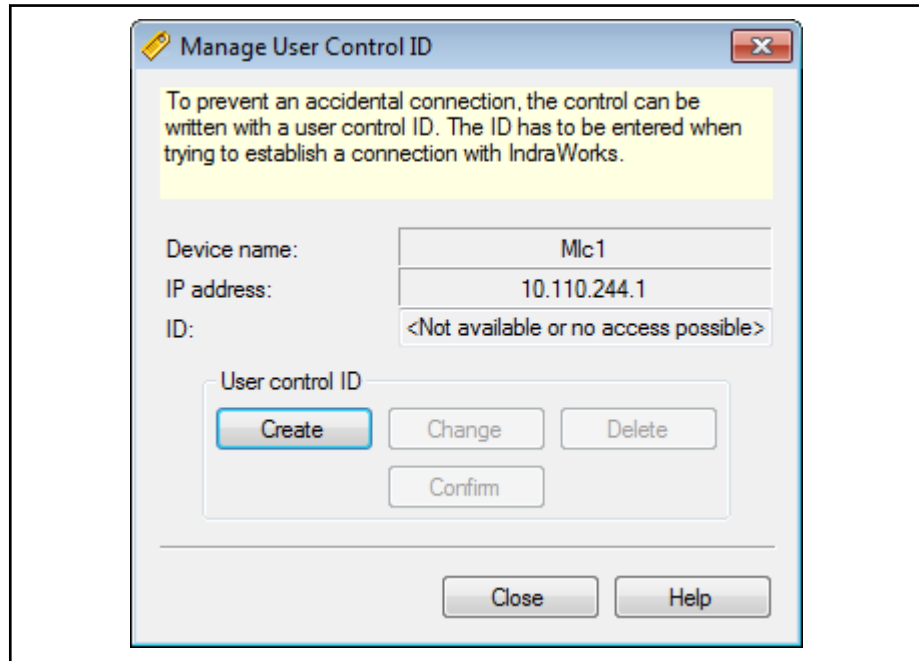


Fig. 4-23: Dialog "Generate User Control ID"

Click on "Create" in the dialog "Manage User ID Control" to open the dialog to generate an ID.

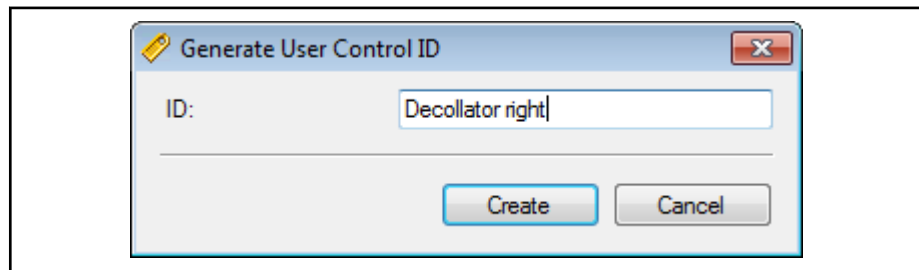


Fig. 4-24: Dialog to enter the user control ID to be generated

Enter an ID for the current control in the dialog "Generate User Control ID". Click on **Create** to write the entered ID to the control.

As IndraWorks is already connected to the control and the ID was just entered, the ID is not prompted anymore during the current IndraWorks project

session after closing the connection to the control and establishing it again. Only after closing and opening the IndraWorks project again, IndraWorks prompts to enter the user ID when the next connection is established.

Rename or delete ID

If a control is provided with a user control ID, the dialog "Manage User Control ID" can be used to rename or delete the ID.

Exchanging hardware

When exchanging the control, and the CF card of the old control is used in the new one, IndraWorks detects that CF card when establishing a connection between IndraWorks and the new control. This also applies if the content of the CF card of the old control is copied to the CF card of the new control. IndraWorks provides that information in the following dialog:

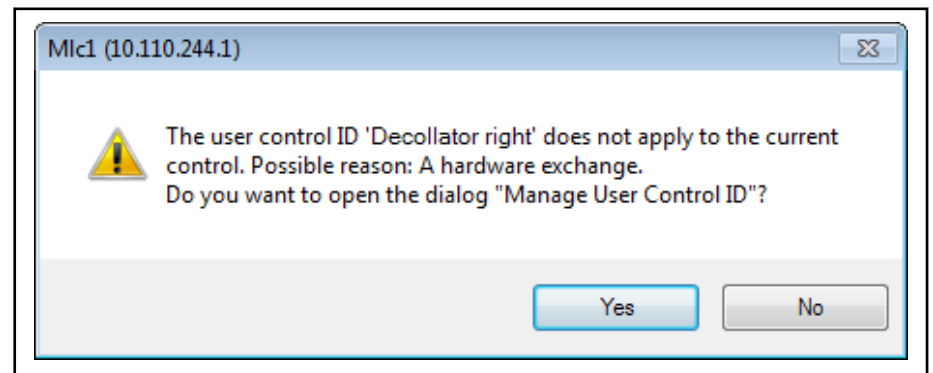


Fig. 4-25: Information after detecting an invalid control ID

Click on **Yes** and the "Manage User Control ID" dialog opens. Use this dialog to validate the user control ID for the new control or to delete the ID on the new control.

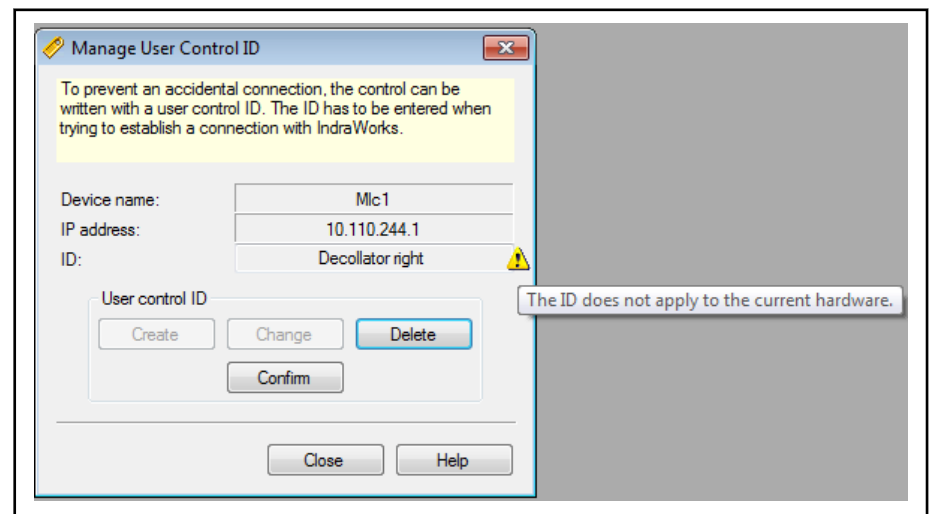


Fig. 4-26: Dialog "Manage User Control ID"

If the user control ID does not apply for a syntactically correct identifier (here: "Decollator right") for the current control, the ID including the identifier can be validated or deleted.

4.4.4 Language settings

- Call** Context menu of the control: **Language settings**.
- Target** Use this dialog to set the language of the control and of all connected devices (IndraDrive/HydraulicDrive).

Control

- 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, click on "Rewrite" to apply the language again.

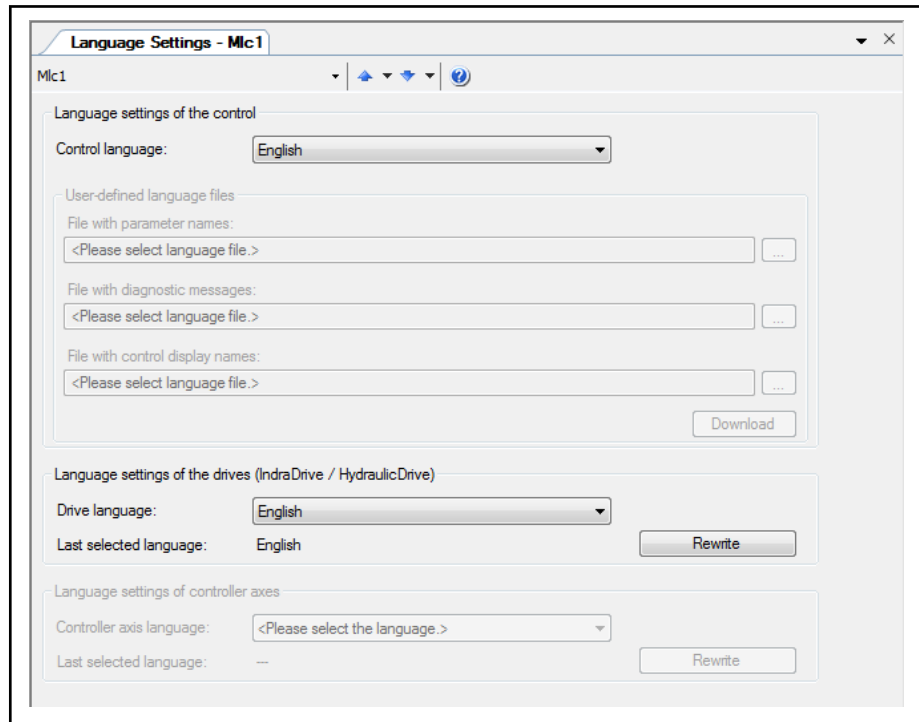


Fig. 4-27: Language settings

For more information on the language setting, refer to [chapter 11 "User-defined languages for parameter names, units, diagnostics and menu texts"](#) on page 443.

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

4.4.5 Cam explorer

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

- Cam Pool
- Compact Flash
- Control

The "Cam Explorer" can copy cams to each of these sections (via drag&drop or arrow keys) and retrieve them.

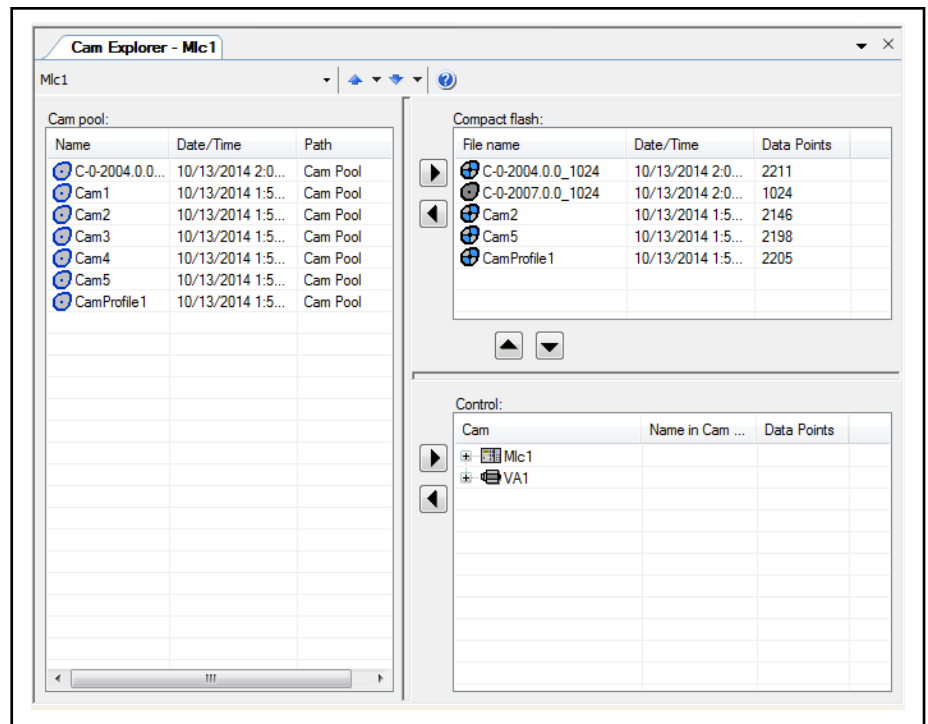


Fig. 4-28: "Cam Explorer" dialog:

The cam stock is displayed in the left of the dialog below "Cam Pool". In general, they can be generated with the CamBuilder. The cams can also be retrieved from the CF card, control or an axis.



To use the CamBuilder, a license is required. The license 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 applying a cam to 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 listed parameters that only contain the points. Retrieving 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 retrieved 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 modify existing segments at runtime (e.g. program-controlled fine adjustment). When modifying cam tables, the complete cam table has to be applied.

In the dialog on the upper right, 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

Control

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. MotionProfiles 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 interpolated in the control. This 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 to the other. There are three options:

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



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

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

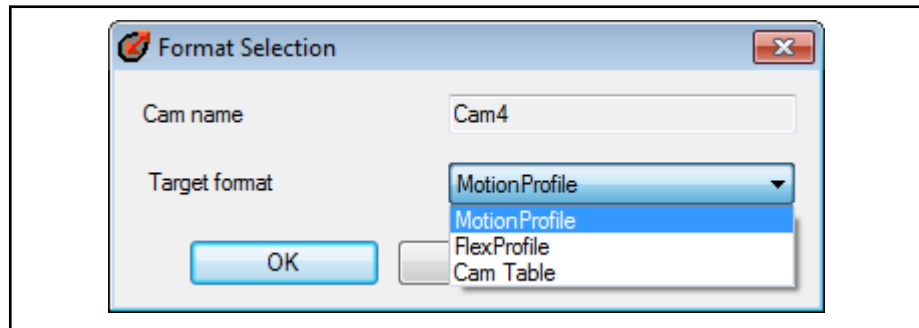


Fig. 4-29: Selecting the target format on the CF card

If the target format is a cam table, it is always converted into a cam table. Another dialog, displaying the number of data points and the file names on the CF card, is shown.

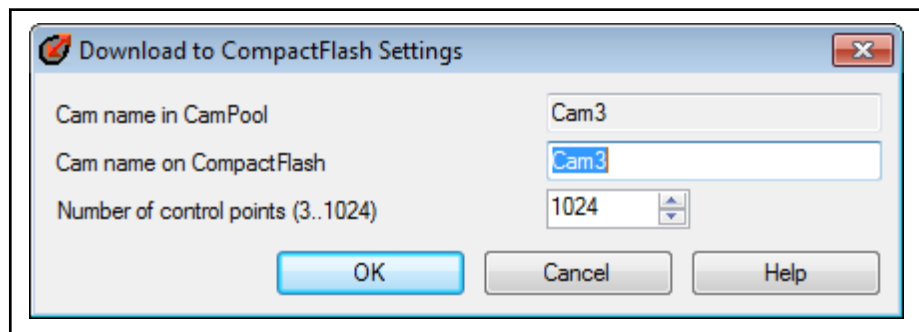


Fig. 4-30: Specifying name and number of data points

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

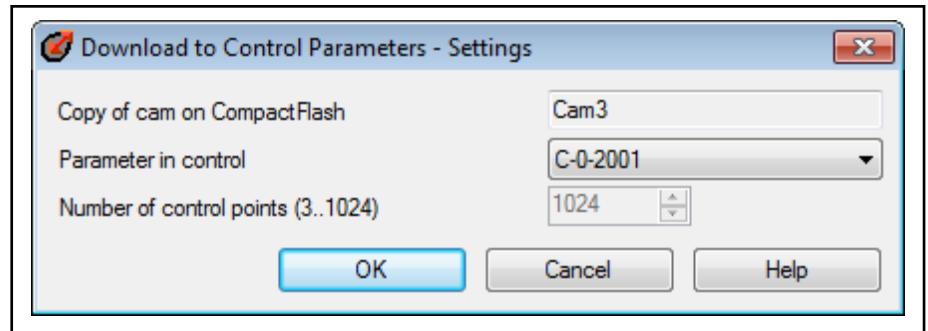


Fig. 4-31: Settings to copy a cam table from the CF card to the control



Cam tables copied to the CF card are stored so that they can be copied to 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 CF card to copy them into the parameters involved with reasonable effort.

Cam tables stored on the CF 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 has to be queried. The following window is opened for selection:

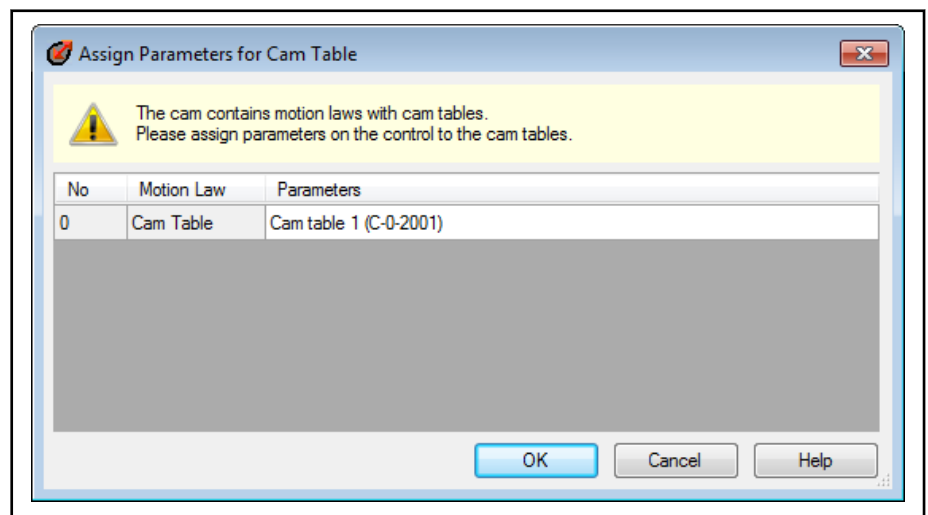


Fig. 4-32: Selecting cam table parameter while downloading a MotionProfile or FlexProfile to an axis

Parameters involved:

Cam tables

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

MotionProfile

Common parameters for both sets

- A-0-2900, MotionProfile, master axis switching position
- A-0-2901, MotionProfile, motion step 1, initial slave axis position

Control

- A-0-2904, MotionProfile, basic configuration
- A-0-2930, MotionProfile, control word
- A-0-2931, MotionProfile, status word

MotionProfile 0-parameters with parameter names	MotionProfile 1
A-0-2902, MotionProfile 0, diagnostics	A-0-2903
A-0-2910, MotionProfile 0, number of motion steps	A-0-2920
A-0-2911, MotionProfile 0, master axis velocity	A-0-2921
A-0-2912, MotionProfile 0, list of initial master axis positions	A-0-2922
A-0-2913, MotionProfile 0, list of motion laws	A-0-2923
A-0-2914, MotionProfile 0, list of distances	A-0-2924
A-0-2915, MotionProfile 0, list of slave axis velocities	A-0-2925

FlexProfile *Common parameters for all four sets*

- 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 command position 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.

FlexProfile 0-parameters with parameter names	FlexProfile 1	FlexProfile 2	FlexProfile 3
A-0-3020, FlexProfile 0, master axis velocity	A-0-3050	A-0-3080	A-0-3110
A-0-3021, FlexProfile 0, list of master axis sources	A-0-3051	A-0-3081	A-0-3111
A-0-3022, FlexProfile 0, list of slave axis distances	A-0-3052	A-0-3082	A-0-3112
A-0-3023, FlexProfile 0, list of master axis ranges	A-0-3053	A-0-3083	A-0-3113
A-0-3024, FlexProfile 0, list of master axis ranges, units	A-0-3054	A-0-3084	A-0-3114
A-0-3025, FlexProfile 0, list of motion laws	A-0-3055	A-0-3085	A-0-3115
A-0-3026, FlexProfile 0, list of motion step types	A-0-3056	A-0-3086	A-0-3116
A-0-3027, FlexProfile 0, list of slave axis starting velocities	A-0-3057	A-0-3087	A-0-3117
A-0-3028, FlexProfile 0, list of slave axis starting accelerations	A-0-3058	A-0-3088	A-0-3118
A-0-3029, FlexProfile 0, list of slave axis starting jerks	A-0-3059	A-0-3089	A-0-3119
A-0-3030, FlexProfile 0, list of slave axis end velocities	A-0-3060	A-0-3090	A-0-3120
A-0-3031, FlexProfile 0, list of slave axis end accelerations	A-0-3061	A-0-3091	A-0-3121
A-0-3032, FlexProfile 0, list of slave axis end jerks	A-0-3062	A-0-3092	A-0-3122
A-0-3033, FlexProfile 0, list of slave axis travel velocities	A-0-3063	A-0-3093	A-0-3123
A-0-3034, FlexProfile 0, list of slave axis travel accelerations	A-0-3064	A-0-3094	A-0-3124

FlexProfile 0-parameters with parameter names	FlexProfile 1	FlexProfile 2	FlexProfile 3
A-0-3035, FlexProfile 0, list of slave axis travel jerks	A-0-3065	A-0-3095	A-0-3125
A-0-3036, FlexProfile 0, list of turning point displacements	A-0-3066	A-0-3096	A-0-3126
A-0-3040, FlexProfile 0, events, list of motion step assignments	A-0-3070	A-0-3100	A-0-3130
A-0-3041, FlexProfile 0, events, list of trigger modes	A-0-3071	A-0-3101	A-0-3131
A-0-3042, FlexProfile 0, events, list of action modes	A-0-3072	A-0-3102	A-0-3132
A-0-3043, FlexProfile 0, events, list of trigger values	A-0-3073	A-0-3103	A-0-3133
A-0-3044, FlexProfile 0, events, list of trigger values, units	A-0-3074	A-0-3104	A-0-3134
A-0-3046, FlexProfile 0, command: Check and apply event	A-0-3076	A-0-3106	A-0-3136
A-0-3047, FlexProfile 0, diagnostics	A-0-3077	A-0-3107	A-0-3137
A-0-3048, FlexProfile 0, command: Check and apply profile	A-0-3078	A-0-3108	A-0-3138
A-0-3049, FlexProfile 0, checksum	A-0-3079	A-0-3109	A-0-3139

4.4.6 Parameters

Parameter editor

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

Availability

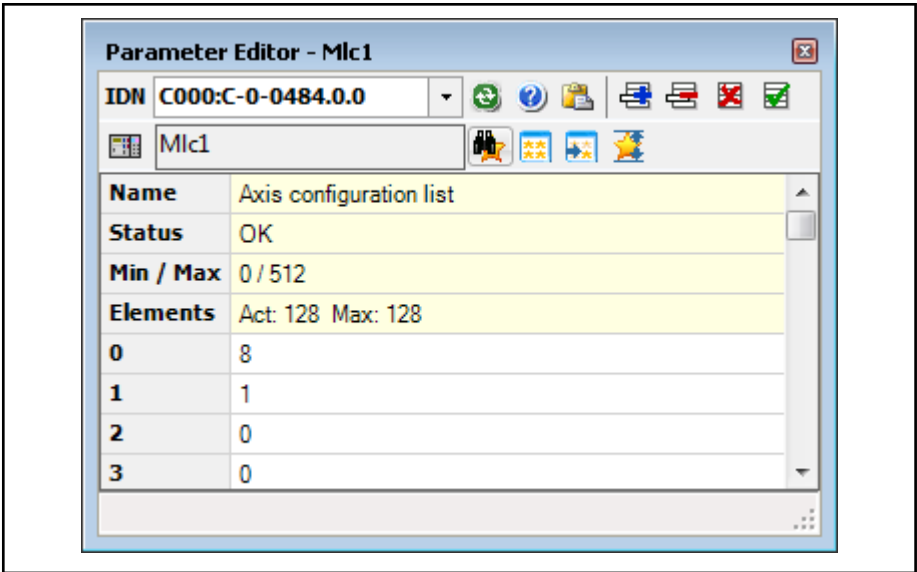
This function is enabled in the following modes:

- Online mode - Access to the real parameters of the control, axes or drives
- Offline mode - Access to the parameters of the control, axes or drives

Structure

Open the parameter editor in the context menu of control, axes and drives and right-click on a parameter in any dialog:

Control



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. 4-33: 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 retrieved 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 applied with <ENTER>.

Toolbar of the parameter editor

For the following functionalities, go to the toolbar above the tabular parameter display:

Icon	Description
	Opens the "Parameter selection list" on page 111
	Opens the parameter group
	Applies the parameters to the parameter group
	Adjusts window size to content
	New loading, updates the display

Icon	Description
	The parameter help opens the .chm help file
	Copies the parameter values to the clipboard
	Inserts a list element into 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 to the corresponding parameter

Tab. 4-1: 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).

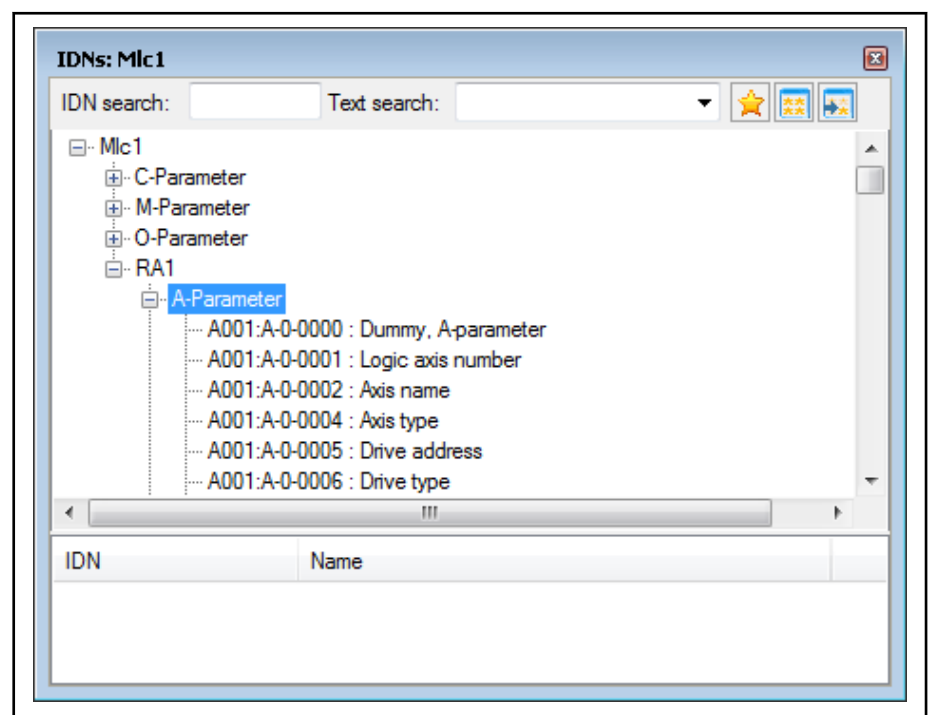


Fig. 4-34: Opening the parameter list in the parameter editor

Parameter group

Based on the individual parameter displayed in the [chapter "Parameter editor" on page 109](#), the user can compose a list of parameters whose values can be evaluated and changed.

Control

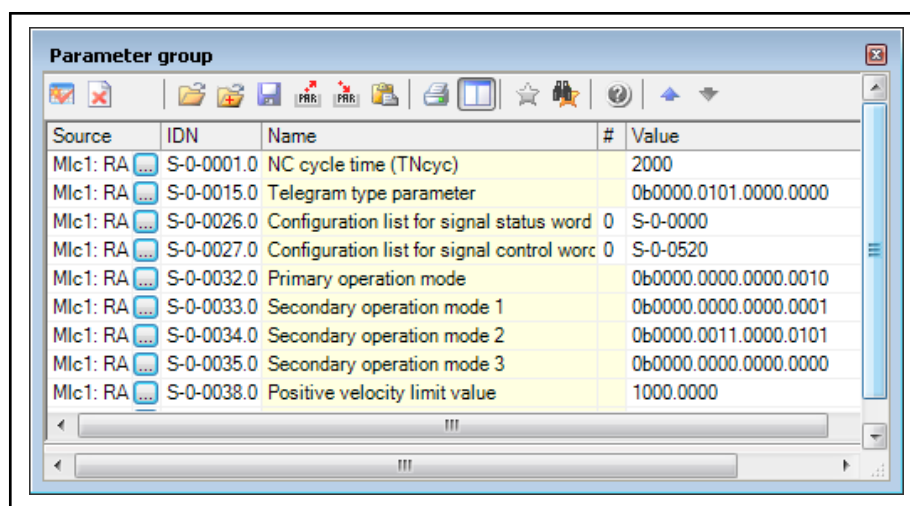


Fig. 4-35: Parameter group

Availability

This function is enabled in the following modes:

- Online mode - Access to the real parameters of the control, axes or drives
- Offline mode - Access to the parameters of the control, axes or drives

Toolbar of the parameter group

For the following functionalities, go to the toolbar above the tabular parameter display:

Icon	Description
	Inserts a new line to enter parameters (selection dialog)
	Deletes the current line
	Loads the parameter list
	Saves the parameter list
	Imports a parameter file (*.par)
	Exports a parameter file (*.par)
	Copies the content to the clipboard as text
	Prints the parameter group
	Displays the parameter source
	Open the parameter editor (also refer to chapter "Parameter editor" on page 109)
	Parameter selection (selection dialog), see also chapter "Parameter editor" on page 109

Icon	Description
	Parameter description, opens the help file
	Moves the line up/down

Tab. 4-2: Toolbar of the parameter group

Finding parameters

The "Find Parameters" window searches for parameters using Ident numbers or parts of the name. All parameters currently available on the axis can be selected.

Saving parameters in drives

This function is used to explicitly save parameters in Sercos to the drive. 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, axes or drives
- Offline parameterization mode; it refers to the parameters of the emulated control, axes or drives

Parameters - Export

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

Open the export function via the context menu of the respective device node in the project tree.

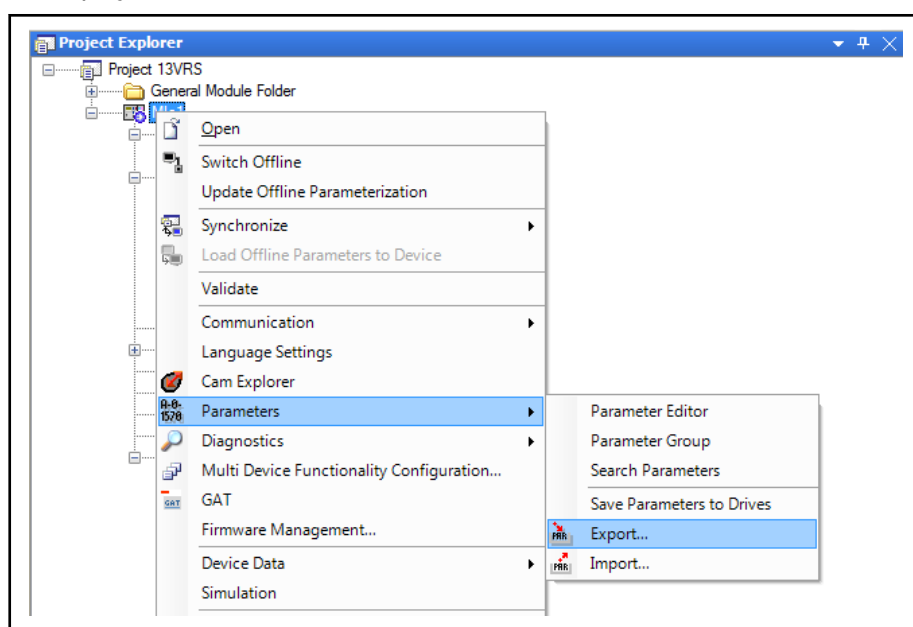


Fig. 4-36: Context menu item "Export Parameters" of the control

Availability

This function is enabled in the following modes:

- Online mode, the parameter export refers to the real parameters of the control or axes/drives

Control

- Offline mode, the parameter export refers to the parameters of the control or axes/drives in the project

Structure Depending on the node called for the parameter export in the dialog, the devices/axes in the online project are displayed.

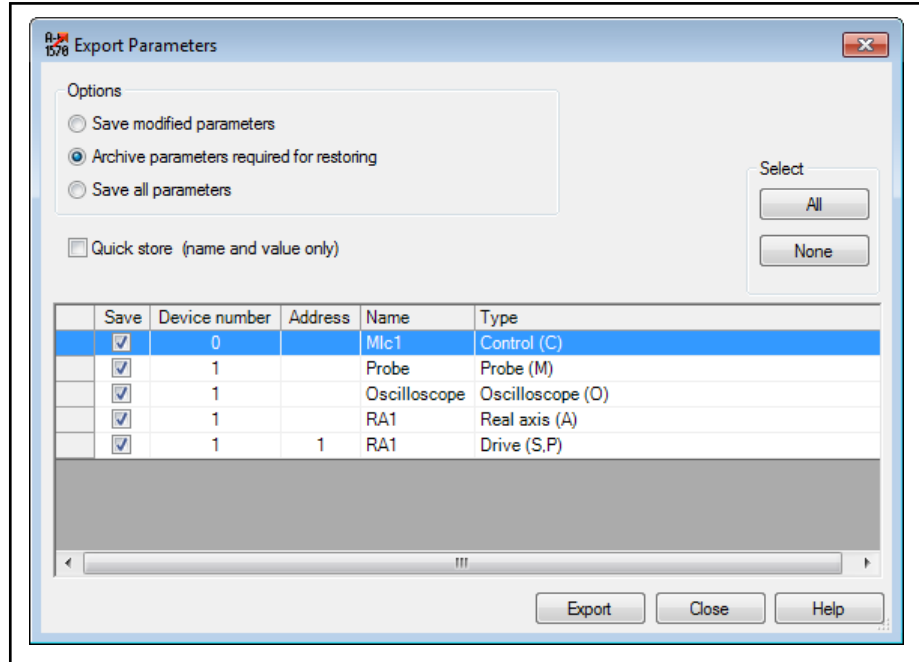


Fig. 4-37: Parameter export, selecting parameters

Column "Save":

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.

Select or deselect via **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 required 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"

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.

Open the import function via the context menu of the respective device node in the project tree.

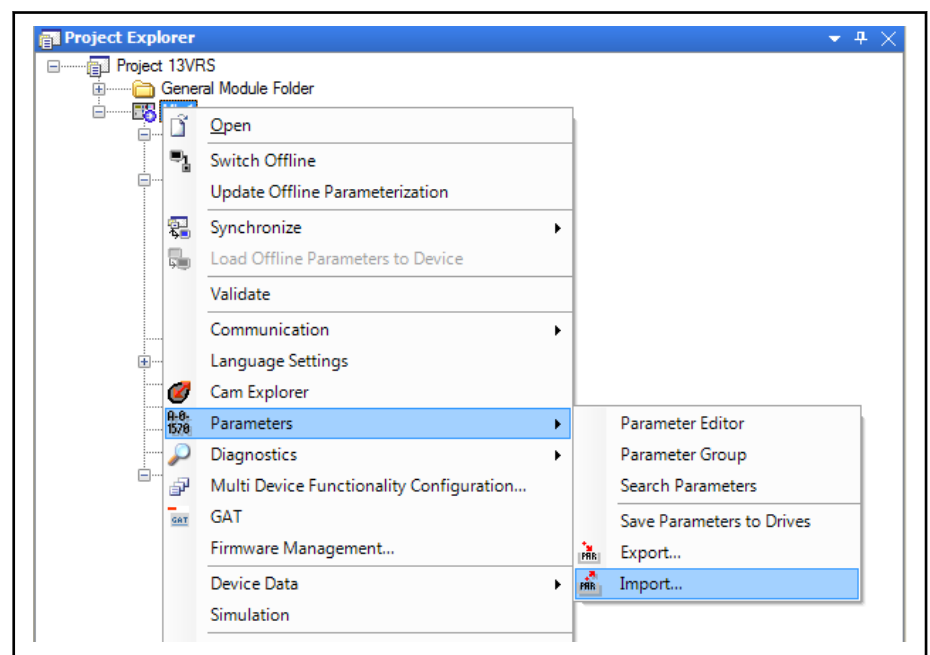


Fig. 4-38: Context menu items "Import Parameters" of the control

Availability This function is enabled in the following modes:

- Online mode, the parameter import refers to the real parameters of the control or axes/drives
- Offline mode, the parameter import refers to the parameters of the control or axes/drives in the project

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

Depending on the node opened for the parameter import, the devices/axes in the parameter file are displayed.

Control

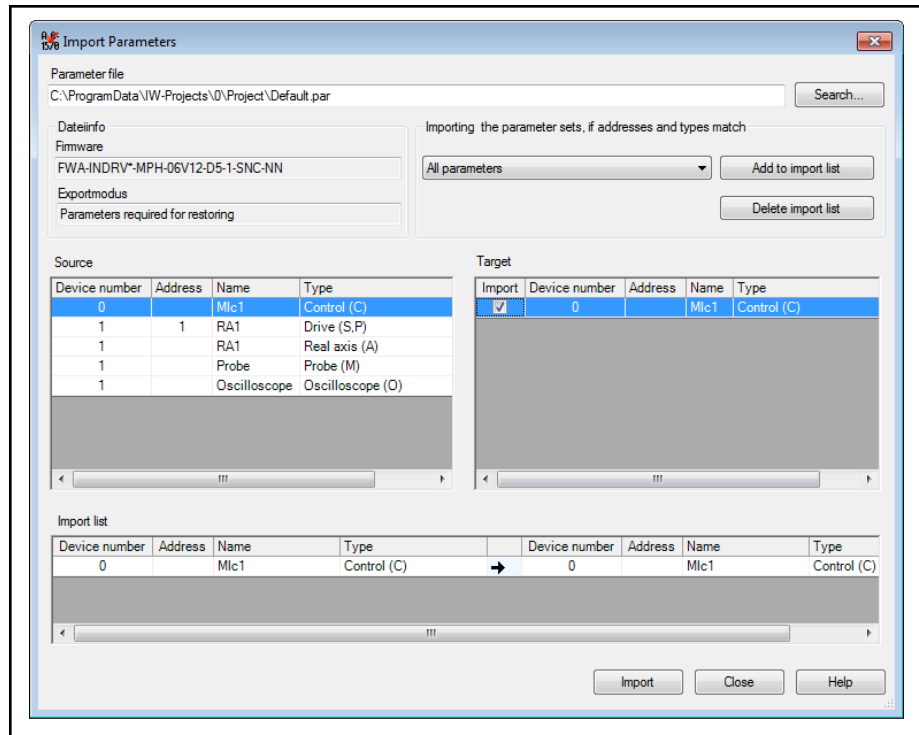


Fig. 4-39: Import settings

Parameter file:

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

Press **Search** to load another parameter file.

File info:

- Firmware**

The firmware release, which 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 loaded.

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

Control and axis names are not overwritten by the parameter import to also parameterize identical system components occurring multiple times in one system. Control and axis names are refreshed each time from the project when going online.

Modify the parameter import using the following function:

- "Importing parameters if address and type match"
Use a respective selection and click on **Add to Import List** to automatically create an import list including address and type.
- Delete the complete import list via **Delete Import List** button
- Start the parameter import via **Import**

4.4.7 Diagnostics

Device status

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

Device	IP address	Control name	Control mode
0 Mlc1	10.110.244.1	Mlc1	STOP

System diagnostics: A0200003, P2 STOP, Motion Phase 2 erreicht, PLC in Stop

User name: karl.walt

Firmware: CML65s-MLC-13T15.0632

System time of control: 2/6/2022 2:19:58 PM

Axis states for all axes

Axis N.	Ad...	Type	Name	Diagnostics	Position	Motion	Status bit
1	1	Real axis	RA1	Axis is in 'StandStill'	0.0000	Standstill	S----

Fig. 4-40: Example: Device Status dialog of an IndraMotion MLC

Delete all present errors of the control with **Clear Error**. 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 [chapter "Error/diagnostic memory" on page 120](#) opens

Press **More Details** to open the window "Advanced Properties" (see [chapter "Advanced properties" on page 119](#))

Control

Set System Time applies 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 feedbacks of the axis are indicated in a compressed manner. 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 into the axis list and a context menu with additional functions opens:

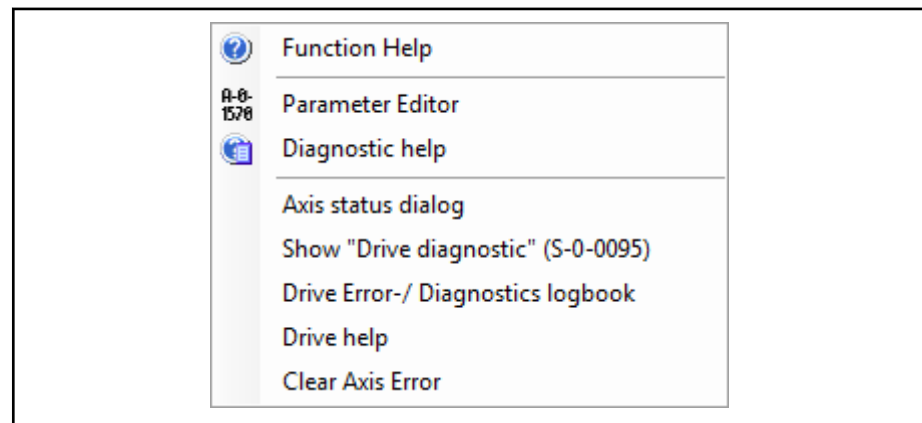


Fig. 4-41: Context menu with additional functions



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:

Advanced Properties - Mlc1

Mlc1

Hardware

Hardware version: CML65.1-3P-500-NNNN-NW

Serial number: SN: 005168645

Hardware details:

Name	Value
MaterialIndexNr	102
MaterialNr	R911170900

Versions

Firmware: CML65s-MLC-13T05.0261

PLC firmware: 13T05.06

BSP: IC_CML65_BSP_V13.1.21.0

Current values

Controller address: 0

Temperature

Current: 45 °C

Maximum: 45 °F

Operating hours counter: 8484.15 h

Operating hours counter, fan: 247.76 h

☐ Fan on

☐ Temperature warning

CPU load

Warning from: 100 %

Minimum: 16 %

Maximum: 35 %

30%

Copy Data to the Clipboard

Reset Min/Max

Fig. 4-42: Example: "Advanced Properties" dialog of an IndraMotion MLC

In the "Current Values" container, 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-0-0031, Control address
- 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
- C-0-0417, CPU load, minimum
- C-0-0418, CPU load, threshold value for warning

Control

- C-0-1051, Command: Resetting the parameters 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.



If this command is executed, the pending warnings of the control are deleted. Thus, these warnings are reported again.

Clear Error can be opened from different locations:

- In the context menu of the control via **Clear Diagnostics/Error**
- In the dialog [chapter "Device status" on page 117](#)
- In the dialog [chapter "Error/diagnostic memory" on page 120](#) and
- In the [chapter 4.2 "Toolbar of the Control XLC/MLC Info" on page 82](#)

Error/diagnostic memory

Overview

The "Error/Diagnostic Memory" dialog provides an overview on all diagnostic messages of the control and the connected devices. The diagnostic messages are sorted automatically. The latest messages are on top of the list.

Category	Time	Source	Status Code	Text
Message	2/6/2022 8:50:31 AM	IndraControl	A00C0002	EnClear, Fehler gelöscht
Message	2/6/2022 7:19:18 AM	Kinematics ...	A0290002	Kinematik wurde angelegt
Message	2/6/2022 7:19:17 AM	Axis 2 (VA2)	A0110002	Achse wurde angelegt
Message	2/6/2022 7:15:25 AM	Axis 1 (VA1)	A0110002	Achse wurde angelegt
Message	2/6/2022 2:49:22 AM	IndraControl	A0200023	P2 RDY, Motion Phase 2 erreicht, PL...
Message	2/6/2022 2:49:20 AM	IndraControl	A0200021	P0 RDY, Motion Phase 0 erreicht, PL...
Message	2/6/2022 2:46:39 AM	IndraControl	A0200025	BB RDY, Motion betriebsbereit, PLC...
Message	2/6/2022 2:46:39 AM	IndraControl	A0200024	P3 RDY, Motion Phase 3 erreicht, PL...
Message	2/6/2022 2:46:36 AM	IndraControl	A0200023	P2 RDY, Motion Phase 2 erreicht, PL...
Message	2/6/2022 2:46:33 AM	IndraControl	A0200021	P0 RDY, Motion Phase 0 erreicht, PL...
Message	2/6/2022 2:46:33 AM	IndraControl	A00C0001	BOOT END, Booten der Steuerung b...

Total: 11 entries

Fig. 4-43: Example of an MLC diagnostic logbook representation



A maximum of the latest 1,000 messages is saved in the control. Older messages are automatically deleted.

The dialog can be opened online and offline. In offline mode, some functions are disabled (the value "Display duration of messages in the display" can neither been seen nor changed for example). For older controls apply that the message list is only displayed online.

Messages

The biggest dialog section presents the message list. Each message obtains the following information:

Category: The category indicates the severity of the message (whether it is an error or a warning). Each diagnostic message belongs to one of the following categories:

Category	Description
 Message	This is a general information (on a new operating state of the control for example)
 Error	This error has not yet been acknowledged with the "Clear Error" function
 Error	This error was acknowledged with the "Clear Error" function The error date indicates when the error occurred
 Error	This specifies that the error was acknowledged with the "Clear Error" function The date indicates when the error was acknowledged
 Warning	This is an active warning (the warning cause is still present)
 Warning	This is an inactive warning (the warning cause is not present anymore). The warning date indicates when this warning occurred
 Warning	This specifies when the warning was not present anymore. In contrast to errors, warnings cannot be acknowledged manually. They are carried out automatically

Tab. 4-3: Message categories

Time: The time when message was received.

Source: The device or the logical unit that created the message. The following sources are defined:

- **IndraControl:** The control itself
- **Function block:** A function block processed in the PLC program. If this function block operates on an axis, number and name are specified
- **Axis:** One axis. Its number and name are additionally specified
- **Device:** A Sercos device. Its Sercos address and name are additionally specified
- **Kinematics:** One kinematics. Its number and name are additionally specified

 The source name is determined using its logical number or the Sercos address and the current project data. If the configured elements have changed since the error occurred, the current and not the old name are displayed.

Status code: This is a unique code used to identify the message, for example to look for more information on the cause or troubleshooting in the online help.

Text: This is the message text. The language of the message text does not depend on IndraWorks, but on the language set in the control.

Filters

There is the filter bar right above the error list. Enter a criterion to specify how to filter the list. A criterion can be entered in every column. The individual filter criteria of the columns are combined in an AND operation and used to filter the message list. The individual filter criteria:

Control

- **Category:** Filters for errors, warnings or messages in this list. Filters in individual categories with an OR operation
- **Time:** Enter the starting and end time of the messages to be displayed. If at least one "time" was entered, the corresponding button is displayed as pressed and the selected time range is displayed as tooltip
- **Source:** Enter the text used for filtering
- **Status code:** Enter the text used for filtering
- **Text:** Enter the text used for filtering

If a text is entered into a text field of the filter bar, it is immediately filtered in the respective column. It is sufficient if the entered text is anywhere in the column. Enter the value "20" into the "Status code" column for example and the codes "A0200010" and "F2004002" are displayed. A preceding exclamation mark (negation) ensures that only messages **not** containing the searched term are displayed. The value "!A02000" thus causes most of the messages to be hidden at phase switching.

The wildcards "*" and "?" can be used additionally. The "?" is a placeholder for any character. The "*" is a placeholder for any number of characters (usually a word).

Example:

Using wildcards

- The filter "PLC in *" finds the texts "PLC in Init", "PLC in Ready" and "PLC in Run"
 - The filter "A020002?" finds all status codes from "A0200020" to "A0200029"
-

"View" Area

The area includes control elements used to influence the representation in the diagnostic list.

- **Filters On/Off:** Use this button to enable and disable the filter. Thus, fast switching between the filtered and unfiltered view is possible
- **Clear Filter:** Use this button to delete the filter to quickly enter a new filter for example
- **Automatic Scrolling On/Off:** Use this button to set whether moving to the most upper line is to be performed automatically when there is a new message. Enable this button to display the latest message without any further operating actions. To analyze an error in detail without seeing the new messages, disable the button

"Control" Area

The area includes control elements used to influence the control response.

- **Clear Error:** This button functions like the error deletion button via the context menu of the control. See also [chapter "Clearing errors" on page 120](#)
- **Display duration of messages in display:** Set additionally 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. The warnings or errors are displayed until they are acknowledged. Note that this field can only be changed if the control is on-line. Otherwise, the field is disabled

Context menu of the diagnostic list

Right-click on the message list and a context menu with the following entries is displayed:

- **Function Help:** This menu item opens the help on the dialog "Error/Diagnostic Memory"
- **Help on the message:** This menu item calls the help for the currently selected message. Alternatively, double-click on a message in the list
- **Copy Selected Line:** This menu item copies the content of the currently selected message to the Windows clipboard. With the set format for clipboard, the message can be directly inserted into a spreadsheet (e.g. MS Excel).
- **Copy Complete Table:** This menu item copies the content of all messages to the Windows clipboard. With the set format for clipboard, the messages can be directly inserted into a spreadsheet (e.g. MS Excel).

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 using an example:

The "positive velocity limit value" is to be increased. Thus, the user exceeds the maximum admissible input value.

The control is in the operating mode **BB**. The input is immediately applied to the drive that responds with an error message.

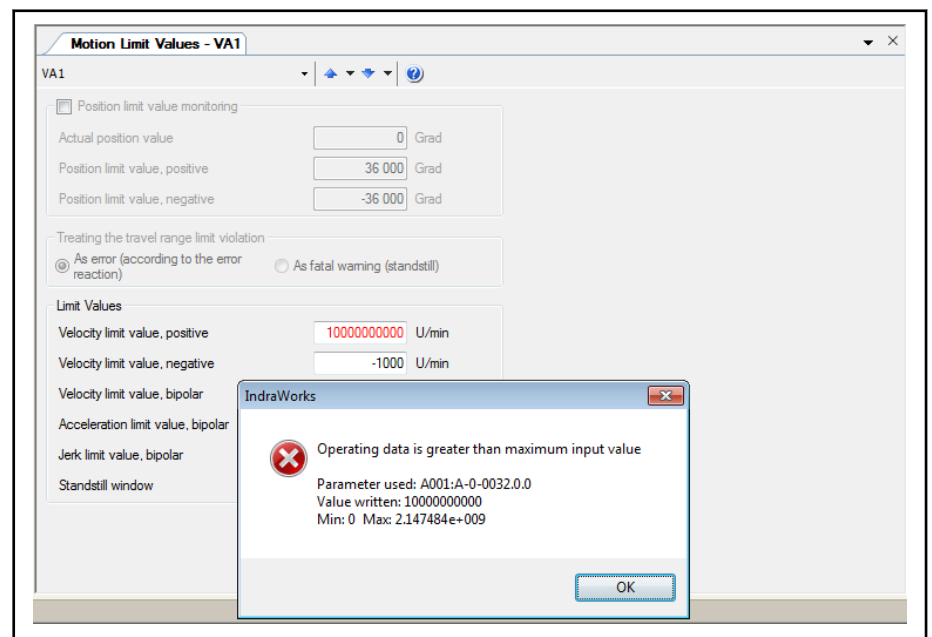
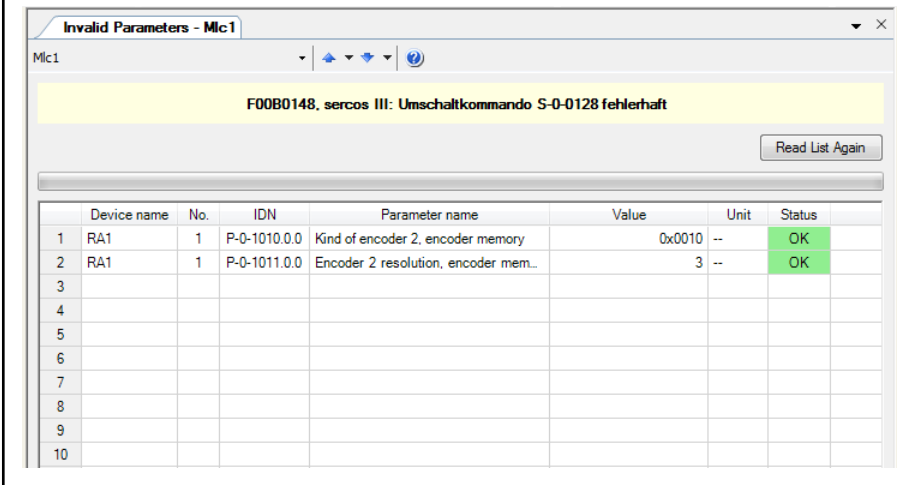


Fig. 4-44: Incorrect entry, value not within the valid value range

Control

If the change is made in the parameterization mode **P2**, it is only applied when 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 opens automatically.



	Device name	No.	IDN	Parameter name	Value	Unit	Status
1	RA1	1	P-0-1010.0.0	Kind of encoder 2, encoder memory	0x0010	--	OK
2	RA1	1	P-0-1011.0.0	Encoder 2 resolution, encoder mem...	3	--	OK
3							
4							
5							
6							
7							
8							
9							
10							

Fig. 4-45: List of invalid parameters

Now, make the appropriate correction(s) and repeat the attempt to switch from **P2** to **BB**.



Frequently, lines, which seem to be identical at first glance, are listed. The reason is that the S- and P-parameters determined as invalid are often overwritten with A-parameters. In this case, it is necessary to write to the corresponding A-parameter. Writing to 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 5.4.5 "Task list/configuration" on page 186](#).

Task viewer

The "Task Viewer" displays the activity of tasks in a task-time diagram

This dialog is described in [chapter 5.4.6 "Task viewer" on page 188](#).

Control

⚠ WARNING

Deleting MLPI connection has unknown consequences for the client establishing this connection. Depending on the programming, the reaction of the client can be between new connection establishment and complete crash.

Delete only the connections that are not used anymore.

4.4.8 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.

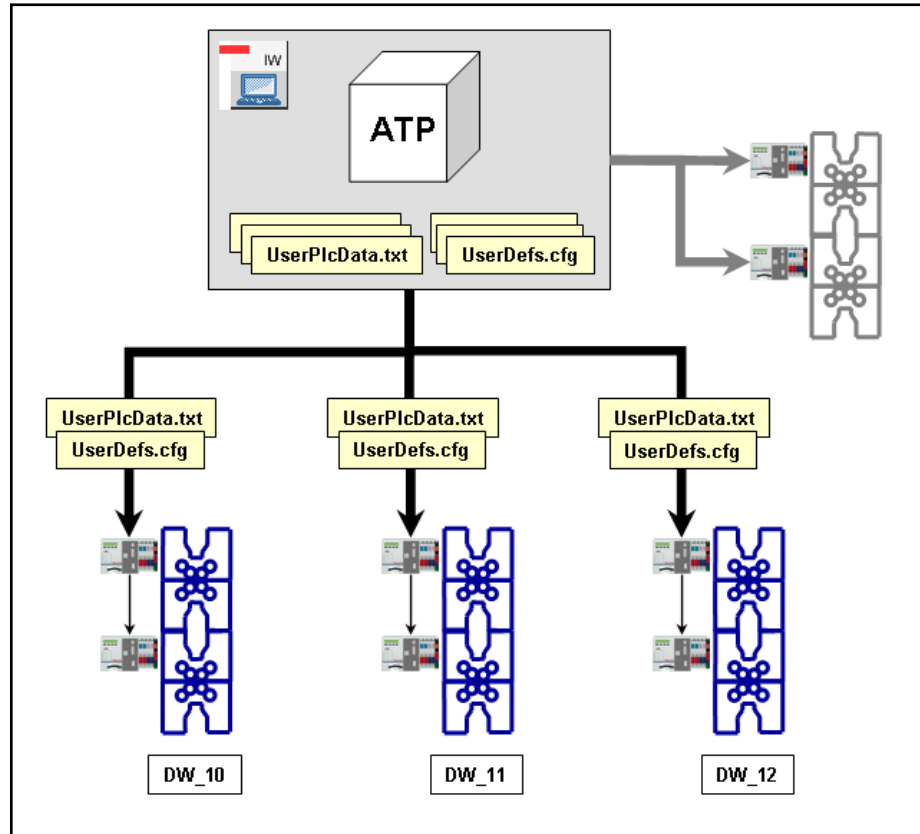


Fig. 4-47: An IndraWorks project ("ATP") downloaded together with device-specific files to several controls.

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 in the multi-device table and can be individually selected and enabled to go online (Motion) or to log in (PLC).

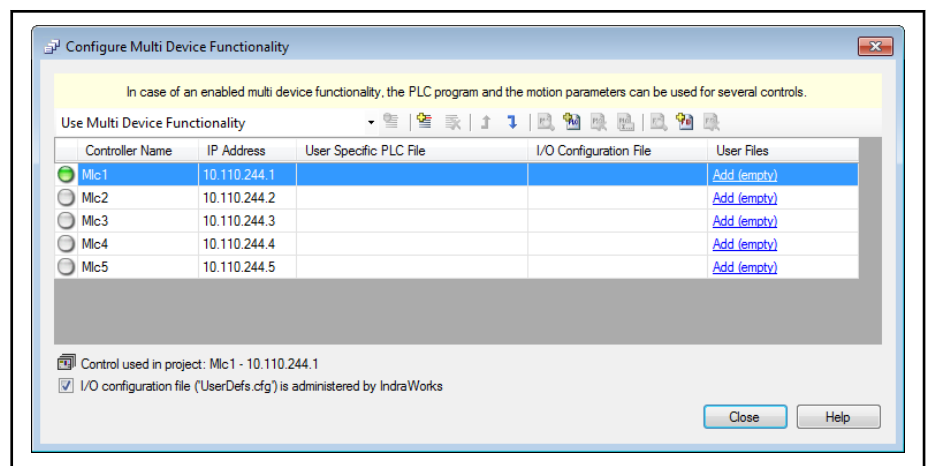


Fig. 4-48: The "Multi-Device Configuration" dialog with the activated multi-device functionality and the multi-device table

Brief description of the multi-device table

Opening the multi-device table

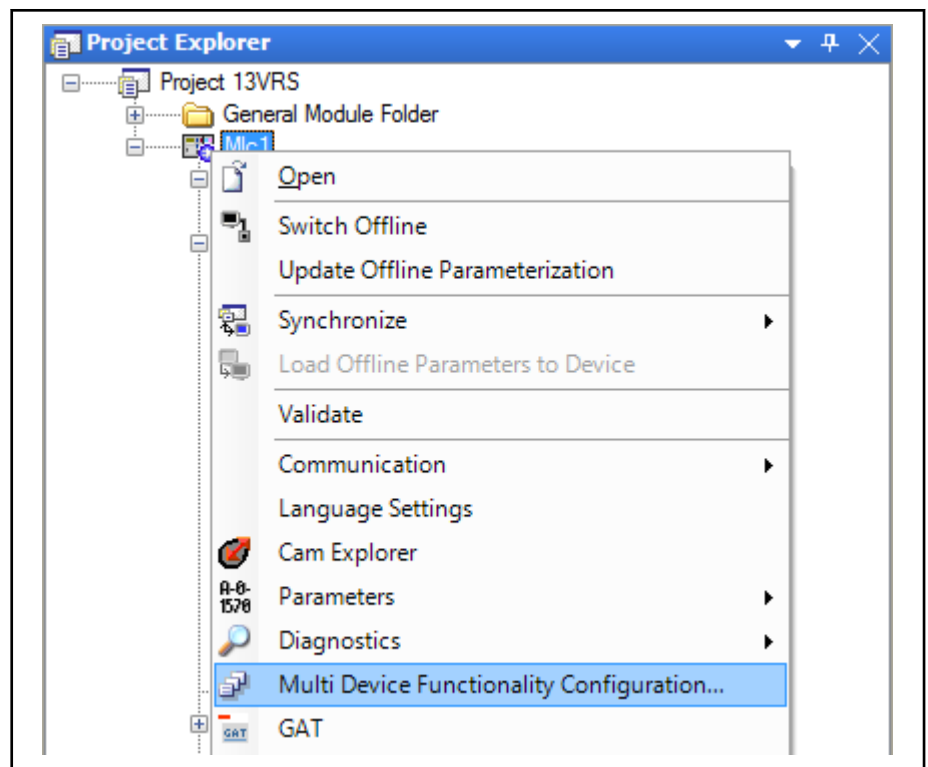


Fig. 4-49: Opening the dialog with the multi-device table

In IndraWorks, exactly one multi-device table can be assigned to each control node (IndraLogic XLC) 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

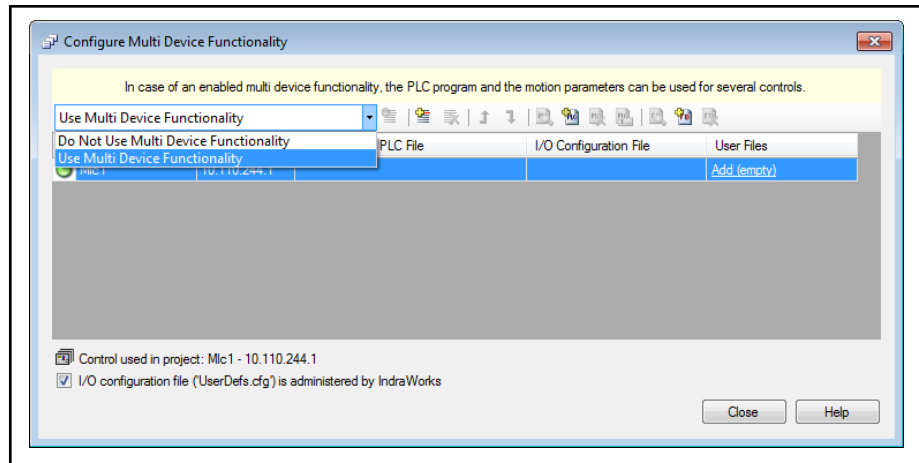


Fig. 4-50: Activating the multi-device functionality

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

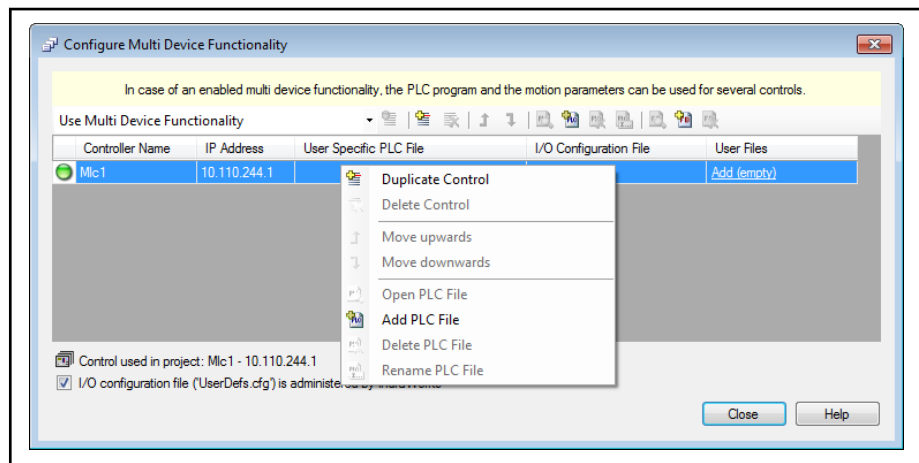


Fig. 4-51: Adding another control based on the selected control (duplicating)

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 (indicated 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. 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 disabled ↔ enabled

Multi-device disabled

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 disabled when logging into the PLC, no "user-specific PLC data" and no "user data" are applied.

Multi-device enabled

If the multi-device functionality is enabled, 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.

At each login to the PLC, the following data is transferred to the Compact Flash card of the control (user partition, directory "/ata0b"):

- User-specific PLC data file (if available)
- Configuration file to disable the Profibus DP slaves (if available)
- User files (if available)

The parameter C-0-0030, "Control name" is additionally read-only during a parameter import.

Basic behavior of the multi-device table

Editing the table

To edit the cells of the multi-device table, click or double-click.

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 in a line, the data of this line is internally applied to IndraWorks.

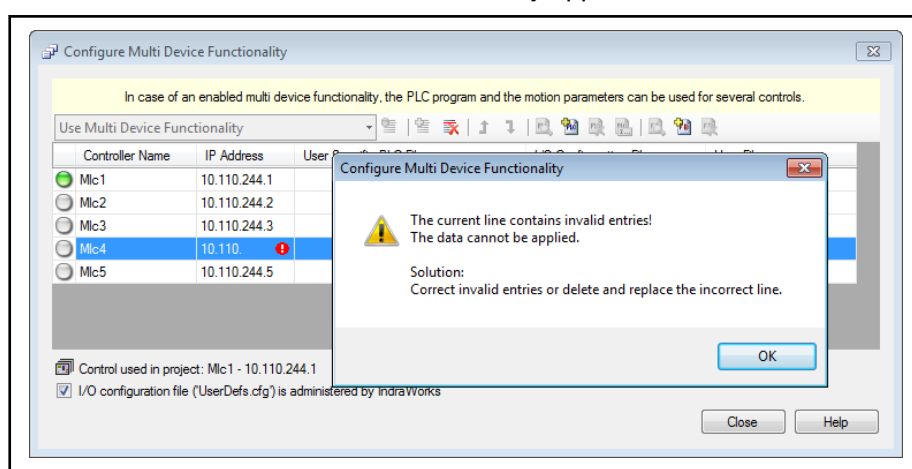


Fig. 4-52: Note on invalid entry

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 opened 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 has to be unique in a multi-device table.

Control

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, different user-defined data files (see [chapter "Column User-specific PLC data" on page 130](#)) can be used for one physical device for example.

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 into the PLC.

If specific data - such as initialization values - is to be used for the individual controls in 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 in the multi-device table. The task of the PLC program is to analyze this user-specific PLC data file at runtime. The IndraLogic XLC 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 these devices. 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"

Configured, 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 figures of the IndraWorks interface.

Project planning and configuration

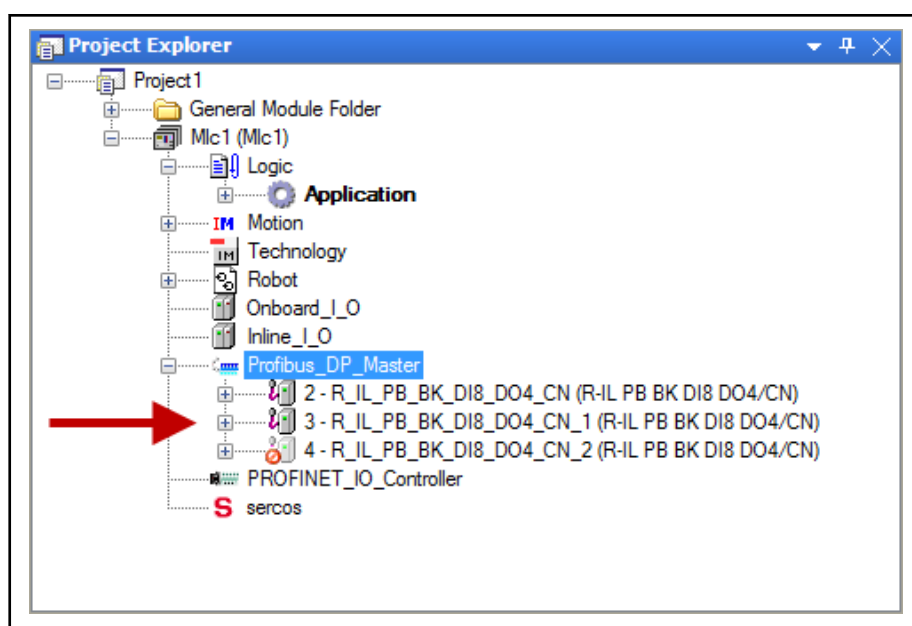


Fig. 4-53: Configured Profibus DP slaves

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

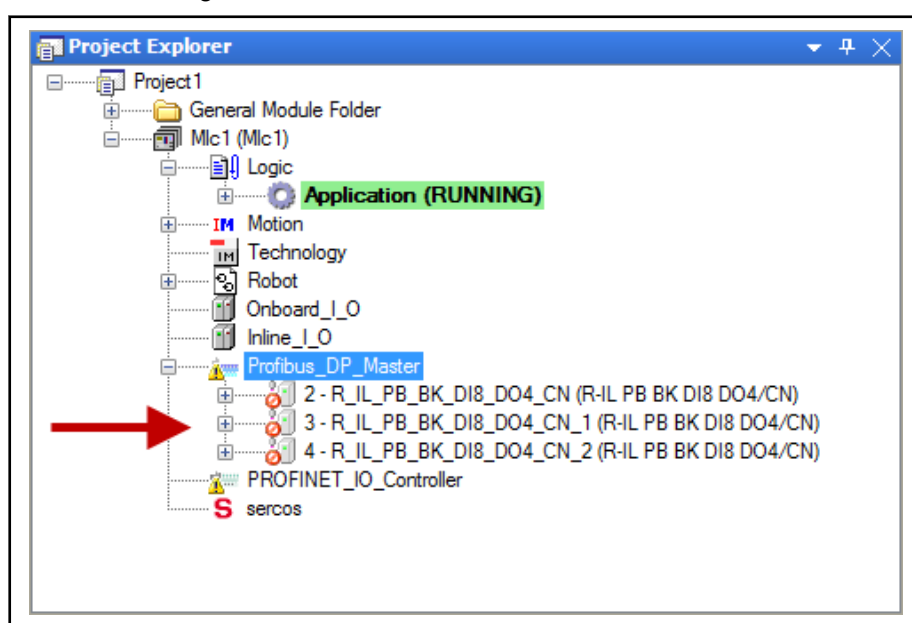


Fig. 4-54: Configured Profibus DP slaves disabled

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 in 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,

Control

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 renamed.
- 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 into 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 files can be assigned to each individual control in the multi-device table:

- exactly one user-specific PLC data file
- exactly one configuration file "UserDefs.cfg"
- several user files (described here)

The PLC program has to use these "user files" properly. The IndraLogic XLC 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 in the directory are overwritten. 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 tables.

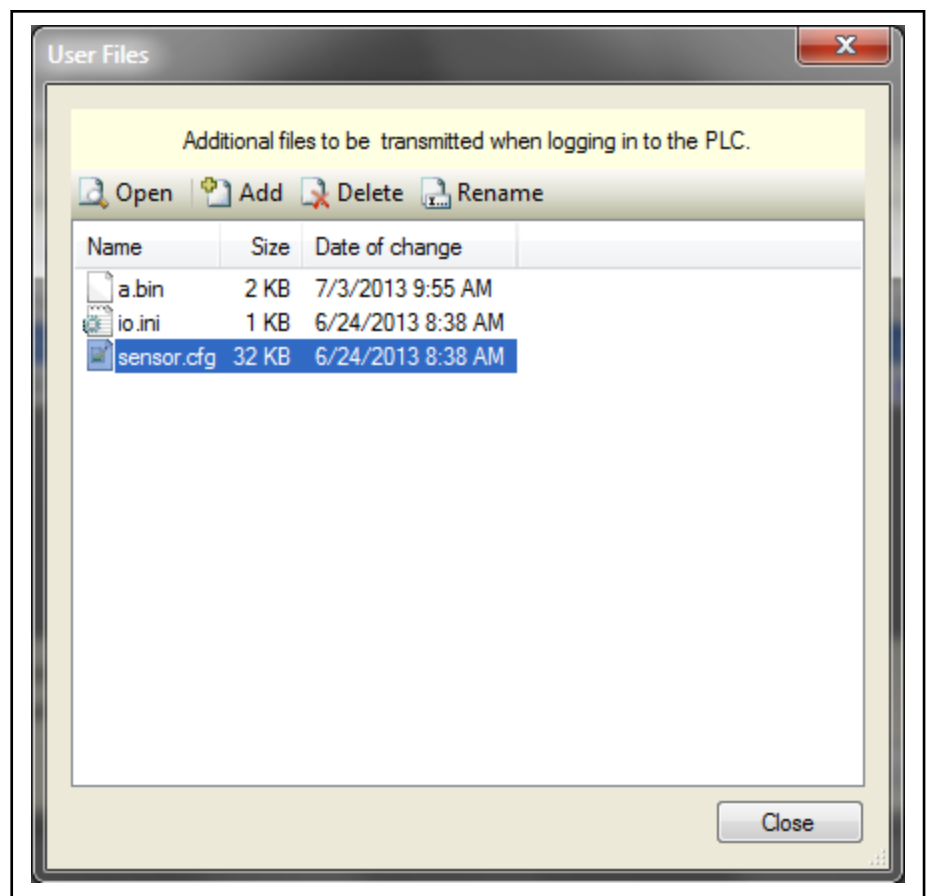


Fig. 4-55: "User Files" dialog

I/O configuration file to disable field bus slaves

Disabling slaves of different field buses

The configuration file "UserDefs.cfg" to disable field bus slaves can be used to deactivate field bus slaves configured in IndraWorks. The deactivation is performed via a file on the Compact Flash card of the control.

Thus, diagnostic messages are impeded for slaves configured in IndraWorks, but that do not really exist.

For detailed information on

- Sercos III slave deactivation
- Profibus DP, slave deactivation
- INLINE, module deactivation

refer to the documentation "Rexroth IndraWorks 13VRS Field Buses" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Control

Managing the configuration file "UserDefs.cfg"

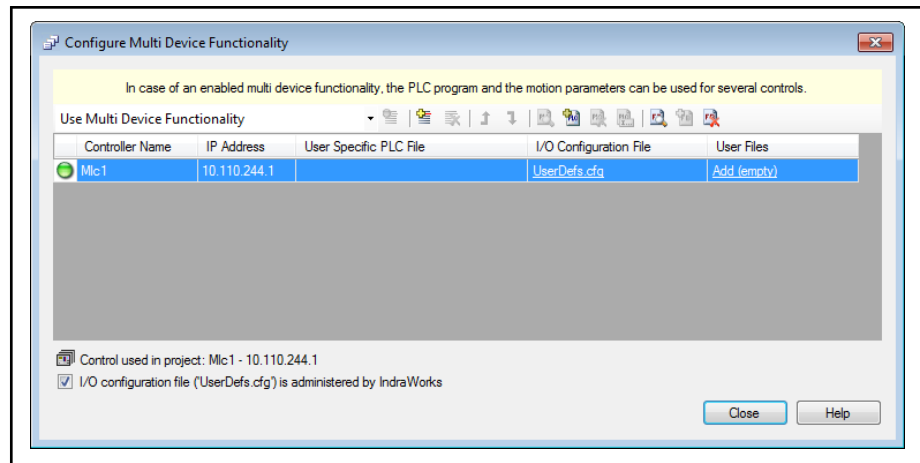


Fig. 4-56: Either IndraWorks or the user can manage the I/O configuration file "UserDefs.cfg" in the compatibility mode 13V04 and higher

Compatibility mode up to 13V02

To disable field bus devices, use the I/O configuration file "UserDefs.cfg". If the current IndraWorks project is used in the compatibility mode 13V02, IndraWorks manages the file "UserDefs.cfg". Thus, if multi-device is used and if the file "UserDefs.cfg" is in the multi-device table in the line of the control used, this file is written to the PLC upon each login and upon each PLC program download to the user partition of the CF card of the IndraLogic XLC. If the file "UserDefs.cfg" is not available in the multi-device table in the line of the control used, the file - if present - "UserDefs.cfg" is deleted on the user partition of the CF card of the IndraLogic XLC upon each login to the PLC as well as upon each PLC program download. If multi-device is not used, the file - if present - "UserDefs.cfg" is deleted on the user partition of the CF card of the IndraLogic XLC upon each login to the PLC as well as upon each PLC program download.

This behavior cannot be changed in the compatibility mode 13V02.

Compatibility mode from 13V04

The behavior was changed for the compatibility mode 13V04 and higher. If multi-device is not used in this mode, the file "UserDefs.cfg" - if present - on the user partition of the CF card of the IndraLogic XLC is not deleted (new). If multi-device is used, the user decides whether to manage the file himself (new) or whether IndraWorks keeps managing the file (as up to compatibility mode 13V02). If the user manages the file, the file "UserDefs.cfg" is neither written to the CF card of the IndraLogic XLC nor a present file with the same name is deleted on the CF card when logging into the PLC or during PLC program download.

4.4.9 Firmware management

Target	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 13VRS) and 13VRS (higher than 13V02) are installed • IndraWorks has started and a project containing the control was loaded • 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 13VRS 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 13VRS have to be considered.

Example: If the firmware XLC is to be used for 13VRS, 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 113](#)). The control can contain old program components, parameterizations and PLC data from previous use. These are removed by clearing the memory (see [chapter 3.3.3 "Deleting control memory" on page 37](#)).

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.

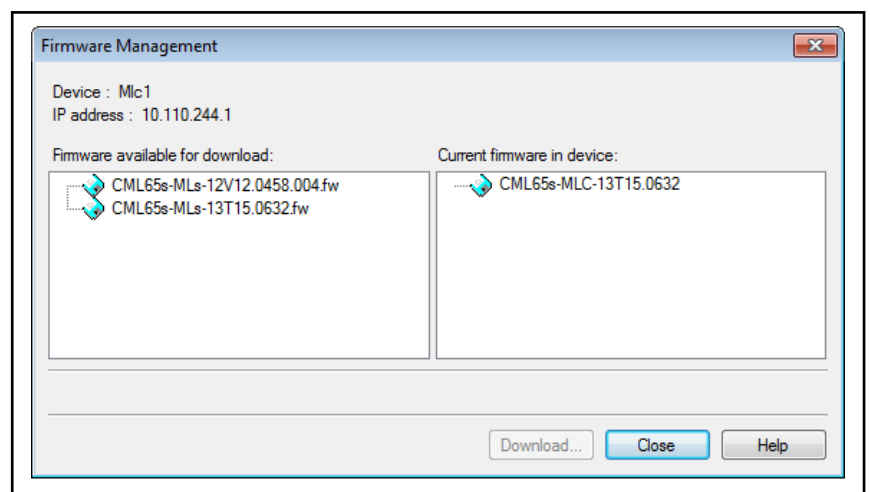


Fig. 4-57: Example: IndraMotion MLC Firmware Management dialog

Enable the window in the context menu of the respective control.

The available firmware versions are located in the left section. Select the desired version and start the download via **Download...**

The versions available for selection are located under:

<Drive>:\All Programs\Rexroth\IndraWorks\<Control>\Firmware.

<Control>: "XLC" or "MLC"

Control



The new firmware is only applied 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".

4.4.10 Drive firmware management

Target Exchanging drive firmware without loss of data.

- Prerequisites**
- The drive is either an IndraDrive or a HydraulicDrive
 - The control and the computer - on which IndraWorks is installed - are located in the same subnet.
 - The control is switched online
 - The drive is connected to the control and enabled

Procedure



Read the release notes for the new firmware first.

The following is assumed: It is an IndraDrive drive, the control runs in **BB RUN** and the axes are active.

It is recommended to follow this sequence:

1. Stop all active axes and ensure that no axes start during the firmware download.
2. Open the firmware download wizard of the corresponding drive below the Sercos node via the context menu item "Firmware Management..." of the drive.
3. Select whether a routing entry to the drive is to be generated and whether the drive parameters and the function package of the drive are to be backed up and restored after a firmware download. Click on **Next>>**.

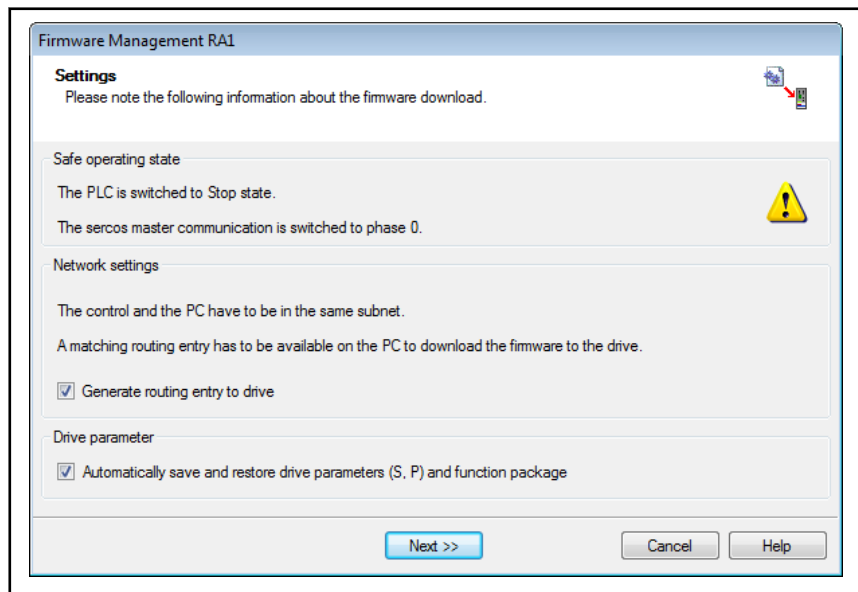


Fig. 4-58: Select the settings for the firmware download

4. Confirm the safety prompts that the control is to be stopped.

Control

5. If selected, the routing entry to the drive is subsequently generated and the drive parameters and the selected function package are backed up.
6. A window is displayed to select the firmware to be loaded to the drive. Select the firmware file via ... and click on **Download...** to download the firmware to the drive. The progress is displayed during download.

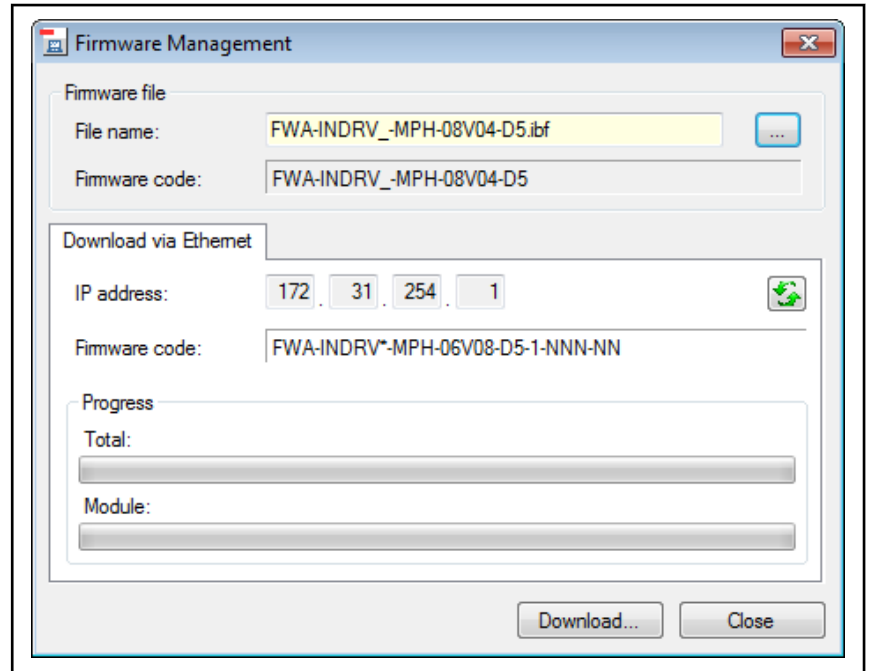


Fig. 4-59: Selecting firmware file for download

7. After the successful firmware download, a message to restart the drive is displayed. Depending on the drive type used, it can be either restarted automatically or manually by switching off and on. The messages displays whether there is an automatic restart or whether a manual restart is required.

Control

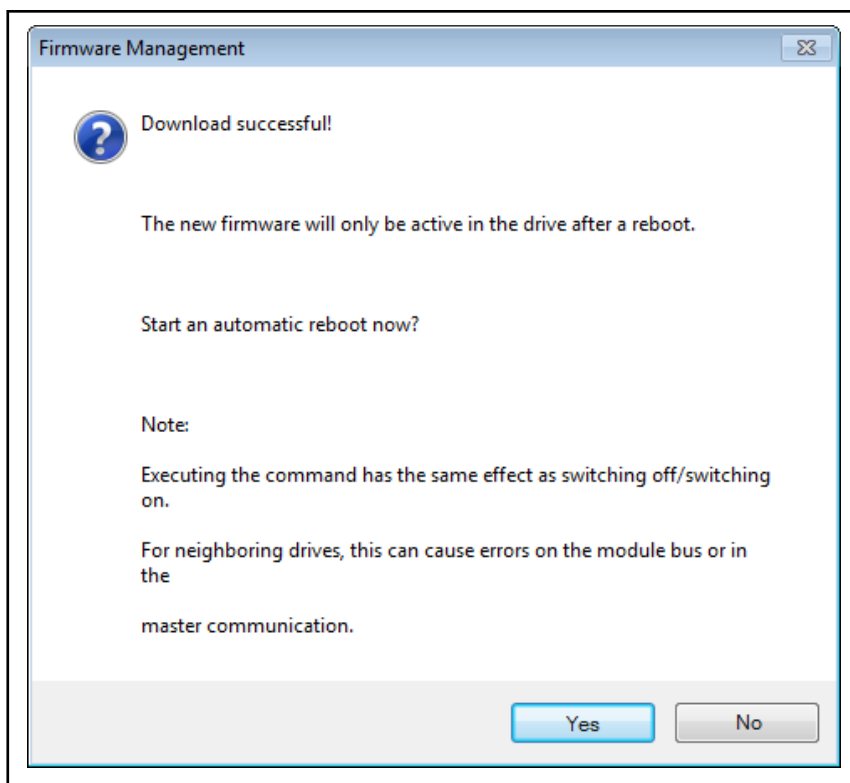


Fig. 4-60: Message displayed if the download was successful and if the drive supports an automatic restart

8. Start the automatic restart with **Yes**.

The progress of the automatic restart is displayed in the following window:

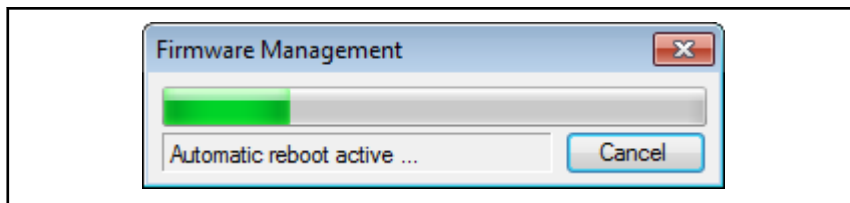


Fig. 4-61: Progress bar "Automatic restart"

9. The drive parameters are restored and the new drive firmware is automatically applied to the IndraWorks project.
10. The result of the firmware download is displayed at the end.

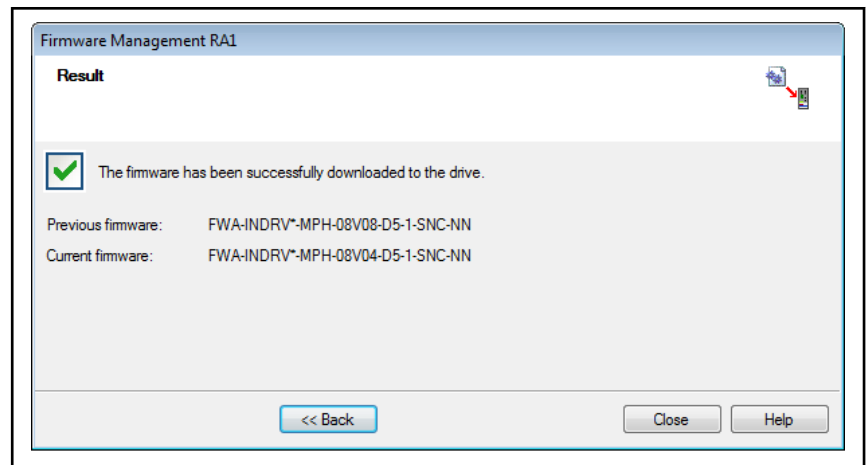


Fig. 4-62: Firmware download result

4.4.11 Device data

Archiving

This menu item can be used to backup the control data (parameters, kinematic data) and the CF partitions of the control in the project 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 archiving.

The following overview shows which data is included in which data type.

Data type	Control data	User partition	OEM partition	System partition
Archives stored on the device			X	
Firmware				X
Control parameters	X			
Drive parameters	X			
PLC retain data				X
Compiled PLC programs			X	
Cam tables applied to the control		X		
Cam tables loaded to the drive	X			
MotionProfiles applied to the control		X		

Tab. 4-4: Overview on data types

Control

User interface

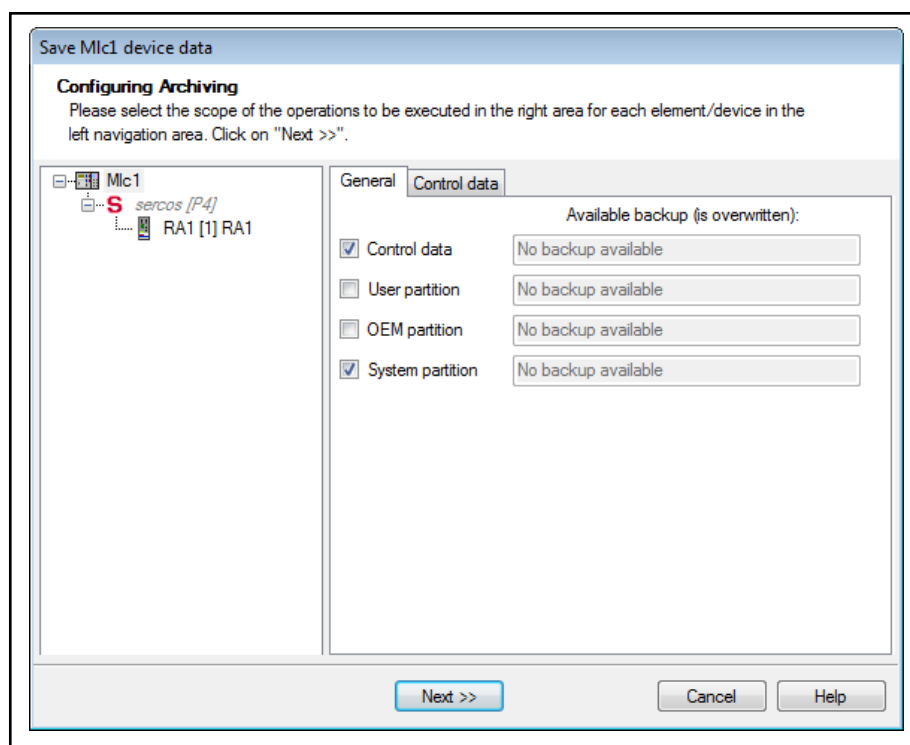


Fig. 4-63: Dialog to configure the control

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 saved in the "Control data" tab.

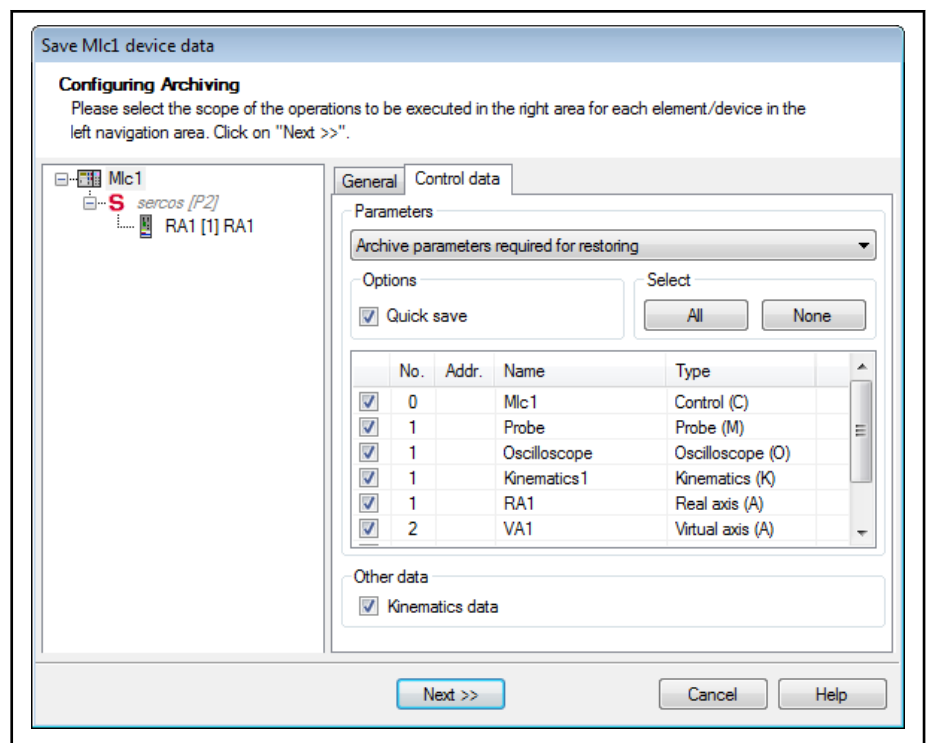


Fig. 4-64: Dialog to select the control data to be archived

Data to be archived is subsequently displayed for confirmation purposes.

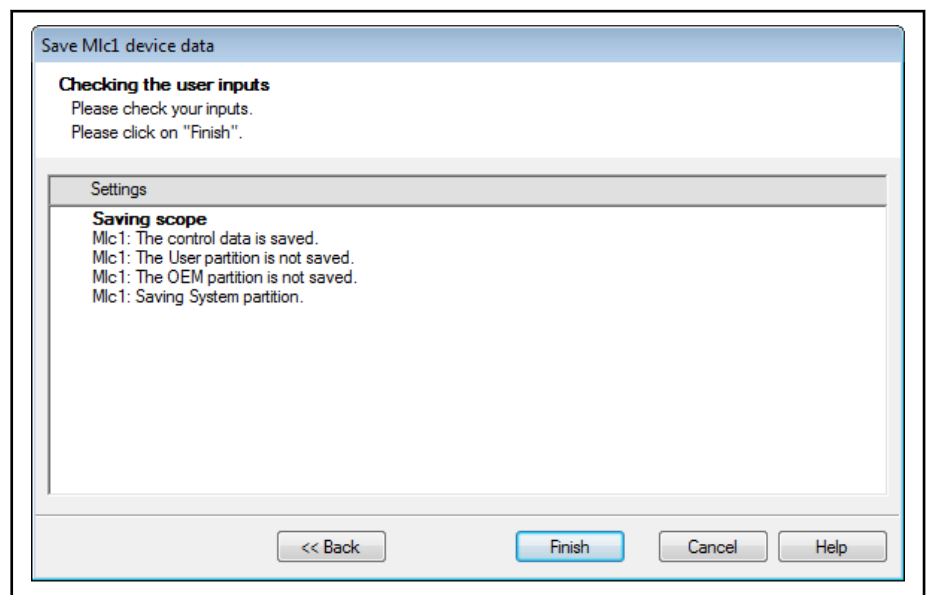


Fig. 4-65: Checking user inputs

Press **Finish** to start the archiving. The total progress is then displayed.

Control

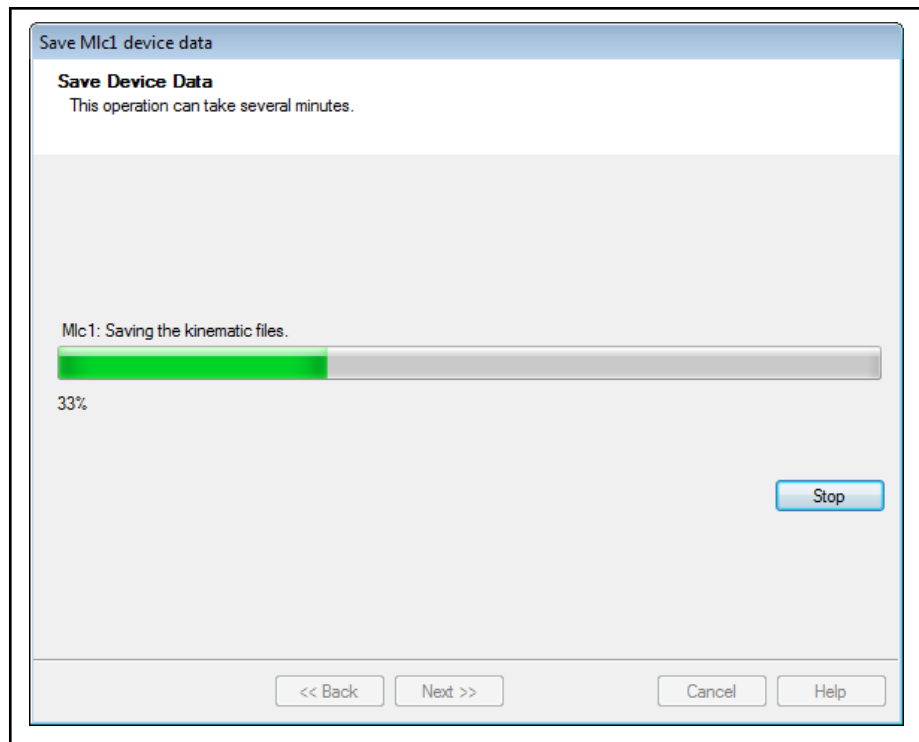


Fig. 4-66: Progress bar of the archiving

At the end, a summary of the archiving results is output.

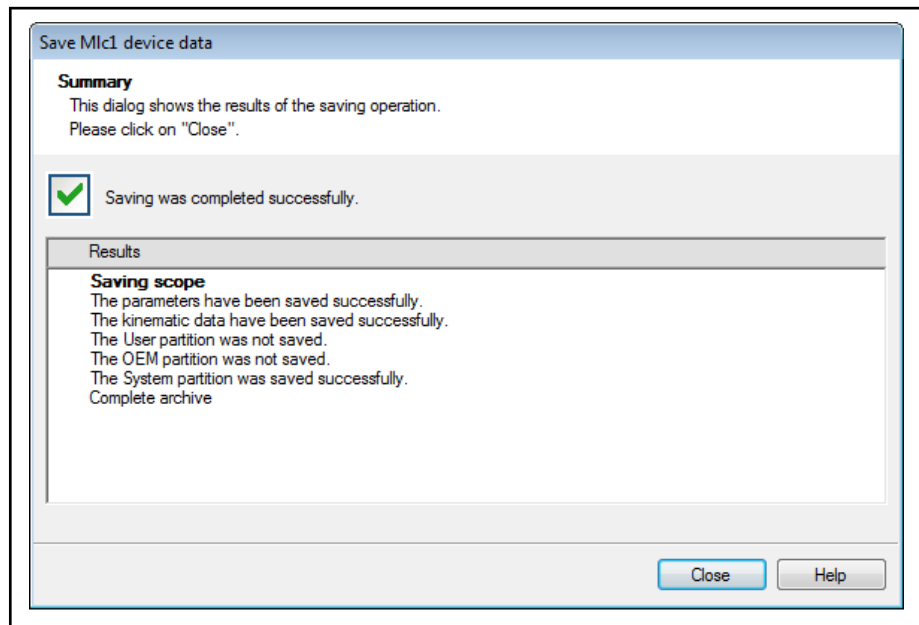


Fig. 4-67: Summary of the archived device data

Complete the archiving dialog with **Close**.

Storage location

The archived data is saved under:

- <Project Directory>\<NameOfControl>\ArchivedData

The individual files are named as follows:

- Parameter: Params.par
- User partition data: CF_User.zip

- OEM partition data: CF_OEM.zip
- System partition data: CF_System.zip

Restoring

User interface This menu item restores control data (parameters) and CF partitions on the control. This function is only available if there is an online connection to the server.

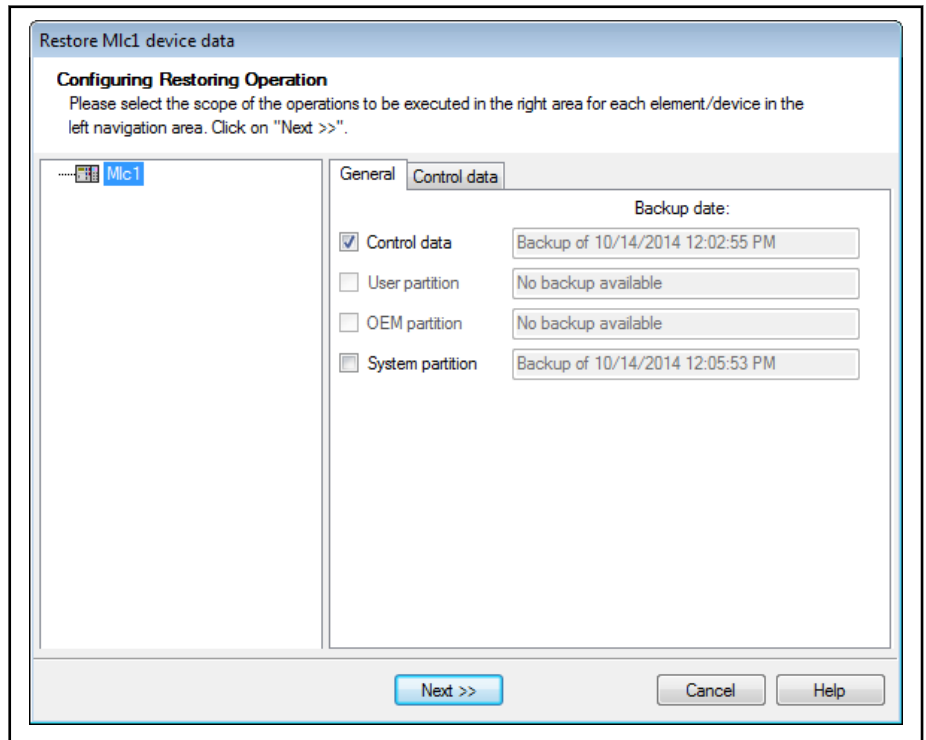


Fig. 4-68: Dialog to restore device data

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, which are not located in the project, cannot be selected.



Restoring the system partition overwrites the current firmware 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.

Control

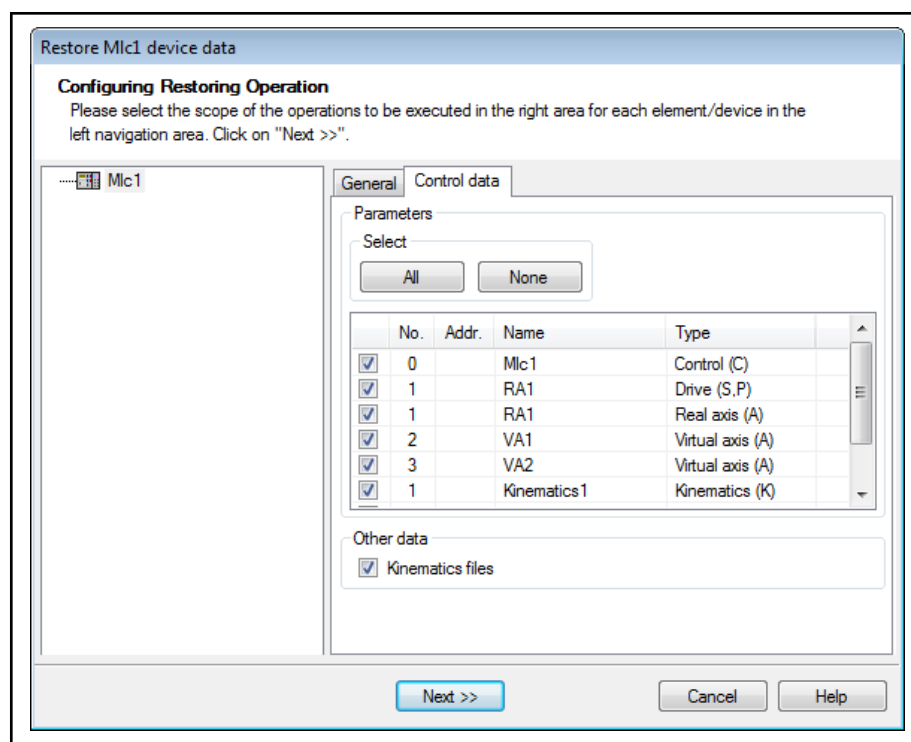


Fig. 4-69: Selecting data to be restored

Finally, the data to be restored is displayed for confirmation purposes.

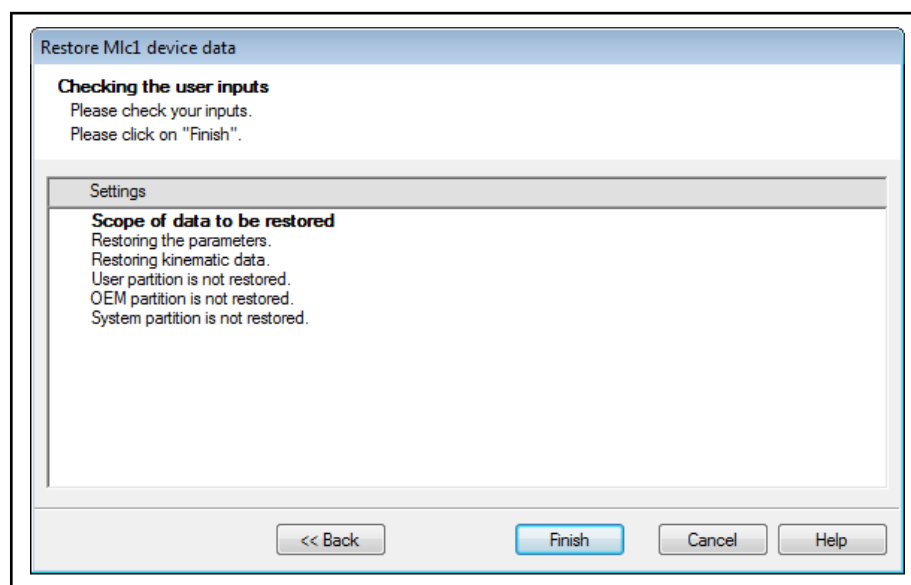


Fig. 4-70: Checking user inputs

Click on **Finish** to start the restoration. The total progress is then displayed.

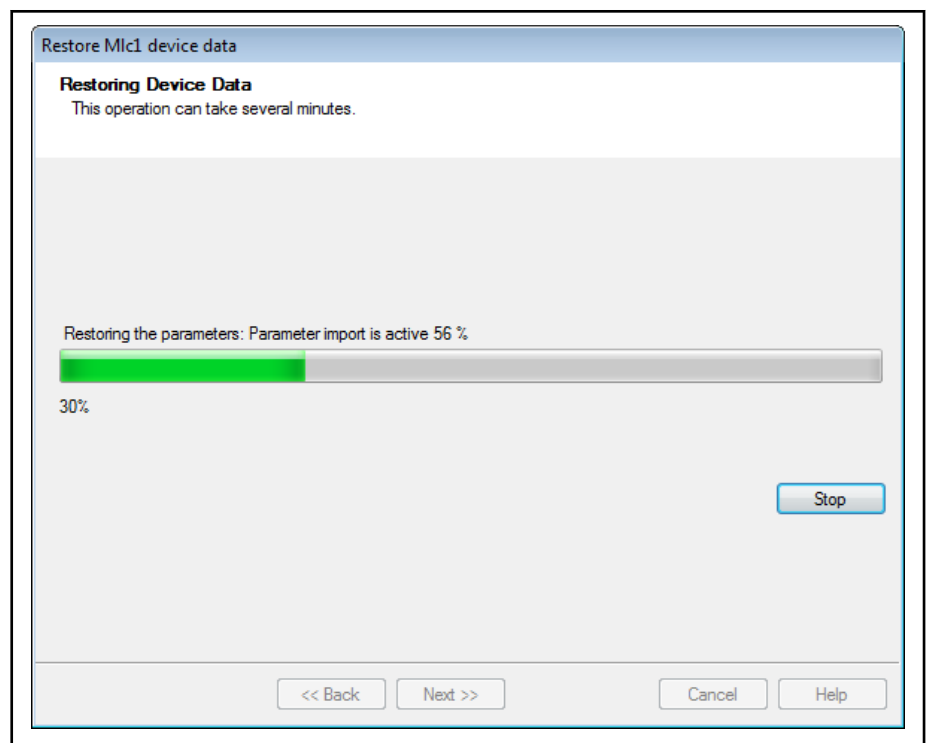


Fig. 4-71: Restoring...

At the end, a summary with the restoration result is output.

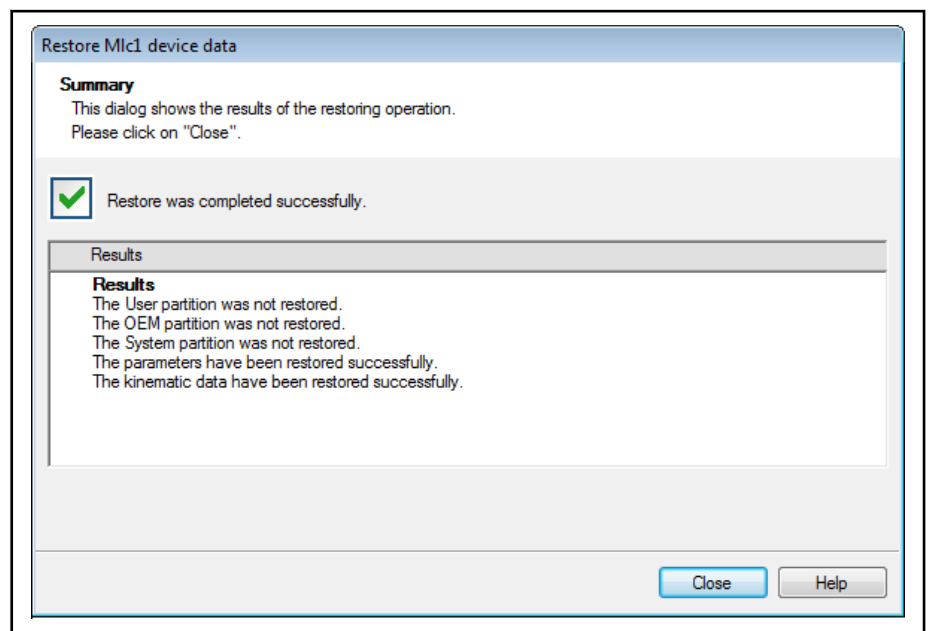


Fig. 4-72: Restoration summary

Explorer

To transfer file and folders between PC and control, use the FTP Explorer. The FTP Explorer allows to access the OEM partition and the USER partition of the control.

- Opening the Explorer
1. Right-click to open the context menu of the control and select **Device Data ► Explorer**

Control

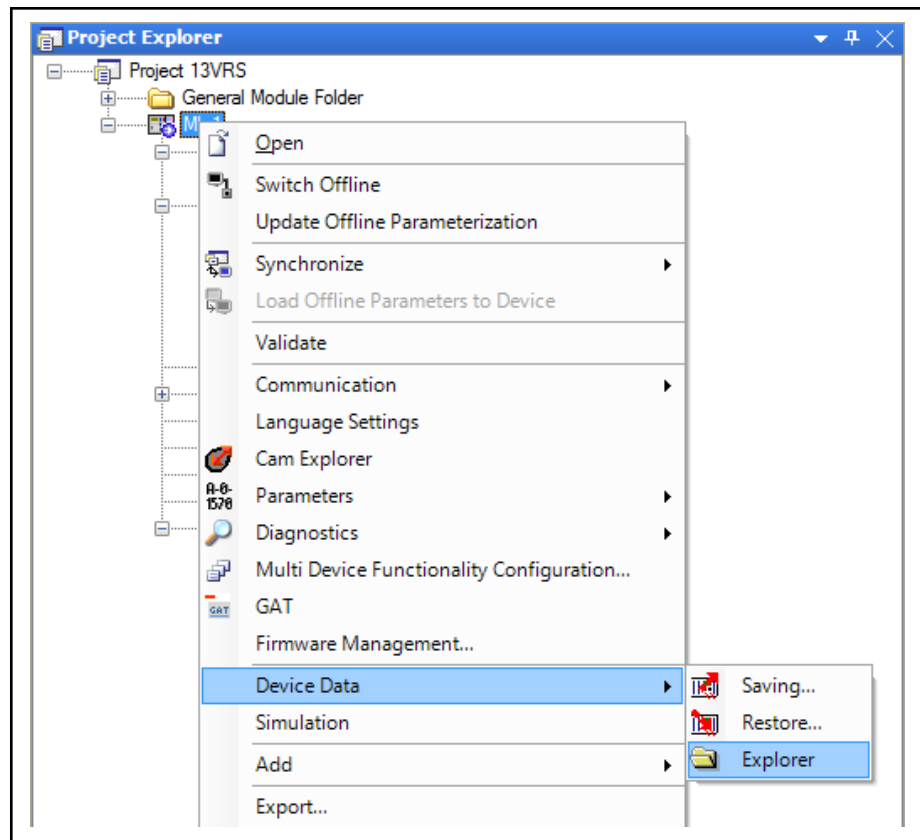


Fig. 4-73: Selecting FTP Explorer

Main window of the Explorer

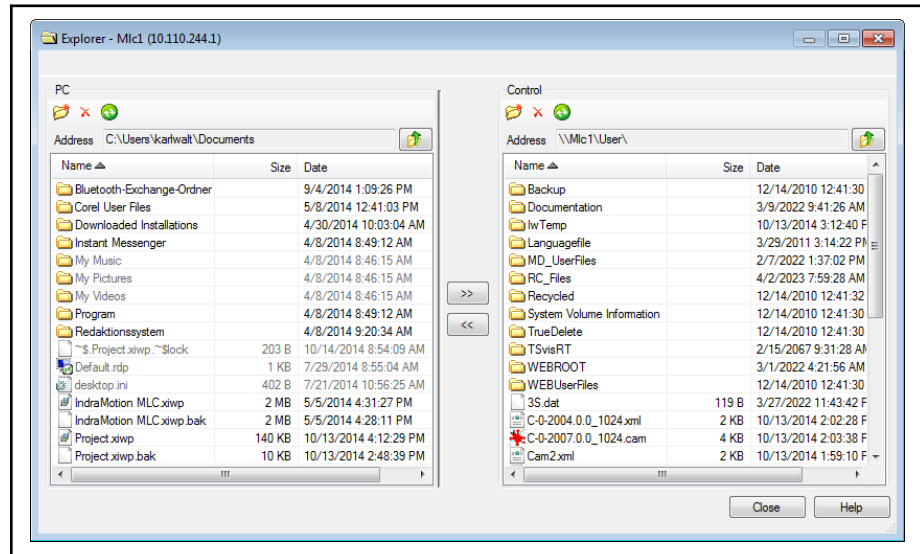


Fig. 4-74: Main window of the Explorer

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 PC and control:

1. Select the files and folders in the left list and click on >> to apply the selected data to the control. Select the files and folders in the right list and click on << to apply the selected data from the control to the PC.
2. Select files and folders and drag them to the 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
Click on  in the toolbar
or
press <DEL>
or
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>
- **Return to previous level:**
Click on the icon  in the toolbar
or
select view and press <Back>

4.4.12 Simulation

The simulation 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.

Control

4.4.13 Exporting

This menu item exports elements to an external file. A wizard executes the export.

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).

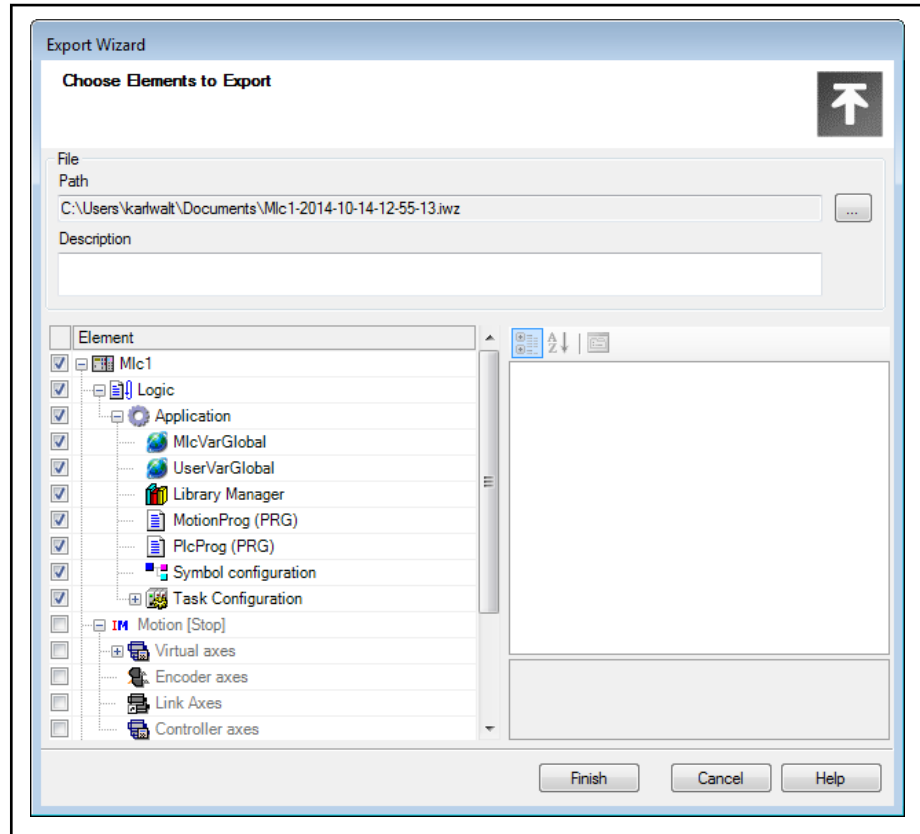


Fig. 4-75: Example: Selecting elements to be exported

In the last dialog, an export summary is displayed. This shows whether the export was successfully completed.

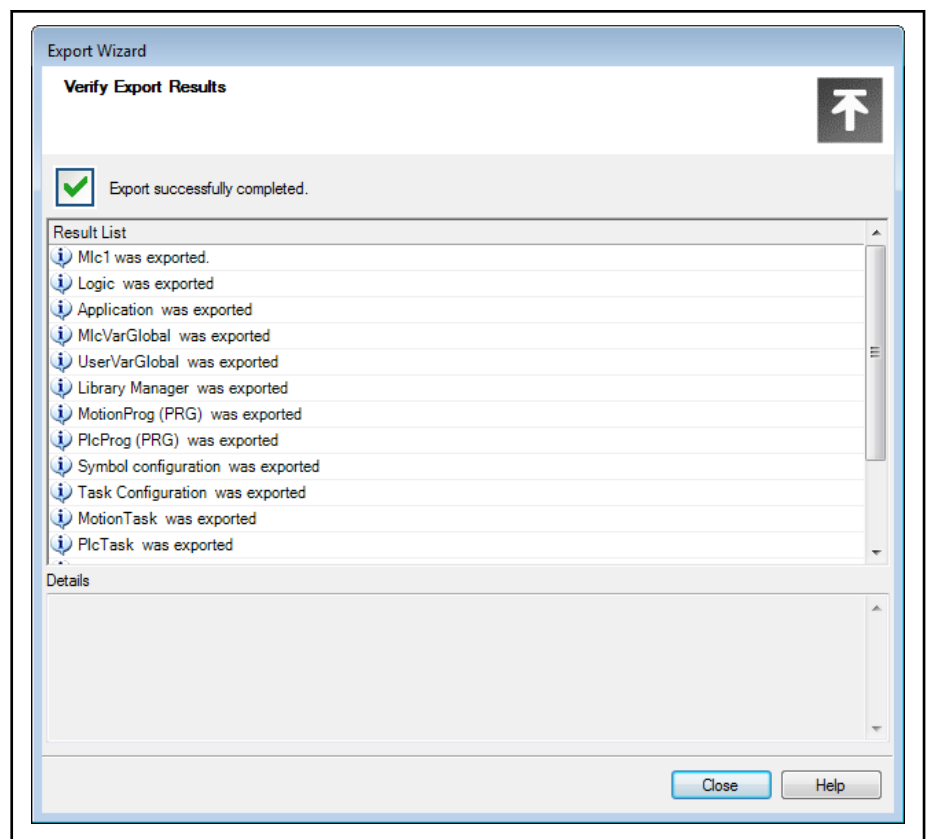


Fig. 4-76: Displaying an export summary

4.4.14 Importing

Use this menu item to import elements from an external export file back to the project. The wizard executes the import.

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:

- **Replace**
Overwrites existing elements, adds new elements and deletes elements not existing in the export file
- **Insert**
Only available when importing 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 -> Mlc11) (see [fig. 4-78 "Displaying an import summary" on page 151](#))
- **Overwrite**
Overwrites existing elements and adds new elements from the export file to the project if required

Control

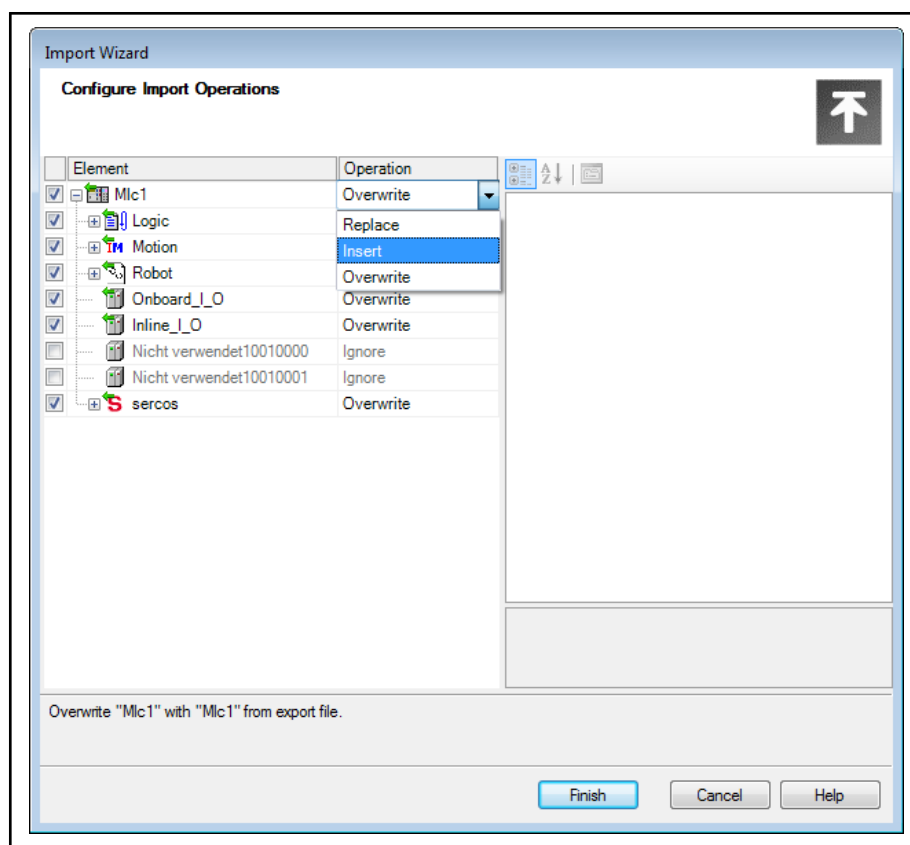


Fig. 4-77: Example: Selecting the area to be imported (in the example the complete device Mlc1)

The last dialog displays an import summary. This shows whether the import was successfully completed.

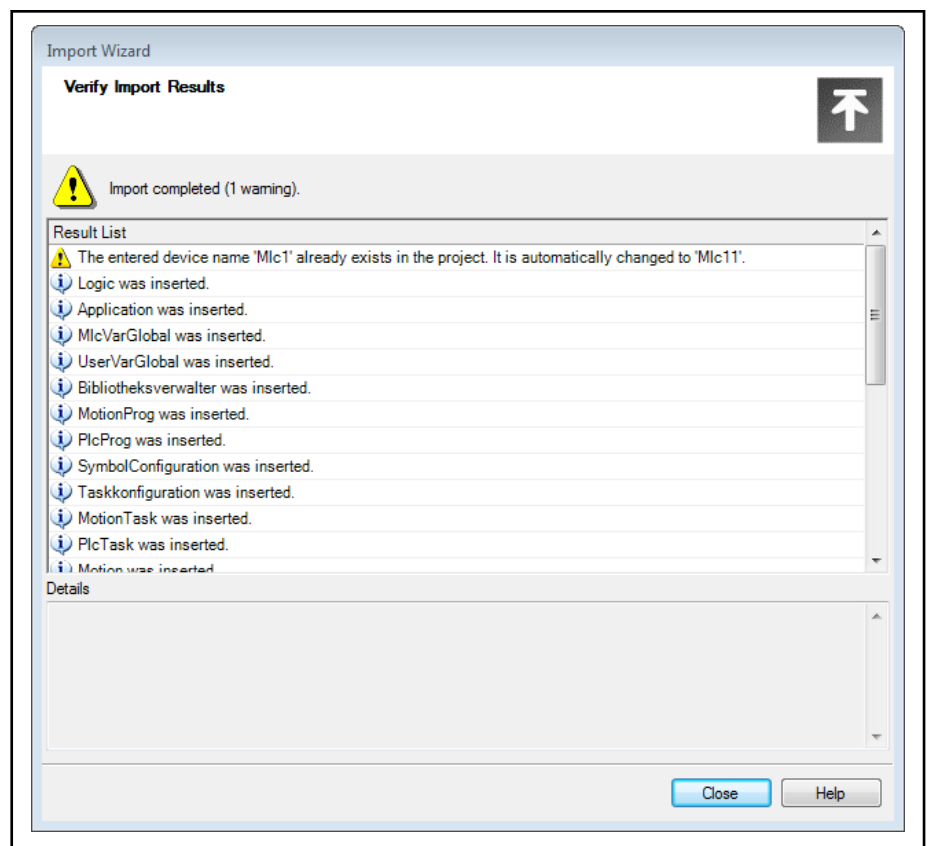


Fig. 4-78: Displaying an import summary

4.4.15 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

Control

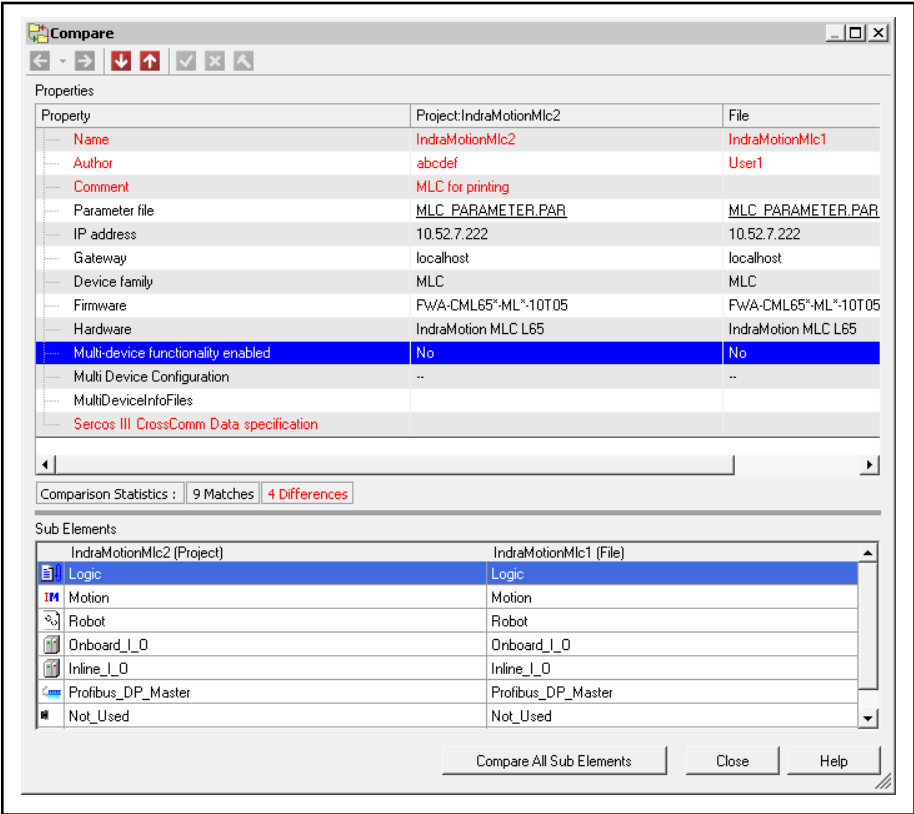


Fig. 4-79: “Compare” dialog

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. IndraLogic XLC L65 and IndraLogic XLC L45), the firmware cannot be merged

Axes

General information

Real axis

An axis can only be compared and merged with an axis of the same type.

Compare 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.

Function modules

General information

A function module can only be compared or merged with a module of the same type.

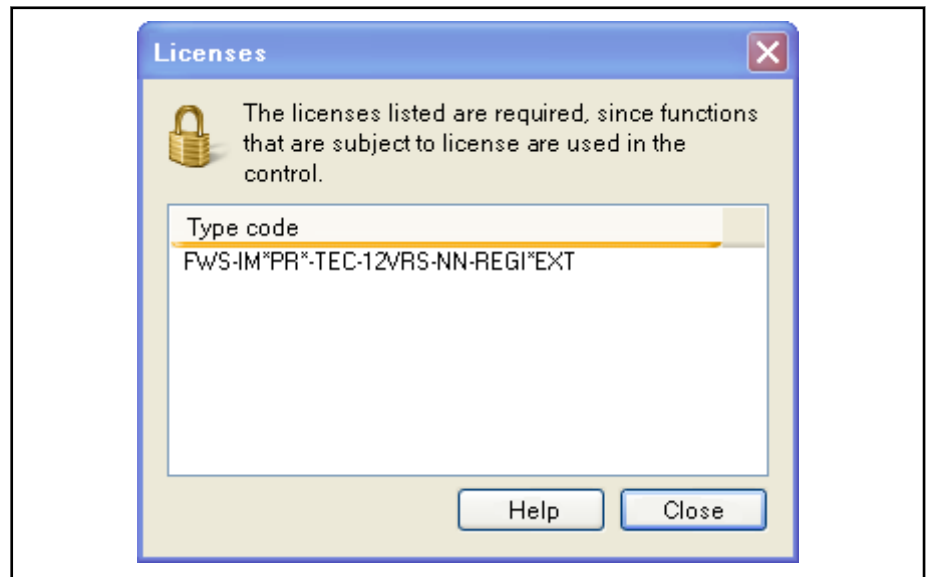
4.4.16 Runtime Licenses

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" dialog via the context menu of the control.



Type code: Type code of used function that is subject to license

Fig. 4-80: Calling the "Runtime Licenses" dialog

4.4.17 Print preview

This menu item can be used for a print preview of the following data categories of a control:

- Configured properties
- Properties of the device
- Parameters

In the first dialog box, the control as well as the data for the print preview can be selected.

Control

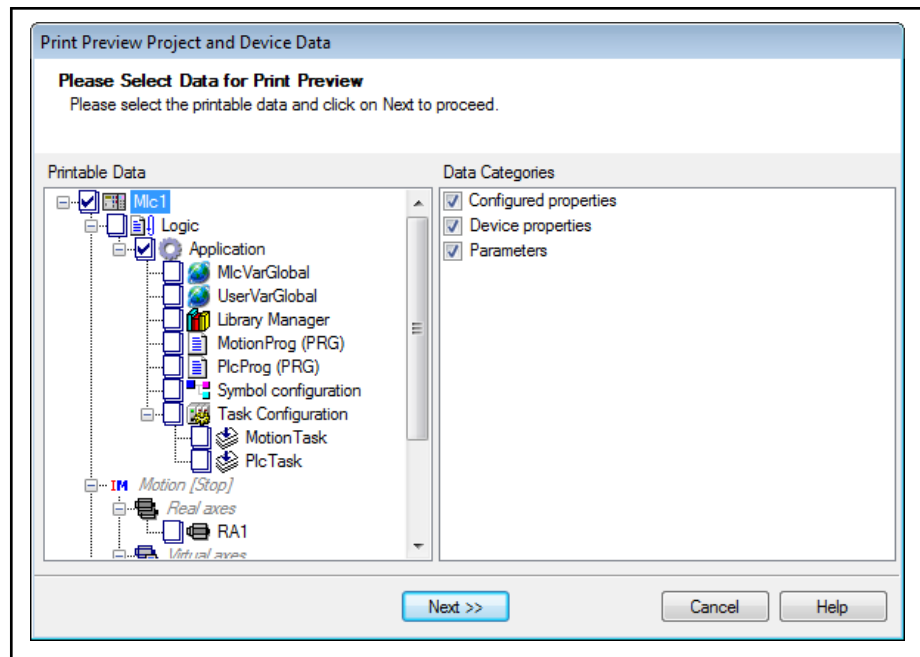


Fig. 4-81: Example: Selecting data for the print preview of an IndraMotion MLC

Configured properties

Includes project details such as project name, creation date, device names, general data, firmware versions and communication settings.

Properties of the device

Includes project details such as project name, creation date, device names, function modules and axis configuration.

Parameters

Lists all parameters, project name, creation date and device name.

In the following dialog box, check the previously selected settings and set the print settings (printer selection, layout, paper size and page setup) via **Page Setup**.... **Finish** starts the print preview.

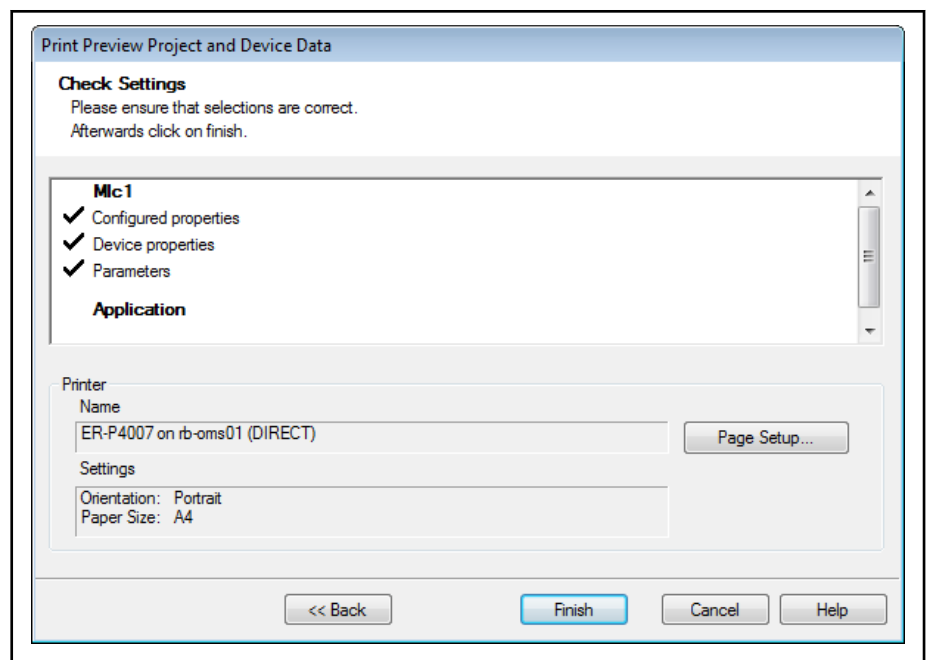


Fig. 4-82: Example: Checking print preview settings for an IndraMotion MLC

The print preview can be displayed in the parameterization mode as well as in the operating mode. Especially in the second case, the print preview can take several minutes.

4.4.18 Printing

This menu item can be used for an unformatted documentation of the following data categories of a control:

- Configured properties
- Properties of the device
- Parameters

In the first dialog box, the control as well as the data to be printed can be selected.

Control

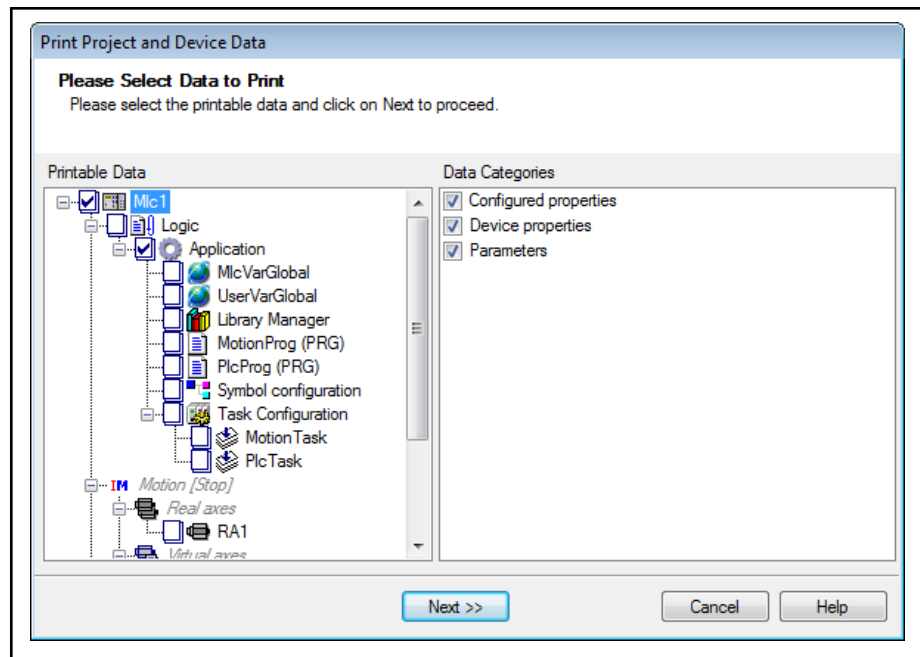


Fig. 4-83: Example: Selecting data to be printed in an IndraMotion MLC

Configured properties

Includes project details such as project name, creation date, device names, general data, firmware versions and communication settings.

Properties of the device

Includes project details such as project name, creation date, device names, function modules and axis configuration.

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/HNC drives 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 set the print settings (printer selection, layout, paper size and page setup) via **Page Setup....** **Finish** starts printing.

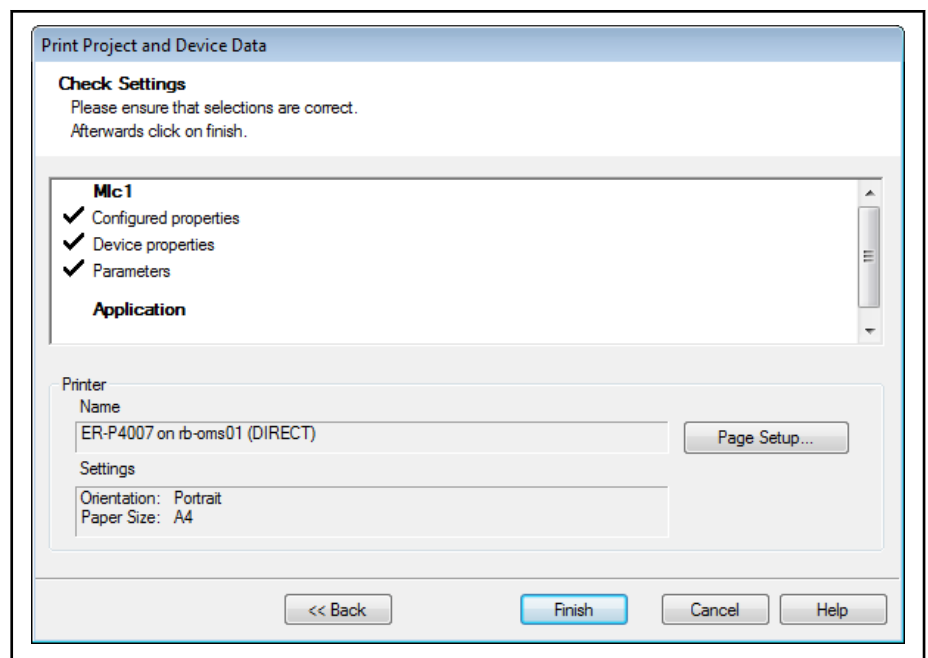


Fig. 4-84: Example: Checking print settings for an IndraMotion MLC

It can be printed in the parameterization mode as well as in the operating mode. Printing can, particularly in the second case, take one or even several minutes.

4.4.19 Control properties

Open this dialog via the context menu item "Properties ..." of a control.



Properties of a control can only be edited if there is no connection to the control (the control is neither online nor logged in).

The "Properties" dialog has three tabs.

"General" tab

The following settings can be changed in this tab:

- Device name: This name has to be unique and IEC-compliant. Otherwise, a warning is displayed
- Comment: This text is displayed as tooltip on the control if neither a warning nor an error is present for the control
- Author: The latest editor of a control property can be entered into this field

"Configuration" tab

The following settings can be changed in this tab and the communication connection can be checked:

Device configuration:

- Device type: System and hardware of the configured control can be changed. The control hardware is written on the type plate of the control.

The following hardware types are available for the IndraLogic XLC:

Control

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

Tab. 4-5: IndraLogic XLC hardware variants



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

If the newly set device type does not support certain features of the source device type, this special configurations are deleted during 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: Generally corresponds to the IndraWorks Suite version
- Firmware release: Indicates the firmware version within the version
The release can be subsequently updated taking the set project version into account. That means that the version of the compatibility mode has to be increased if necessary.

Ethernet communication:

- IP Address: The entered control address has to be valid. Otherwise, there is no communication to the control
The IP address corresponds to the IP address set in section [chapter 3.3.4 "Setting the IP address" on page 38](#).
- PLC gateway: Use "localhost" if the PLC gateway is on the computer IndraWorks is installed on. Otherwise, enter the IP address of the computer and of the gateway
- PLC communication: From the version 13VRS, it is possible to choose between two PLC communication types
 - **TCP** is recommended due to a better firewall handling and no required broadcast (no broadcast towers). Due to the protocol, TCP is slower than UDP.
 - Due to internal routing functions, **UDP** is mandatory for SafeLogic
- Connection test:
First, the connection test checks whether a control with this IP address (ping) exists and responds with the device name of the control and the current user name (author). Both can differ from the device name and

the author listed in the previous figure. The PLC communication is checked in the second step

IndraLogic:

- Secure online mode: Enables additional queries for the PLC start and stop as well as for forcing values in the PLC monitoring

"Interfaces" tab

Displays an overview on the current configuration of the hardware interfaces of the control. The interfaces can also be edited directly via the Project Explorer at the interface nodes



Before the configuration of the "Properties" dialog is applied with **OK**, it is checked whether the interface configuration has changed. If an existing element is deleted by the change, an information dialog with a change preview is displayed. The conversion can still be cancelled. The changed configuration is only applied when the change preview is confirmed with **OK**.

4.5 Version control of the control

4.5.1 Introduction

All of the controls in IndraWorks listed below support version control:

- IndraLogic XLC L25
- IndraLogic XLC L45
- IndraLogic XLC L65

For the documentation of the version control system in IndraWorks, refer to "Rexroth IndraWorks 13VRS Engineering" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

4.5.2 Special Features

Project Management

There are some special features in the version control of MotionLogic devices.

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.

Control

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 4.4.8 "Multi-device" on page 126](#).

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.

4.6 Complete data backup

4.6.1 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.

The following data is archived:

- Configuration
- Control parameters
- Drive parameters
- Libraries, device descriptions
- Compact Flash contents

4.6.2 Complete backup

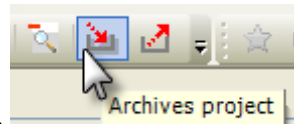


To ensure the archiving of the **current control data**, switch to **on-line**.

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:

Archive Project

Archive properties
Enter a file name (without path) for the target archive and password-protect the archive if required. Subsequently, click on "Next >>".

Archive name:
Project

Comment:

Password:

Confirm password:

Next >> Cancel Help

Fig. 4-85: Enter the archive name

3. Select the storage location (file system and/or FTP server):

Archive Project

Archive target settings
Please select if you want to save the archive on a local file system and/or on an FTP server. Click on "Next >>".

Archive name:
Project.zip

Filing on File System
☒ Store archive on file system
Target directory:
C:\Users\karlwalt\Documents

Filing on FTP Server
☐ Store archive on FTP server (device or computer)
Device name, host name or IP address:
Mlc1

<< Back Next >> Cancel Help

Fig. 4-86: Select the storage location

4. Selection of the components to be archived. If the project is selected, the complete project is archived. If the control (here "Mlc1") is selected, settings for the archiving can be made.

Control

Select on the right, whether the compiled program and the default libraries are included in the archive:

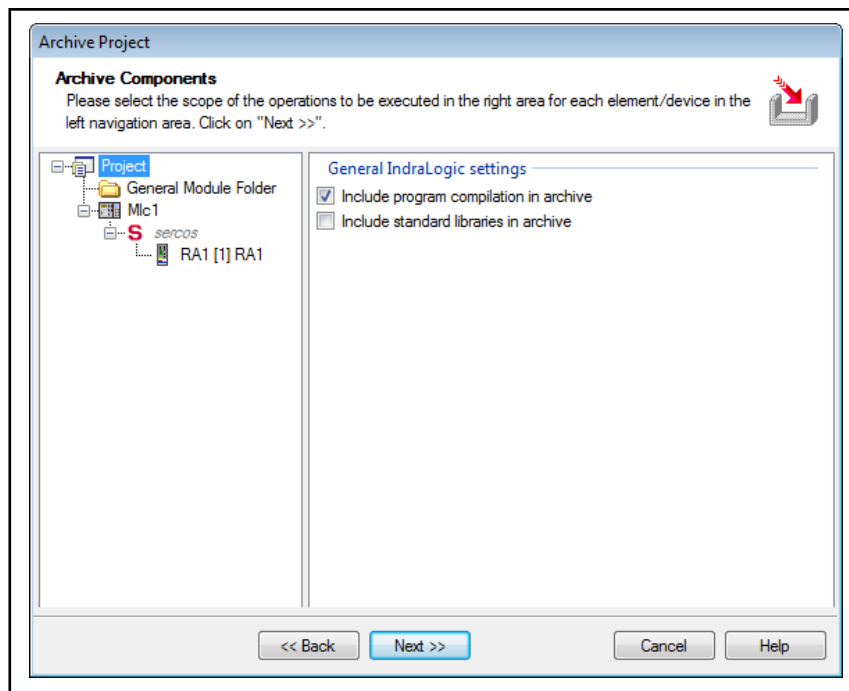


Fig. 4-87: Selection of the components to be archived

5. 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

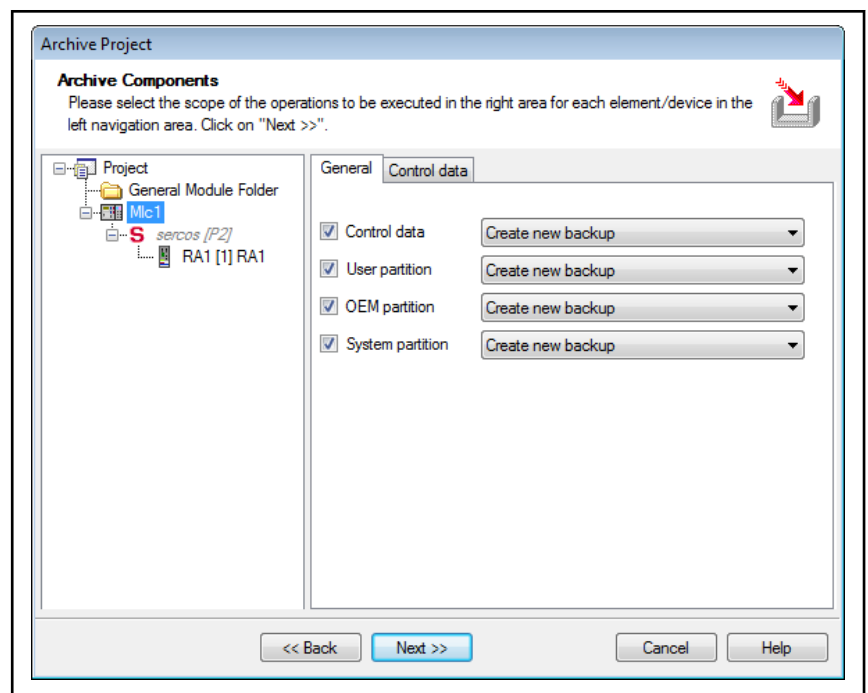


Fig. 4-88: Selecting components to be archived (1)

The parameters to be saved can be specified on the "Control data" tab. It is only possible to make an entry on this page if "Create new backup" was selected on the "General" tab at "Control data".

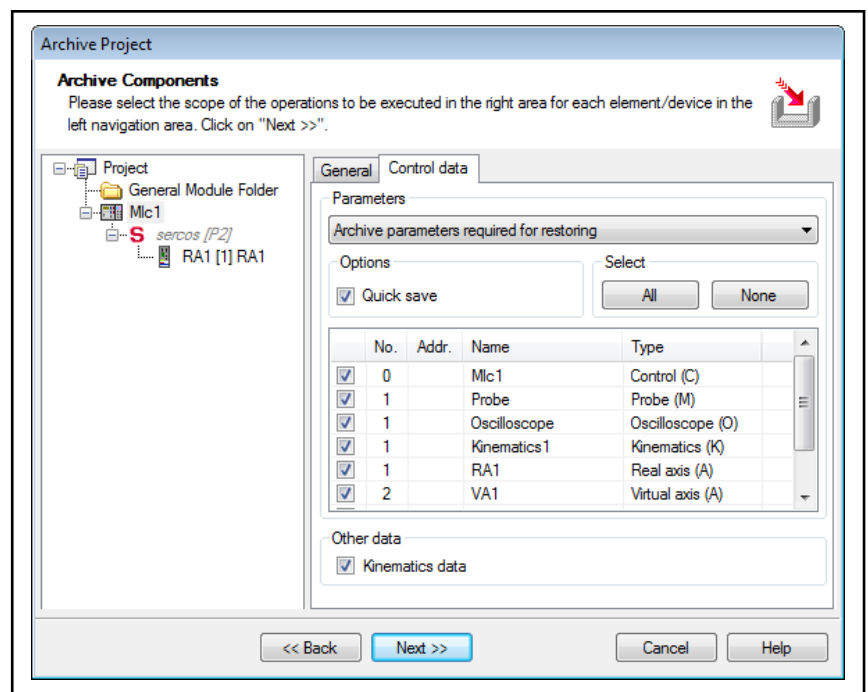


Fig. 4-89: Selecting components to be archived (2)

6. Check user inputs:

Control

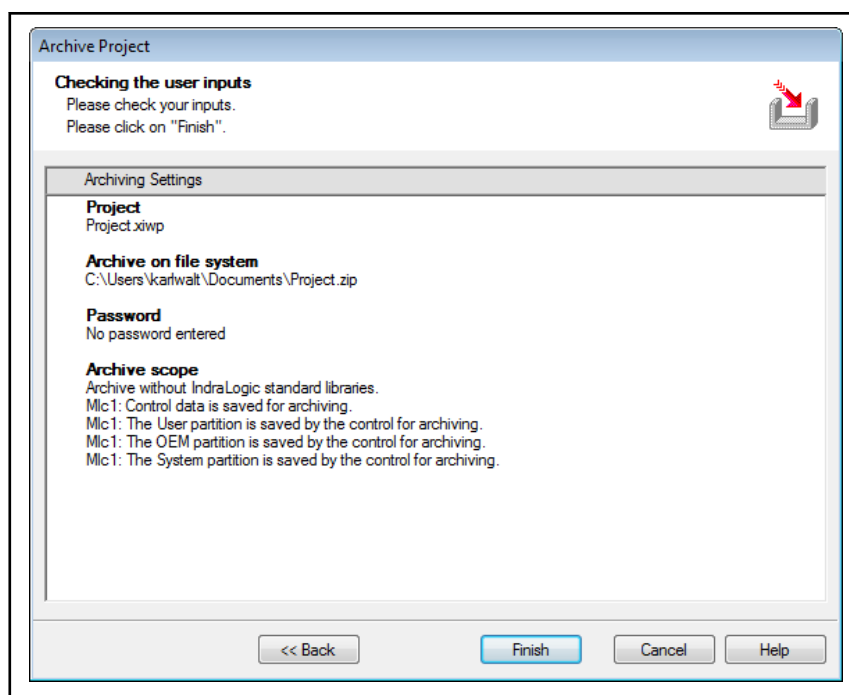


Fig. 4-90: Checking user inputs

7. Archiving with progress bar:

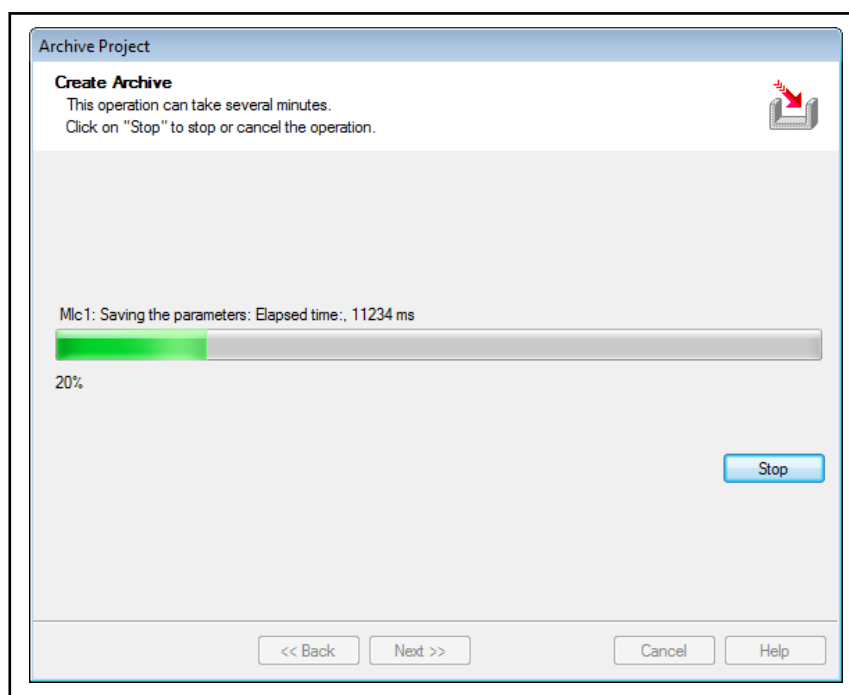


Fig. 4-91: Archiving...

8. Display of a summary:

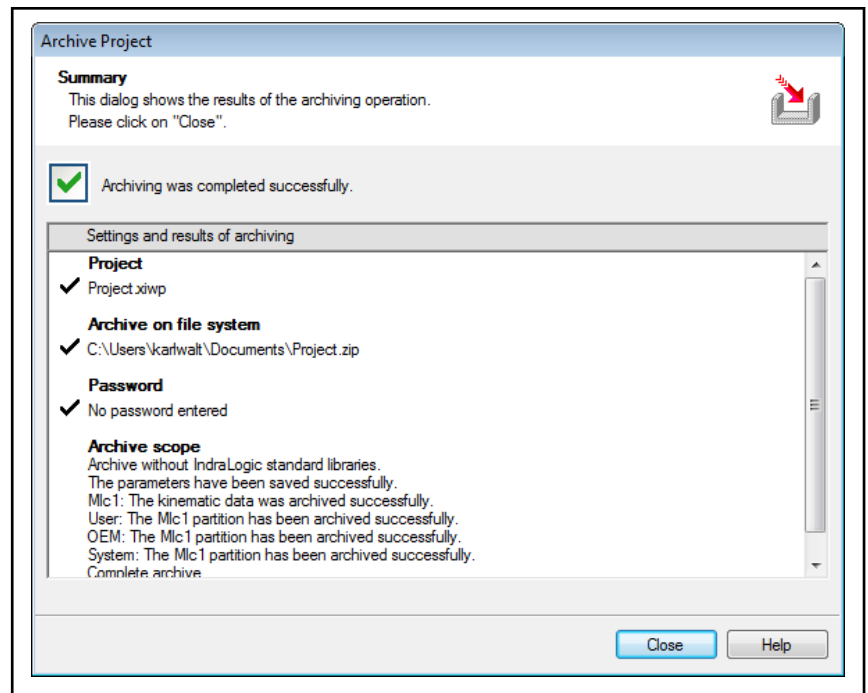
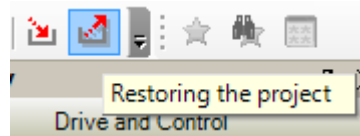


Fig. 4-92: Summary of the project backup

4.6.3 Restoring a project

The project is restored as follows:

1.



Select the **Drive and Control** icon in the IndraWorks toolbar.

2. Specify the restoration type, that is the source (file system or FTP server) of the project to be restored:

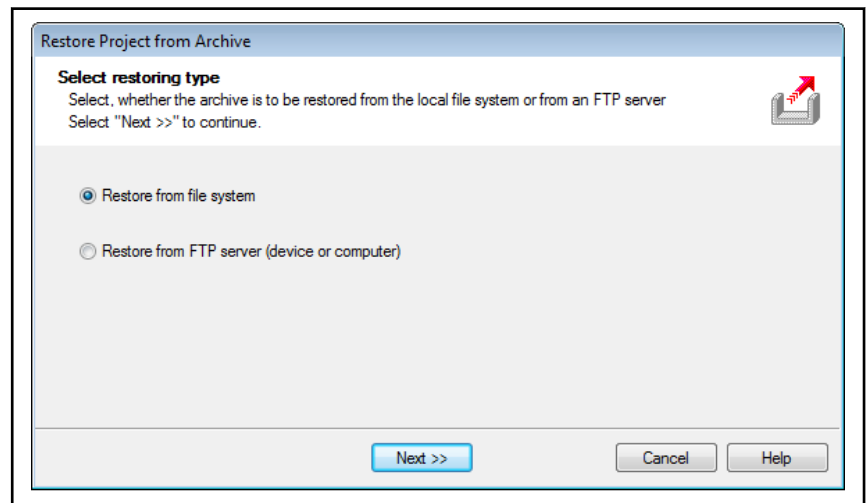


Fig. 4-93: Specifying the restoration type

3. Selecting the archive to be restored:

Control

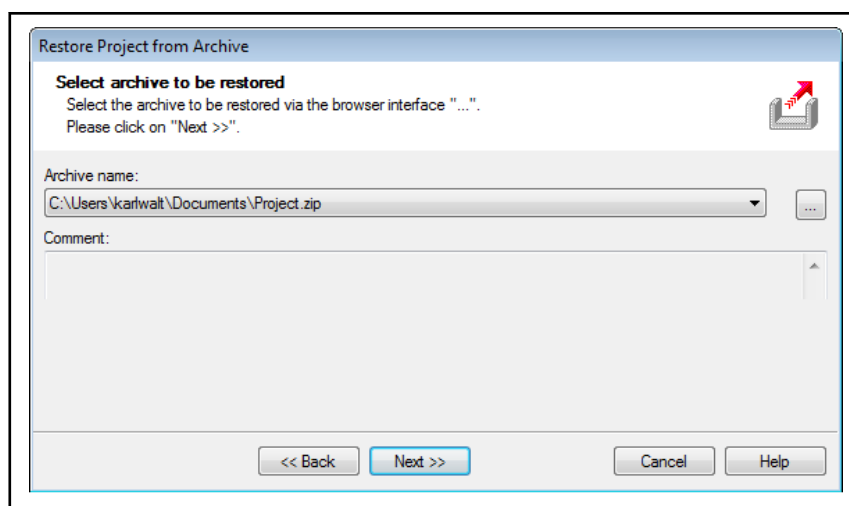


Fig. 4-94: Selecting the archive to be restored

4. Selecting the directory in which the archive is to be restored. A folder with the project name is automatically created.

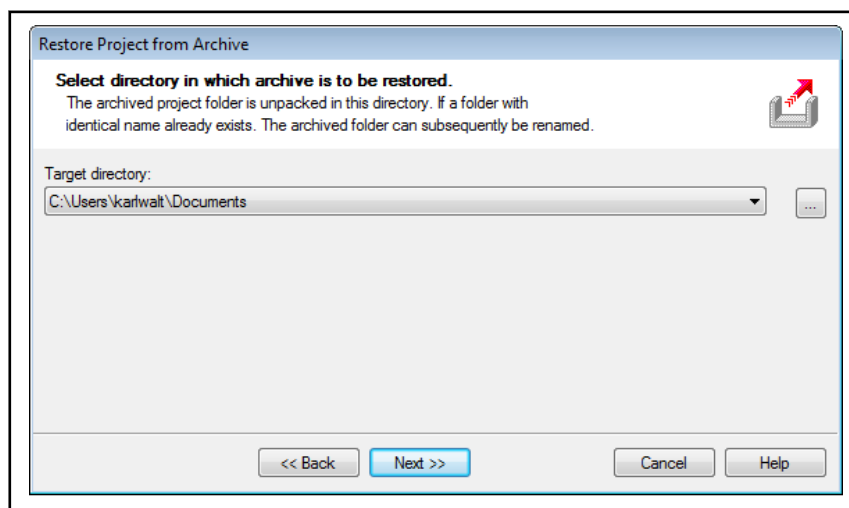


Fig. 4-95: Selecting the directory...

5. Checking the previous specifications:

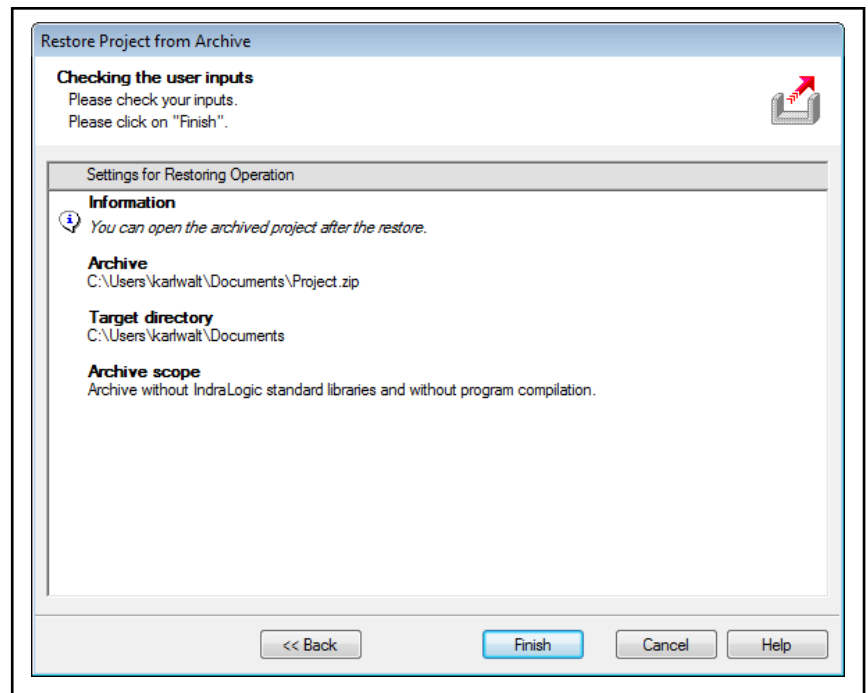


Fig. 4-96: Checking the specifications made

If a password was specified when the project was archived, a dialog to enter the password opens:

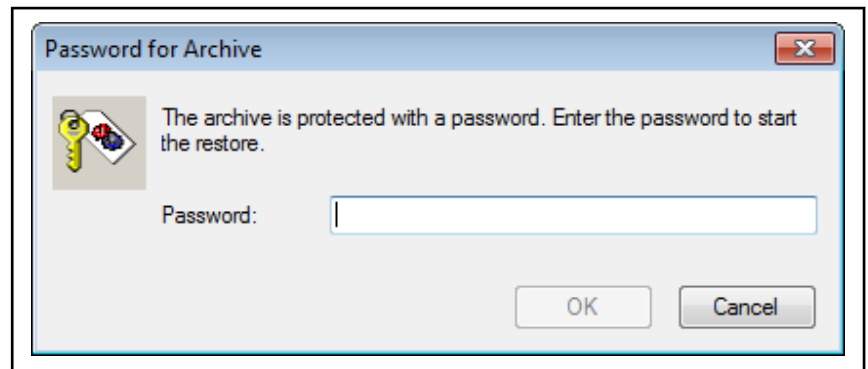


Fig. 4-97: Input dialog for the password

6. A summary of the project restoration is displayed. The selected archive, the target directory, the project directory and the archive size are displayed:

Control

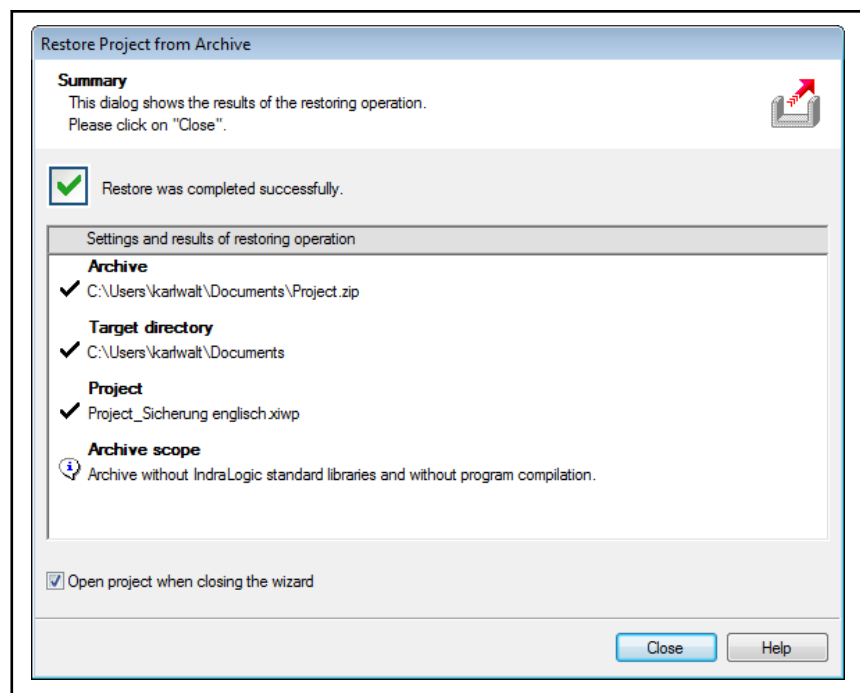


Fig. 4-98: Displaying a summary

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:

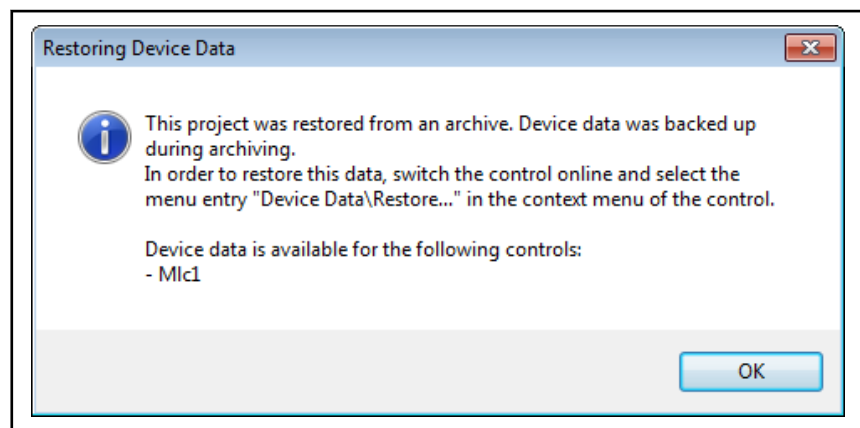


Fig. 4-99: Message to restore device data

The menu item **Restore Device Data** is located in the context menu of the control (see [chapter "Restoring" on page 143](#)).

5 PLC programming

5.1 General information

This chapter provides a brief overview on the PLC programming on the IndraLogic XLC system. Following a short introduction, the chapter deals specifically with the library manager and the task system. I/O access by the PLC is described in the following. The chapter ends with a section on the memory organization of the control and memory access from the PLC.

For a more detailed functional description of the IndraLogic PLC, refer to "Rexroth IndraWorks 13VRS IndraLogic 2G Programming Instruction" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

5.2 Creating PLC programs

5.2.1 Opening "PlcProg object"

Double-click to open the "PlcProg" object in the Project Explorer. Alternatively, click on "Open" to open the "PlcProg" object via the context menu. The workspace of the "PlcProg" object opens. The upper area is used to edit the variables (declaration part) and the lower area to edit the instruction (instruction part).

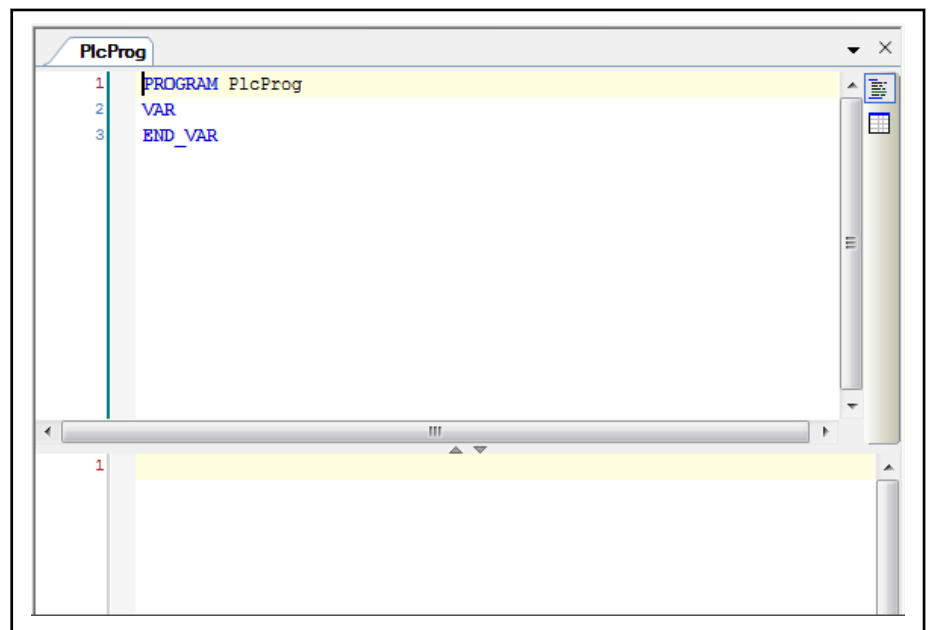


Fig. 5-1: PlcProg

5.2.2 Declaring variables

Enter the line "i:=i+1;" in the code section (lower area) and confirm with "Return". The "Auto Declare" input field opens automatically to declare the variable "i".

PLC programming

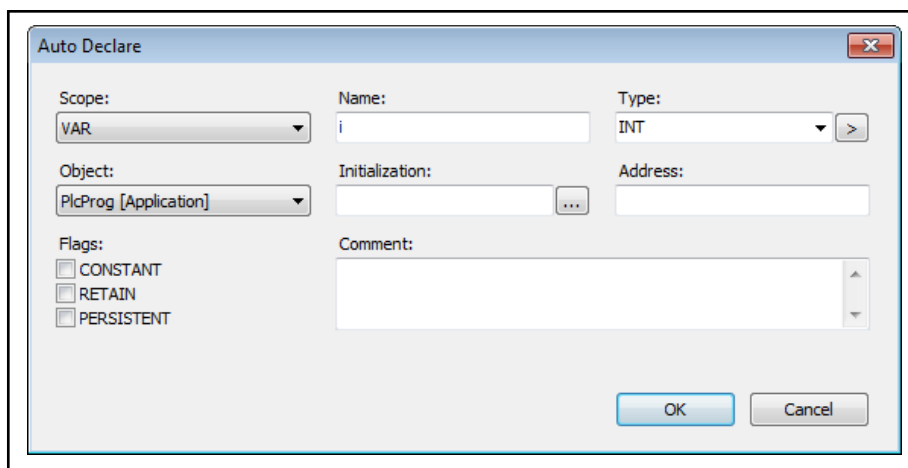


Fig. 5-2: Declaring variables

Confirm the specified values with **OK**. The variable "i" of the "integer" type is now declared.

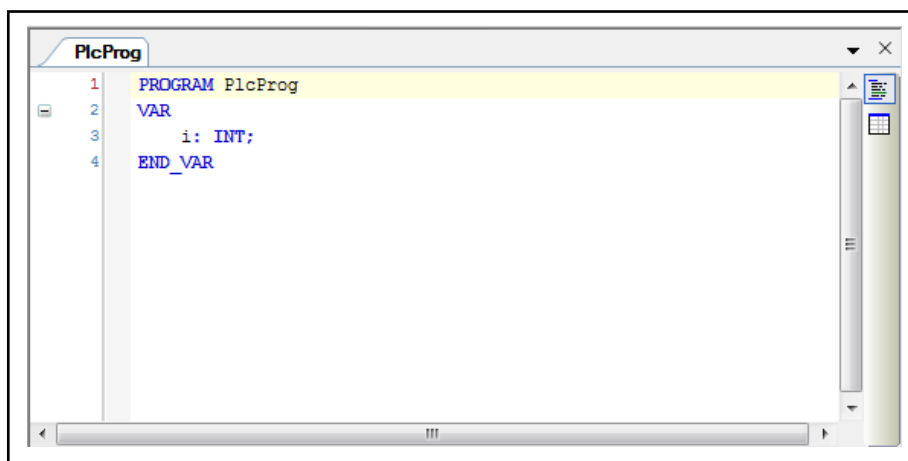


Fig. 5-3: Declared variable



The variable "i" can also be declared in the upper workspace by manual entry.

5.2.3 Editing instruction

It is possible to edit instructions in the lower area of the window.

The following programming languages are available:

- ST (structured text)
- LD (ladder diagram)
- IL (instruction lists)
- CFC (continuous function chart)
- FBD (function block diagram)
- SFC (sequential function chart)

For a detailed description, refer to the online help under IndraLogic 2G Editors.

5.2.4 Compiling programs

To compile the project, select **Create ► Generate Code**.

PLC programming

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.

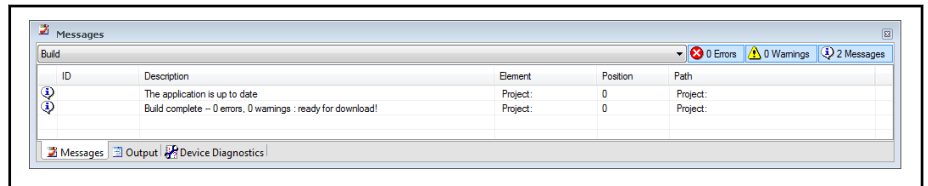


Fig. 5-4: Compiling a program

5.2.5 Loading a PLC program to the control

To load the project to the control, select **Debug ► Login**

The following dialog is displayed.

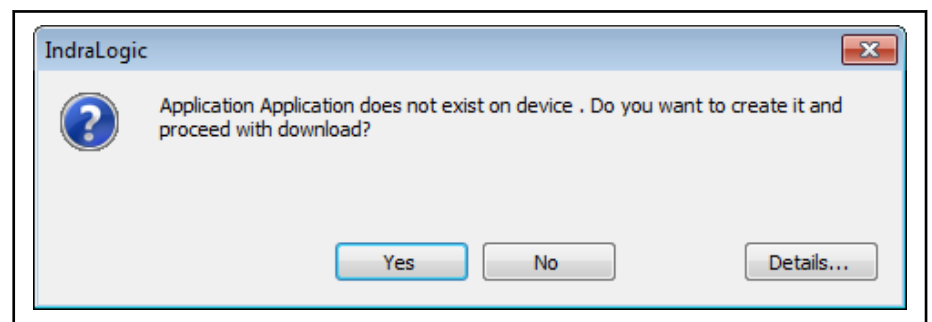


Fig. 5-5: "Load program" dialog

Select **Yes** to confirm the dialog. You are now connected online to your device.

Select **Debug ► Start** in the menu bar to start the PLC program.

Select **Debug ► Generate Boot Application** in the menu bar to create a boot project. If this command is executed online, the compiled project is stored on the control so that the control can automatically load it upon restart.

Select **Debug ► Stop** in the menu bar to stop the PLC program.

Select **Debug ► Logout** in the menu bar to log out of the control.

5.3 Library management

5.3.1 General information

Libraries can provide functions, function blocks, data types, global variables and even visualizations to 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".

Libraries in a project are managed in the library manager. The previous installation on the system was made via 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.

Refer to the following general information on:

- [chapter 5.3.2 "Library overview" on page 172](#)
- [chapter 5.3.3 "Installing on the system and integrating into a project" on page 173](#)
- [chapter 5.3.4 "Referenced libraries" on page 174](#)

PLC programming

- [chapter 5.3.5 "Library versions" on page 174](#)
- [chapter 5.3.6 "Unique access to library function blocks \(namespace\)" on page 174](#)
- [chapter 5.3.7 "Creating libraries" on page 175](#)
- [chapter 5.3.9 "Integrating external code" on page 177](#)

5.3.2 Library overview

Libraries available for the control:

Name	Description	Documentation ¹⁾
RIL_CheckRtv	Automatic monitoring of internal range exceedances	Rexroth IndraWorks 13VRS Basic Libraries IndraLogic 2G
RIL_CommonTypes	Data types of the PLCopen function blocks, system-comprehensive	
RIL_Utilitys	General IL functions and function blocks	
RIL_LoopControl	Controller function blocks	
RIL_Inline	Diagnostic functions for Rexroth Inline modules	Rexroth IndraWorks 13VRS Field Buses
RIL_ProfibusDP_02	Profibus DP-V1 services, diagnostics, Sync and Freeze	
RIL_ProfinetIO	Functions for Profinet I/O	
RIL_ProfinetIODevice	Functions for the control as Profinet I/O device	
RIL_EtherNetIP	Diagnostic and communication services for Ethernet IP device	
RIL_EtherNetIPAdapter	Functions for Ethernet IP adapter	
RIL_SercosIII_IO	Functions for Sercos III I/O	
RIL_FieldbusTypes	Internal library for field bus data types	
RIL_NetXLoad	Internal library to operate the Profibus/Profinet interface	
RIH_CMLx	Hardware information CML45/L65 (operating hours, temperature, el. type plate, display output)	Rexroth IndraWorks 13VRS Basic Libraries IndraLogic 2G
ML_Base	Basic system library	Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Libraries
ML_PLCopen	PLCopen function blocks for Sercos drives	
MP_PLCopenFieldbus	PLCopen function blocks for field bus drives	
ML_TechInterface	AxisInterface	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Ge- neric Application Template

¹⁾ Refer to [chapter 1.4 "Required and supplementing documentations" on page 18](#)

Name	Description	Documentation ¹⁾
ML_TechBase	Function blocks for basic technology functions	Rexroth IndraMotion MLC 13VRS Technology Libraries
ML_TechMotion	Function blocks for general Motion applications	
ML_TechRegi	Function blocks for register controllers	
RMB_TechWinder	Function blocks for winders	
ML_SeqProg	Functions for sequential programming	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Se- quential Program- ming
RIL_SocketComm	Functions to communicate via TCP/IP and UDP	Rexroth IndraLogic XLC IndraMotion MLC 13VRS Se- quential Program- ming
RIL_ModbusTCP	Functions for ModbusTCP-compatible HMI devices	
RIL_Diagnosis	HMI diagnostic functions	
RIL_HMI_Uilities	Functions to load HMI M-keys	
IL_VCP_DP	Functions to trigger small control panels (VCP) via Profibus-DP	Rexroth IndraWorks 13VRS Basic Libraries IndraLogic 2G

Tab. 5-1: Basic Libraries

5.3.3 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, install it on the local system in such a repository. This is done in the Library Repository dialog.
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 name 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 used in addition to this default category in libraries created in IndraLogic XLC, 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 opened in the "Project Information" dialog to select a category. For more information on the categories, refer to chapter 5.3.7 "Creating libraries" on page 175.
Library manager	Libraries are integrated into a project using the Library Manager. In a "default project", it is automatically assigned to the default device first. But it can also

1) Refer to [chapter 1.4 "Required and supplementing documentations" on page 18](#)

PLC programming

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 5.3.4 "Referenced libraries" on page 174](#).

If the library is not encrypted and the ".library*" file is also present, double-click on the respective entry to open the library POU's listed in the library manager.

Searching sequence

If a library function is addressed in the project, the libraries and repositories are searched in the order in which the libraries and repositories are listed in the "Library Repository" dialog, see also [chapter 5.3.6 "Unique access to library function blocks \(namespace\)" on page 174](#).

5.3.4 Referenced libraries

A library can reference other libraries (referenced libraries). The nesting depth is arbitrary. If such a "father" library is integrated into a project, the libraries referenced in the project are also available. However, only those libraries integrated into the "father" library in the global "General module" folder are available!

Visibility

When creating a library project that references other libraries, it can be defined in the Properties of each integrated library how it will respond once it is integrated into a project using the "father" library.

1. Its visibility in the library manager, indented below the "father" library, can be disabled. 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, 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, use this option only when creating a container library project!

5.3.5 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, the version to be used (a specific one or always the latest one) depends on the definition in the Library Properties. If multiple versions are located on different levels (with referenced libraries for example), enter the corresponding namespace (see [chapter 5.3.6 "Unique access to library function blocks \(namespace\)" on page 174](#)) for a unique access to library function blocks.

5.3.6 Unique access to library function 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 local project function blocks and to function blocks available in integrated and libraries referenced in these. In such cases, add the namespace in front of the function block name to ensure uniqueness.

- The default namespace for a library is defined in the Library properties. If the namespace 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 access 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 enabled, "module1" is 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 causes a compiler error, since the call is no longer unique; see points (1) and (4)
(4)	In the object "module1" defined in the "General module" folder	"module1.var1"
(5)	In the object "POU1" defined in the "General module" folder	"POU1.var1"

Tab. 5-2: Namespaces

5.3.7 Creating libraries

A library can be created and saved from the global "General module" folder. Highlight the "General module" folder and select the menu item **Create Library...** in the context menu. Alternatively, create a library via the main menu items **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 on the "Library Info" object or use **Open** in the context menu to open the "Project Information" dialog.

PLC programming

In Project Information on the "Summary" tab, enter a "title", a "version" and the "company". If the "default namespace" differs from the library name, it can be defined here. Using the option to enter a "category" is also recommended, since this category is later used 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, afterwards, 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 to the entire repository to display the categories.

- Note the behavior of referenced libraries 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 information on a function block to library users, 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

5.3.8 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 items **Project ▶ Data Transfer**, it can be selected whether these references are to be retained, replaced or deleted. If the references 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 the system libraries 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. Thus, 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 online help under "Data Transfer".

5.3.9 Integrating external code

External and internal libraries or library function blocks, late linking

- An "external library", in contrast to an internal (IndraLogic XLC library), is a library file that is programmed outside of IndraLogic XLC 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 IndraLogic XLC library as external library later on, i.e. when the application is first operated on the runtime system. In addition, it is now possible to define the late linking individually for every library function block. For this purpose, enable the property in the object properties ("external implementation") of one or all function blocks.

5.4 Task system

5.4.1 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 programs are then processed successively by the task as defined. 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** (free wheeling): The task runs continuously. 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 task occurs

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 after timeout, 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 179](#)).



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 IndraLogic XLC system:

- **EVENT_OP_MODE_MOTION_CYCLIC:** This event is triggered by a Motion cycle. It is used to synchronize a PLC program with the calculation of axis motions. With IndraLogic XLC controls with a Sercos interface, the "Motion-Task" 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. This event is only available on controls with a Sercos bus.

PLC programming

- **EVENT_Local_Input_Bitx**: This event is triggered by the onboard inputs of the control. Every input is mapped on Bit "x" x according to its number. A rising edge at the input triggers the respective event (see also [chapter "On-board and Fast I/Os" on page 192](#)).

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:

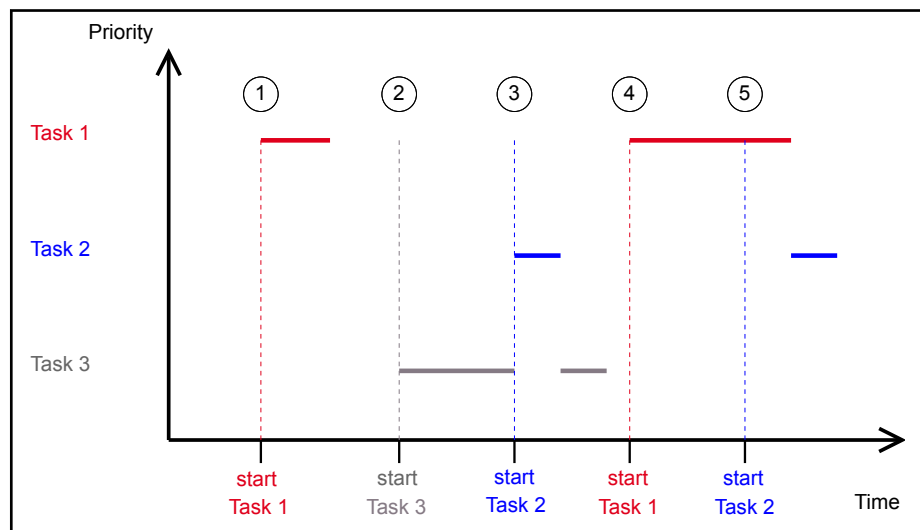


Fig. 5-6: Processing different task priorities

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 executable task 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 this task 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. The priority 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:

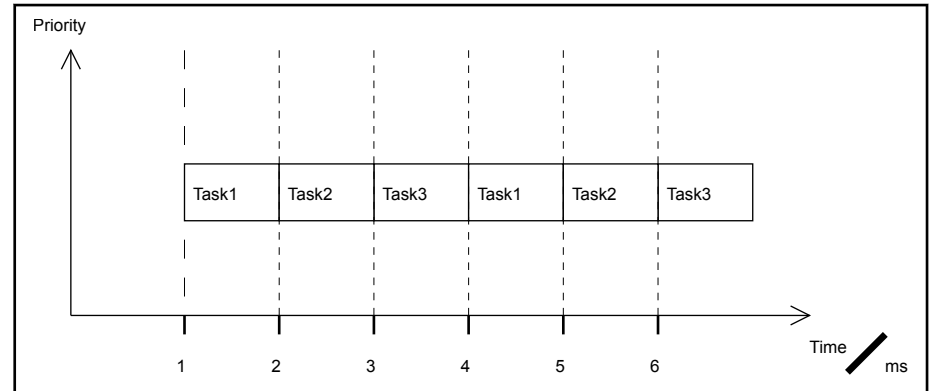


Fig. 5-7: Round-robin method

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. The watchdog 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 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 these global 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.

5.4.2 Node "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.

PLC programming



If the task configuration was deleted by mistake, the task can be subsequently inserted into the application via the context menu using the **Add ► Task...** 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 ► 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 [chapter 5.4.3 "Task editor"](#) on [page 183](#) also providing an online view. The configuration options depend on the target system.

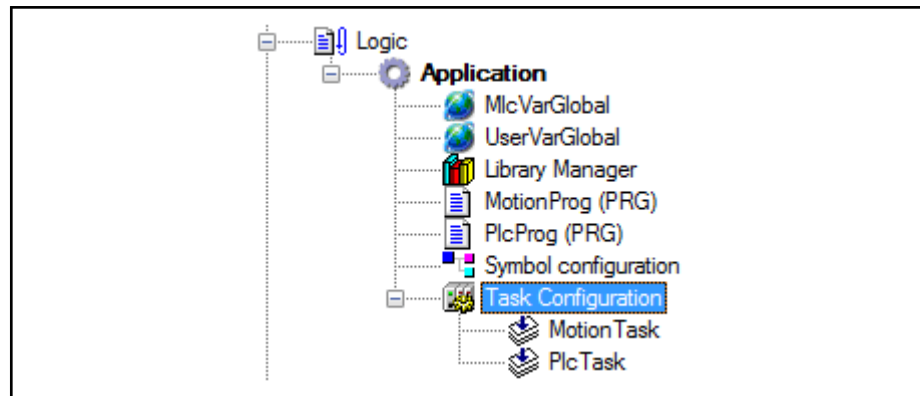


Fig. 5-8: Task configuration below an application in the Project Explorer

In online mode, the "task editor" provides a monitoring view with information on cycles, cycle times and task status.

Default tasks

The following 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. The MotionTask is of high priority and 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 5.4.8 "System tasks"](#) on [page 193](#)).

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

Tab. 5-3: 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 these tasks 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 5.4.8 "System tasks" on page 193](#)).

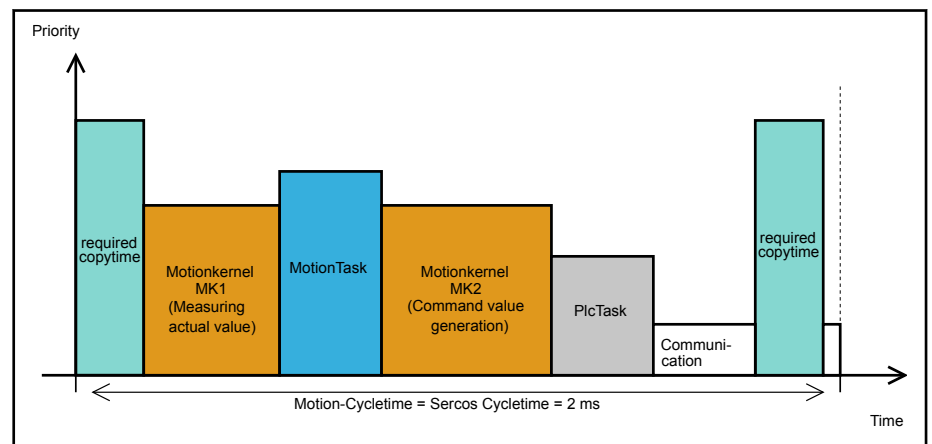


Fig. 5-9: Default setting before adding

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.

PLC programming

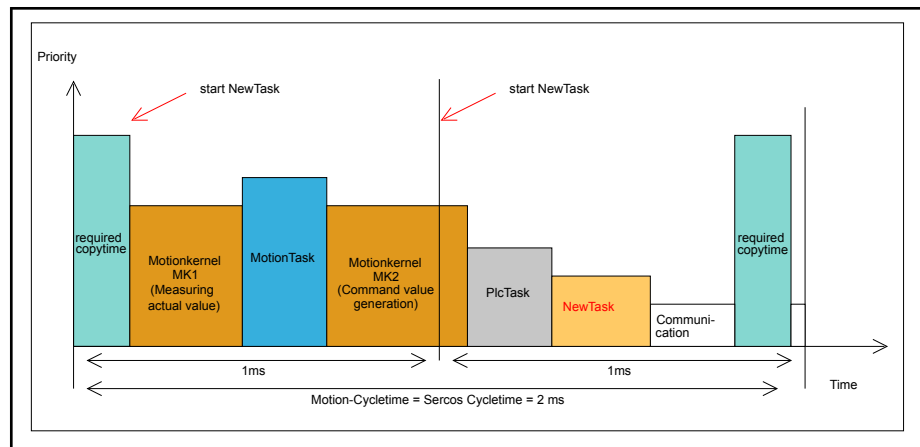


Fig. 5-10: Example 1, creating new PLC task

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 task has already been started twice up to this point at which the NewTask is processed. The task runs only 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".

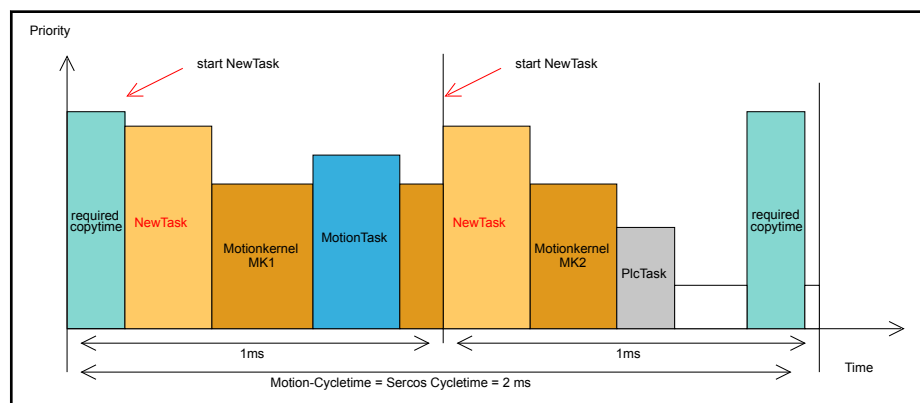


Fig. 5-11: Example 2, the priority of "NewTask" of example 1 is now set to 1

The "NewTask" has now a higher priority than the other tasks. Thus, the task 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 "SercosSyncTask" 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.

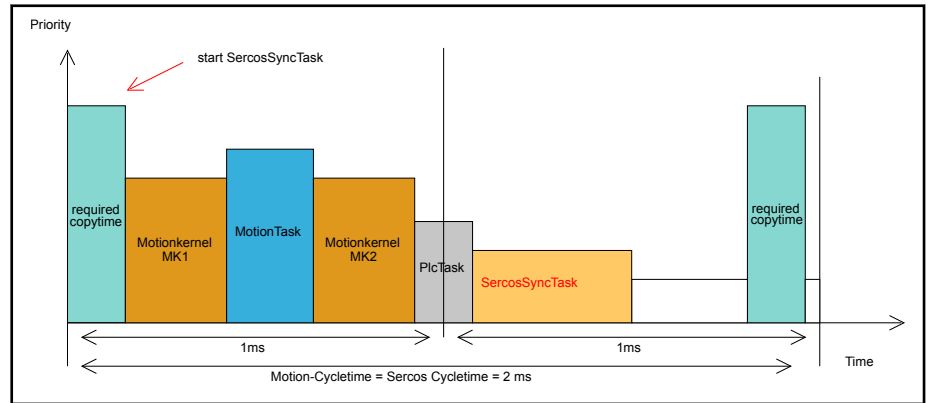


Fig. 5-12: Example 3, creating another task

In this example, the Sercos and the Motion cycle time are equal. The "SercosSyncTask" is started at the beginning of the Sercos cycle. However, this task can only run if all higher priority tasks have been completed.

5.4.3 Task editor

General information

Use the task editor to configure individual tasks 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

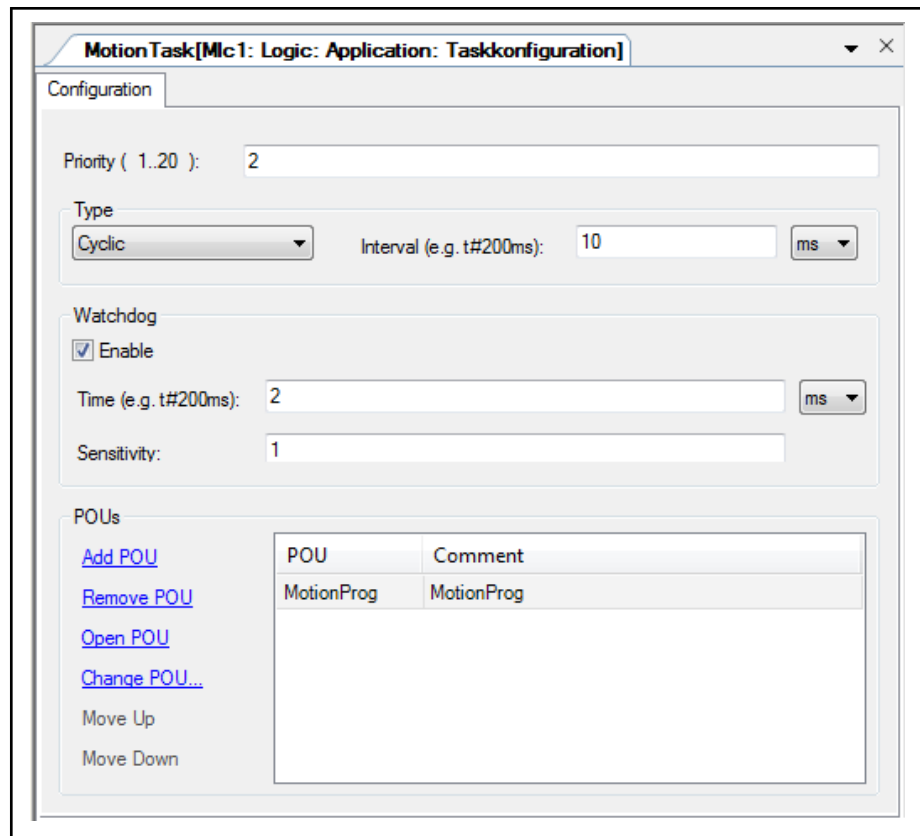


Fig. 5-13: Task editor, dialog to configure a task

The configuration options available in the dialogs depend on the target system used.

Configuring a task

Define the parameters as requested in [fig. 5-13 "Task editor, dialog to configure a task" on page 184](#):



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:
- **Free wheeling:** 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 (Event):** 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 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. "t#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: 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.

- **Enable:** 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 while in stop" (see [chapter "PLC settings" on page 98](#)) 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 times the watchdog time elapsed after which the task is stopped and a watchdog exception is output



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.

PLC programming



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

5.4.4 "Properties" dialog

Double-click to select the node  and the "Properties" dialog opens (refer to the following figure).

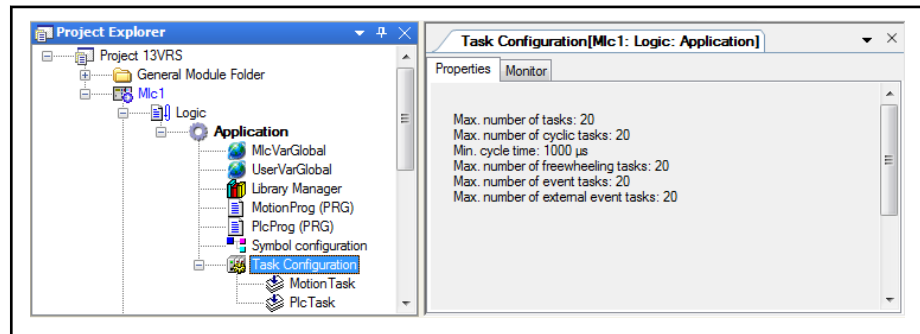


Fig. 5-14: Task configuration, "Properties" dialog, example

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.

5.4.5 Task list/configuration

The task list enables the data of all tasks to be seen at a glance and the tasks to be prioritized.

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 this data is read from the parameters of the control.

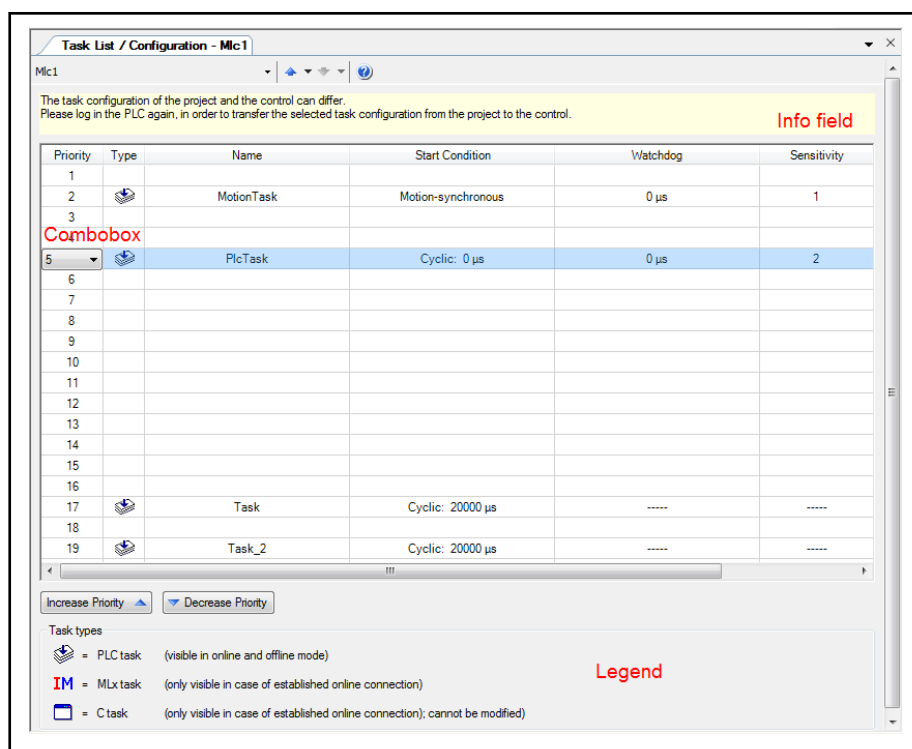


Fig. 5-15: Task list in offline mode

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 into the control to apply 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 (e.g. starting condition) has to be configured via individual dialogs of the tasks (see [chapter 5.4.3 "Task editor" on page 183](#)). 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 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.

PLC programming

5.4.6 Task viewer

This dialog shows the chronological sequence of task processing on the control. For the dialog, go to the context menu item **Diagnostics ► Task Viewer** of the control.

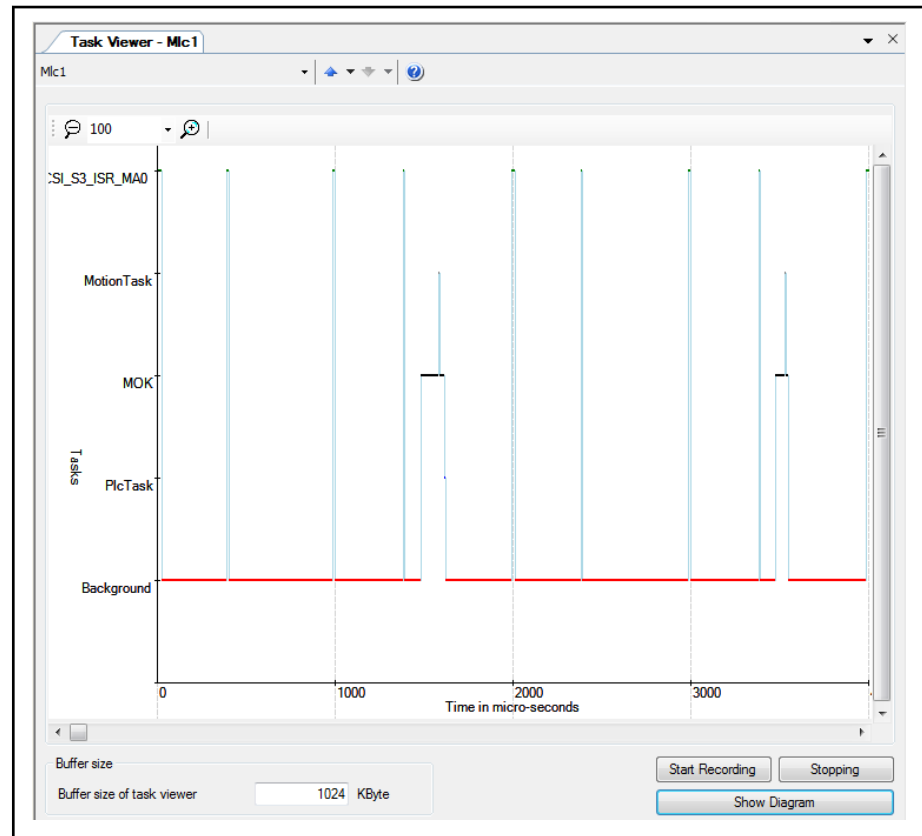


Fig. 5-16: "Task Viewer" dialog

Availability	The "Task 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 Stopping button is pressed. 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  buttons. 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 internal system tasks are shown by a red line. The following tasks are displayed in the "Task Viewer".

- PLC tasks: These are all tasks created in the task configuration (see [chapter 5.4.2 "Node Task configuration" on page 179](#)). 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
- SI_S3-ISR_MAO: 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

5.4.7 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. The turn around times 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. A PLC task 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 task is sequentially processed (e.g. setting outputs). At the end of the PLC task, a consistent output image is generated and the newly calculated data becomes valid. At this point, outputs are set for example.

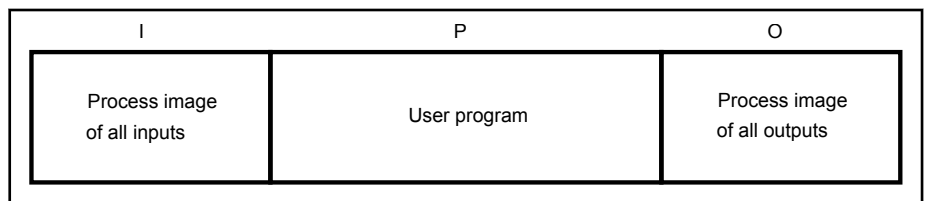


Fig. 5-17: Basic PLC task structure

[fig. 5-18 "Components of I/O turn around time" on page 190](#) 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. Once the output signal is 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

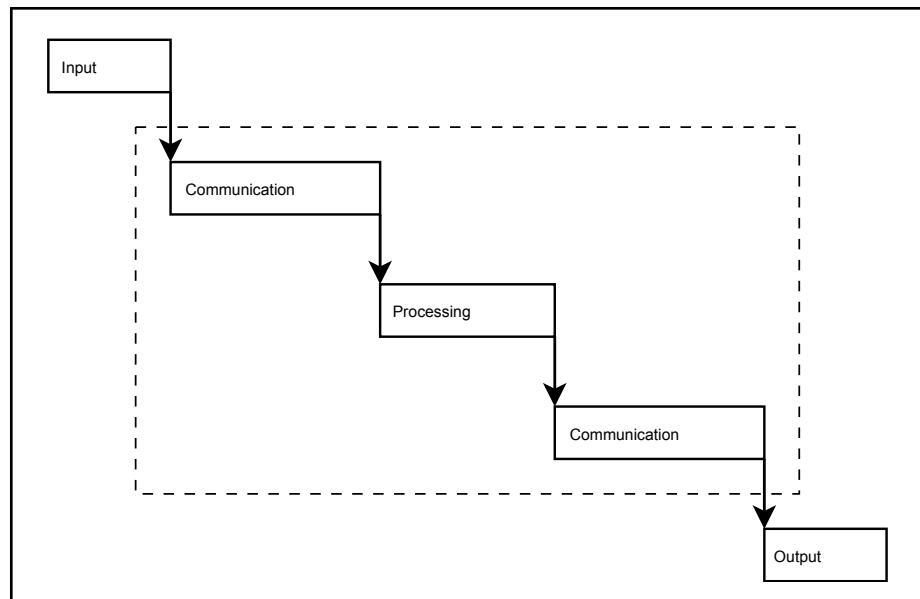


Fig. 5-18: Components of I/O turn around time

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 fast I/O data is processed by the PLC task in which the I/O data is integrated. I/Os are only processed by the PLC (see [chapter "Configuring a task" on page 184](#)). If there is I/O data 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.

Bus cycle task

The point in time of the I/O mapping can be set via the bus cycle task. The configuration can be made:

- globally in the PLC settings
- locally at the corresponding field bus master



The local setting at the local field bus master dominates the global setting.

For more information on this topic, refer to [chapter 5.5 "I/O access" on page 200](#).

Safe state

If the PLC task responsible for the I/O mapping is not running or stopped due to an event, the outputs go into a defined safe state. This state can be set globally in the "PLC Settings" dialog. There are the following options:

- Update I/O while in stop: The inputs and outputs are still updated if the I/O task is stopped.
- Behavior of the outputs at stop:
 - Retain values: The values of all outputs remain. No outputs are re-set
 - Set all outputs to default: The outputs are set to their respective default value
 - Execute program: A settable PLC program is executed. It can set the outputs to a safe state
- Update all variables in the devices

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 here 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). This 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 5.4.1 "General information" on page 177](#))

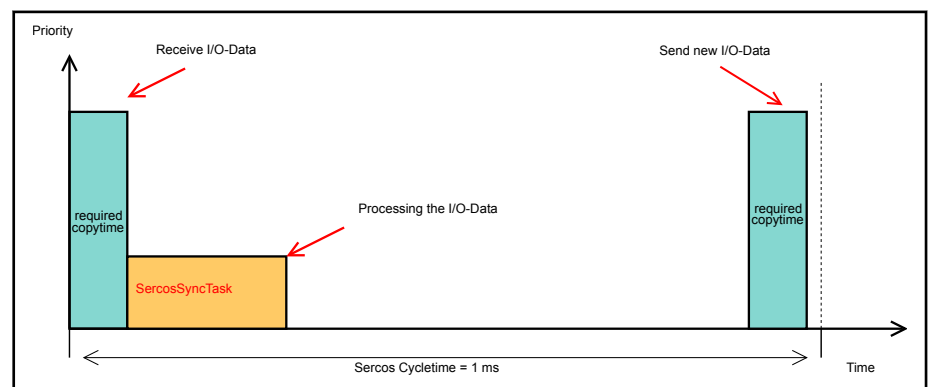


Fig. 5-19: Processing I/O data

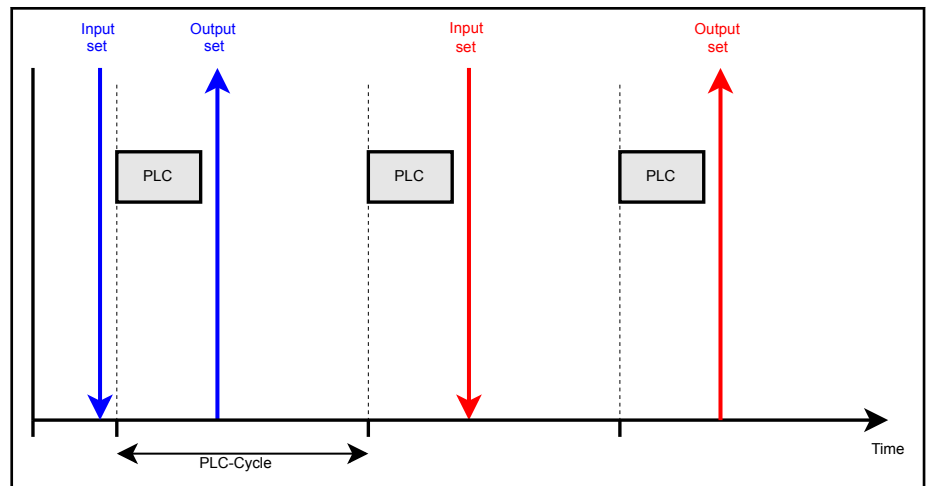


Fig. 5-21: I/O turn around time, Onboard I/O and Fast I/O

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.

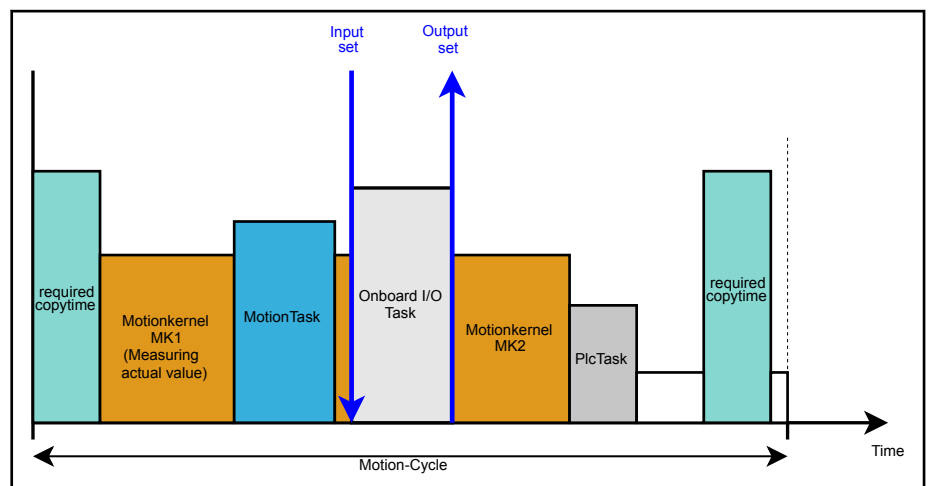


Fig. 5-22: I/O turn around time; task directly started via external event

As shown in the figure, a very short I/O turn around time can thus be achieved.

5.4.8 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.

PLC programming

In the default setting, an entire Motion cycle runs as follows:

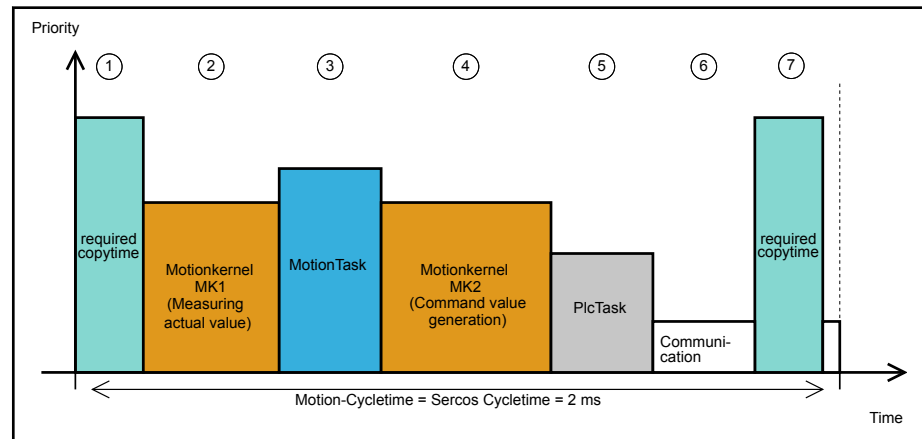


Fig. 5-23: Overview on the temporal behavior

In the default setting, the Motion cycle time (C-0-0400) is equal to the Sercos cycle time (C-0-0503) = 2 ms.

- 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 data is copied, there is time left to perform background tasks up to the next cycle start



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 [chapter 7.2 "Cycle Times" on page 323](#)).

The priority of the Motion calculation can be changed via the parameter C-0-0403 in the priority table (see [chapter 5.4.5 "Task list/configuration" on](#)

[page 186](#)). 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 item "Cycle Times" of the Motion kernel (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.

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.

Command value generation

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" has been 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 item "Cycle Times" of the Motion node.

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.

PLC programming

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 high-priority PLC task:

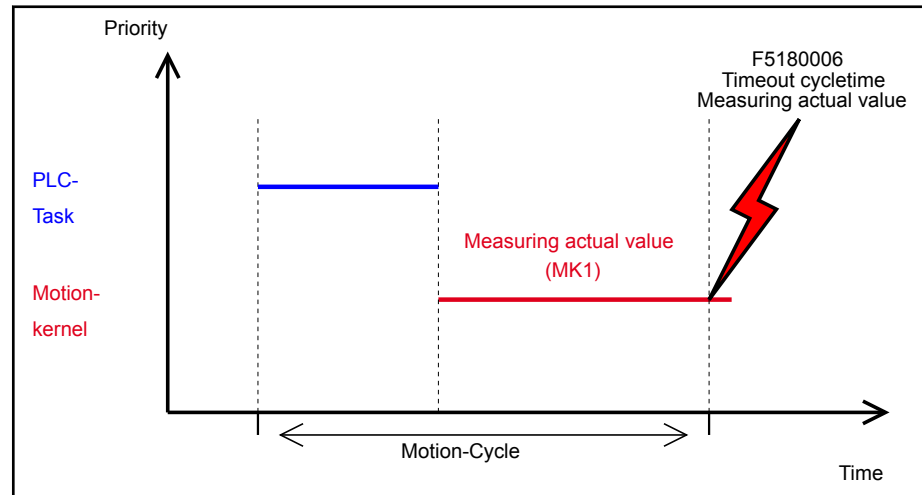


Fig. 5-24: Actual value detection blocked by a higher priority PLC task

Error "Timeout cycle time of Motion kernel" was triggered since the Motion kernel could not be completely processed:

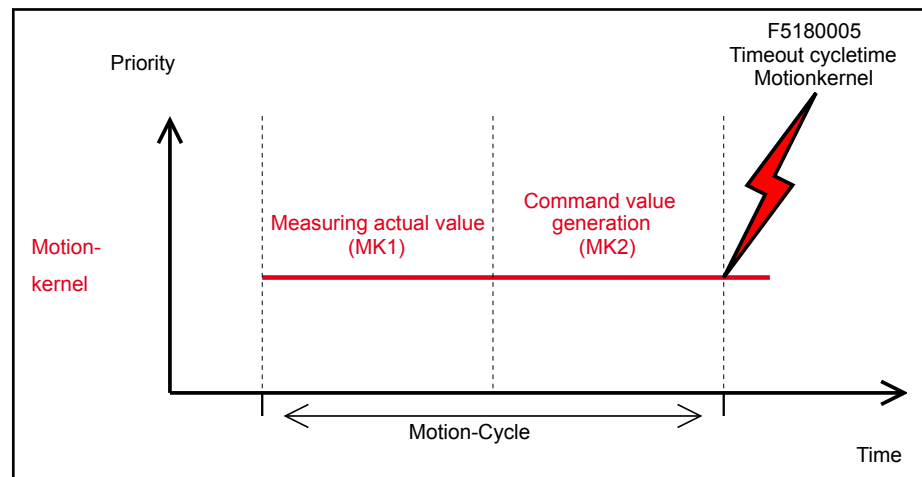


Fig. 5-25: Motion kernel not completely processed

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 5.4.5 "Task list/configuration" on page 186](#)).

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). Consequently, ensure that only the tasks with a priority higher than the priority of the Motion kernel are configured and whose runtime is not that long and for which it is required (e.g. for short turn around times).
- **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:

PLC programming

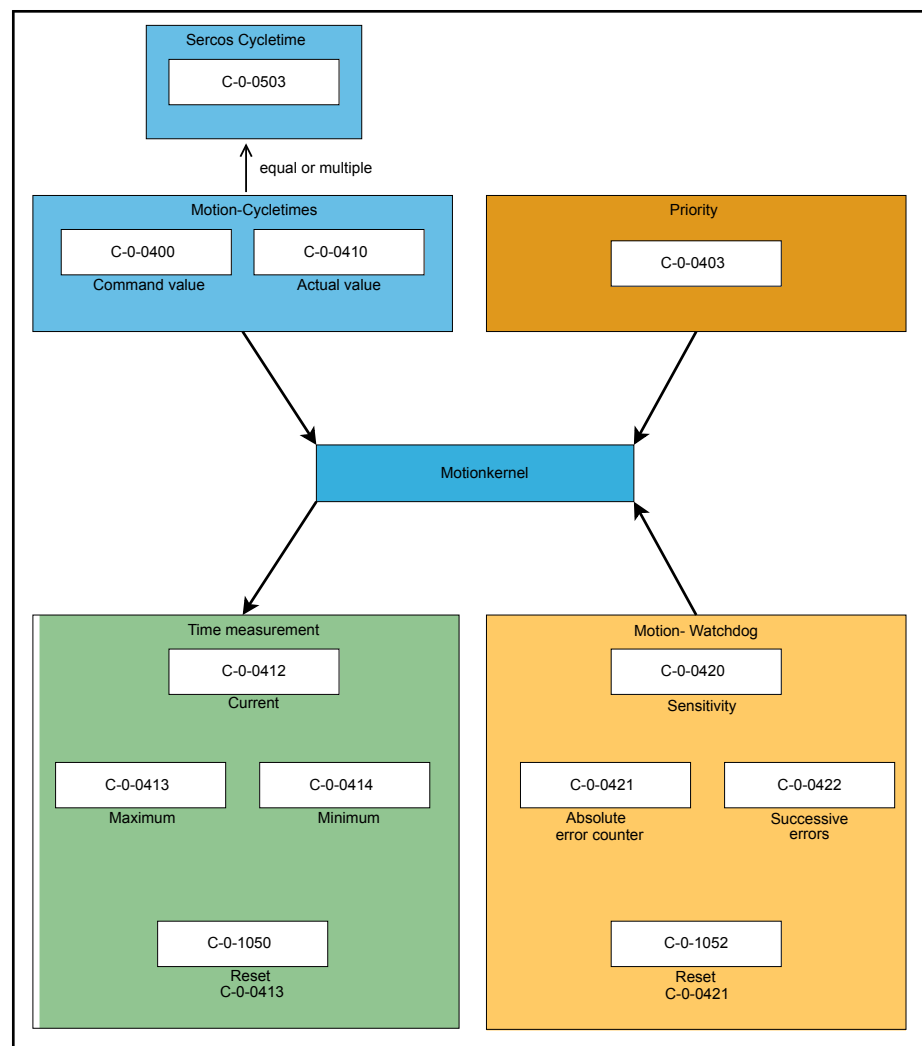


Fig. 5-26: Configuring the Motion kernel

5.4.9 CPU load of the control

The CPU load of the control results from the computing time of all tasks running in the system. These tasks can be system as well as PLC tasks.

Each task to be processed increases the CPU load. Approximately 80 system tasks are processed on the control in the current version. These tasks accept the different control jobs (PLC tasks, motion interpolation, monitoring, communication, field buses, parameter handling, error response, control display, HMI communication,...). Depending on the application, the number of tasks can even increase further.

CPU load factors

The most important factors are:

- Cycle times of the PLC tasks
- Scope of the PLC programs
- Cycle times (Motion and Sercos)
- Number of axes
- Number of kinematics
- C2C devices / number of master axes
- HMI communication

Diagnostic options

- **Control parameter C-0-0415**

The current CPU load can be read via the control parameter C-0-0415. The parameter displays the time (in percent) the system is not in "idle mode". The control is in idle mode when the task with the lowest priority is running. This task can only run if no other task is running due to its priority.

The control parameter C-0-0415 can also be seen via the dialog "Task Settings" in the context menu of the "Motion" node. The control has to be switched to online

- **Task Viewer**

The CPU load contains all tasks. In the Task Viewer ([chapter 5.4.6 "Task viewer" on page 188](#)), only the high priority Sercos copying routines, the motion computation and PLC tasks are displayed. Thus, the system is loaded by low-priority tasks shown in the Task Viewer as background.

For the "Task Viewer", go to the context menu of the control (**Diagnostics ► Task Viewer**). The control has to be switched to online.

Both diagnostic options display different values. The CPU load of the C-0-0415 contains all tasks. In contrast, only the high priority Sercos copying routines, the Motion computation and PLC tasks can be displayed in the Task Viewer. Thus, the system is loaded by low-priority tasks shown in the Task Viewer as background.

Basic CPU load

The basic load is the CPU load of the control after the initial switch-on with a default PLC project (control state: BB RUN).

All components are created at control startup. These components create the required tasks. Some of these tasks are cyclically called and it is checked whether relevant data exists. Thus, up to 15 tasks, which have to be processed, are called in each system cycle (each millisecond). Even if the runtime is very short for the individual tasks, it sums up and affects the CPU load. The default cycle times of the control are 2 ms. Sercos as well as Motion data have thus to be computed every 2 ms. Furthermore, the MotionTask of the PLC is thus also processed in this cycle.

CPU load

A permanent CPU load of the control of 100 % is to be avoided and an average load of 90-95 % is not to be exceeded. The Motion computation and the PLC-IEC tasks are not to require more than 75 % of the computation time. Depending on the requirements for the control, these values can vary.

The CPU load is not the mandatorily decisive factor in this case. The computation time required for the high priority tasks is the decisive factor.

The time available for the background tasks can be read using the Task Viewer. This time is available as computation time for Motion and PLC. Background tasks can be suppressed for a specified amount of time. Additionally, a time span that can be set for background tasks is reserved in each cycle. During this period, these tasks cannot be suppressed by high priority tasks. Depending on the tasks to be performed, the runtimes of all tasks are subject to fluctuations. Thus background tasks are dynamically suppressed and processed. If the available computation time is nearly completely loaded due to the Motion computation and the PLC-IEC tasks, the executed tasks can be slowed down due to the background tasks (communication, data transfer, parameter queries, display).

If there is still sufficient time, more axes can be added or the PLC program can be extended.

PLC programming

To adjust the system the best possible to the desired requirements, take the following factors into account:

- Motion and Sercos cycle times
- Cycle times and task priorities of the PLC-IEC tasks

5.5 I/O access

To use an I/O address, A PLC variable that accesses this address has to be defined. This process is known as "mapping".

The PLC allows two different methods to map a PLC variable to 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 to 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 mapping 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.

IndraWorks automatically specifies the addresses. The addresses can also be specified manually (see [fig. 3-31 "Labeling fixedly specified address" on page 60](#)).

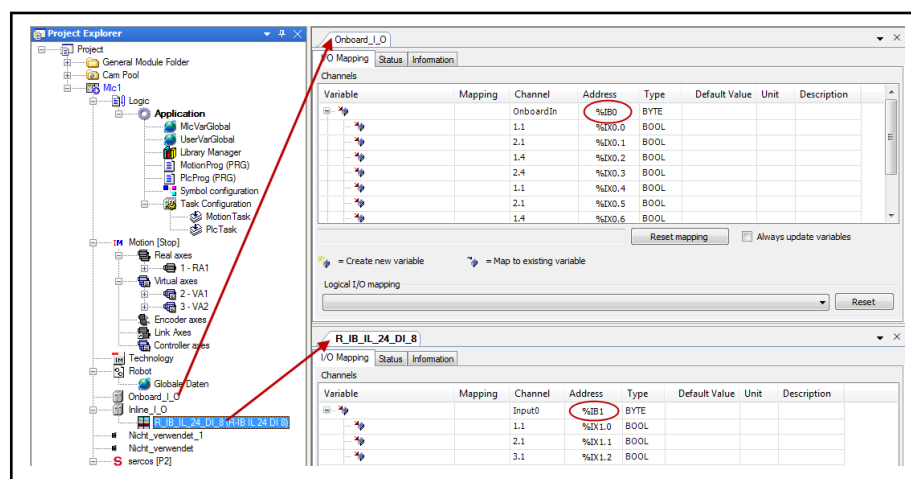


Fig. 5-27: Addresses of an I/O device

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 to 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
```

Declaring via I/O mapping

The second option to map a PLC variable to 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

Left-click on the symbol to switch between the two mapping methods.

Bus cycle task

The bus cycle task is the PLC task. At its beginning or end, the input or output data is copied from the I/O. This ensures data consistency and allows short simultaneous latency times.

The bus cycle task can either be set centrally at the control or separately for each I/O master. The setting at the I/O master has a priority higher than the setting at the control.



If no bus cycle task was determined, the fastest cyclic task was automatically used. This is the PLC task in the preset task configuration.



If other PLC tasks access these I/Os, the tasks can be stopped until the I/O mapping is completed.



Set the bus cycle time to the fastest PLC task accessing the corresponding I/O device.

Central determination of the bus cycle task at the control:

1. Double-click on the "Control" node in the Project Explorer.
The setting window of the control opens.

PLC programming

2. Open "PLC Settings".
3. Set the bus cycle task under "Bus Cycle Options".

Determining the bus cycle time for each I/O master:

1. Double-click on the corresponding "I/O Master" node in the Project Explorer.

A setting window of the corresponding I/O master opens.

2. Open "I/O Image".
3. Set the bus cycle task under "Bus Cycle Options".

For more information, refer to [chapter 5.4.7 "I/O turn-around times" on page 189](#).

5.6 Memory organization and pointer programming

5.6.1 Memory alignment

The memory alignment describes the alignment of data (variables, structures) in the memory.

The alignment methods are distinguished as follows:

- Byte, word and double word alignment:

The variables are aligned irrespective 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 for the IndraLogic XLC 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.

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 10007	Out04 : DWORD (3)	Fill byte
ADR 10008	–	Out04 : DWORD (0)
ADR 10009	–	Out04 : DWORD (1)
ADR 1000A	–	Out04 : DWORD (2)
ADR 1000B	–	Out04 : DWORD (3)

Tab. 5-4: Memory allocation depending on alignment

5.6.2 Byte order

The "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

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
ADR 10002	3C	00111100	2B	00101011
ADR 10003	4D	01001101	1A	00011010

Tab. 5-5: Memory allocation depending on the byte order



Numerical values are stored in "big endian" format in the control IndraLogic XLC.

5.6.3 Pointer addressing



Memory alignment and byte order have to be considered when addressing variables via pointers.

PLC programming

Pointer access to single variable or array

When accessing individual variables or arrays, one pointer should either point to the respective data type of the variable or the array elements. When using a deviating pointer type, e.g. byte pointer to a word or double word variable, observe the byte order.

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 was created according to the "Natural Alignment" method, addressing structural elements via pointers calculated via the basic address of the structure and a corresponding offset at runtime is not permitted.

The offset generation to address a structural element is not safely possible, as the basic address of the structure and thus the number of fill bytes can change if changes are made in the application.

With "Natural Alignment", the compatible use of the structures provides a direct addressing of the elements via the point operator:

StructureName.ElementName := ElementValue;

If the address of a structure is applied 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 *)
```

Invalid addressing

Program:

```
pt2 := ADR (StructInst.Out01); (* Pointer initialization *)
pt2 := pt + 3; (*Offset calc. for 3rd struct. element Out03*)
pt2^:= 2#11110000; (* Access to struct. 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 for IndraLogic XLC for selected data ranges by a corresponding compiler instruction ("pack_mode" pragma) with structural declaration.

**"pack_mode" compiler instruction
example**

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 run-time 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 IndraLogic XLC to slow down, as access to word and double word operands is always performed byte by byte.

6 Axes

6.1 Introduction and overview

Axis types

The IndraLogic XLC 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.

The following axis types are distinguished:

- **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 applied to the drive
 - **SercosDrive**
PackProfile-compatible drives. The axis motion is calculated in the control. Only the position command values are applied 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)
- **Encoder axis**
Axis only providing actual values (e.g. from an additional drive encoder)

The different axis types provide a variety of functionalities:

Functions of the axis types

	Real axis	Virtual axis	Encoder axis
Commanding	X	X	
Use as master axis	X	X	X
Use as slave axis	X	X	

Tab. 6-1: Overview on the functional scope of the axis types

Real axes

Real axes are divided into the following types:

- **IndraDrive**
Creating electric drives is described in section [chapter 6.2.3 "Creating a real axis" on page 213](#)
- **SercosDrive**
Creating a real axis is described in section [chapter 6.2.3 "Creating a real axis" on page 213](#)

Virtual axes

The virtual axis can also be used as master axis. The virtual axis patterns the behavior of a real axis in a mathematically ideal way.

Creating a virtual axis is described in section [chapter 6.3.2 "Creating a virtual axis \(virtual master axis\)" on page 222](#).

Encoder axes

The encoder axis can be used as real master axis.

Axes

- Real master axis data is calculated based on the measuring results of the encoder.
- Creating an encoder axis is described in section [chapter 6.4.2 "Creating an encoder axis \(real master axis\)"](#) on page 228.
- 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 III
IndraControl L25	16
IndraControl L45	32
IndraControl L65	64

Tab. 6-2: Maximum number of axes per control

- In addition to the above mentioned number of axes, drives can also be integrated as PLC device. There is no Motion functionality available for these drives. The drives are freely programmed in the PLC. Refer to [chapter 6.6 "Drive as PLC device"](#) on page 308.
- Field buses for real axes** Real axes can be controlled via the field buses Sercos III and Profibus. To connect via Sercos III, dialogs are available in the interface.

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.

6.2 Real axis

6.2.1 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 the hardware is to be operated.

The following features can be distinguished:

- **SercosDrive**

The functional scope of the Sercos interface specified in the Sercos PackProfile (see www.sercos.com under Technology → Sercos Packaging Profile) is used as Sercos 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 guarantees 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 dialogs are not displayed



The technology function blocks (winder, register controller, tension controller...) are not supported for a SercosDrive.

IndraDrive with interpolation in the control or drive is required.

- **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 operating mode 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. Thus, 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-D5-1-SNC-NN
- FWA-INDRV*-MPD-06VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-06VRS-D5-1-SNC-NN ¹⁾
- FWA-INDRV*-MPC-06VRS-D5-1-SNC-NN
- FWA-INDRV*-MPH-07VRS-D5-1-SNC-NN
- FWA-INDRV*-MPD-07VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-07VRS-D5-1-SNC-NN ¹⁾
- FWA-INDRV*-MPH-08VRS-D5-1-SNC-NN
- FWA-INDRV*-MPD-08VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-08VRS-D5-1-SNC-NN ¹⁾
- FWA-INDRV*-MPB-16VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-17VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-18VRS-D5-1-SNC-NN
- FWA-INDRV*-MPB-19VRS-D5-1-SNC-NN
- FWA-INDRV*-MPM-17VRS-D5-1-SNC-NN
- FWA-INDRV*-MPM-18VRS-D5-1-SNC-NN
- FWA-INDRV*-MPC-17VRS-D5-1-SNC-NN
- FWA-INDRV*-MPC-18VRS-D5-1-SNC-NN
- FWA-INDRV*-MPC-19VRS-D5-1-SNC-NN

¹⁾ the minimum Sercos cycle time allowed is **2 ms** for the drives of the MPB-06VRS-, MPB-07VRS-, MPB-08VRS family. If a lower value is set, the error message "C0171 Maximum length for connection exceeded" is displayed.

It is also distinguished whether 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 [tab. 6-4 "Function blocks in IndraLogic XLC and IndraMotion MLC" on page 210](#). Thus, 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

Axes

command, 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 following operating states.

- "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-D5-1-NNN-NN
- FWA-INDRV*-MPE-17VRS-D5-1-NNN-NN
- FWA-INDRV*-MPE-18VRS-D5-1-NNN-NN

It is also distinguished whether 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 [tab. 6-4 "Function blocks in IndraLogic XLC and IndraMotion MLC" on page 210](#). Thus, only a limited number of axis parameters is synchronized with the drive parameters.



Whether the real axis is to be interpolated in the drive or the control has to be specified when creating the axis.

Motion functionality

Functionality	IndraDrive with interpolation in the drive	IndraDrive with interpolation in the control	SercosDrive
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

Tab. 6-3: Overview on the Motion functionality

Function block	Real axes					Virtual axes	Encode/link* axis	Controller axis*
	IndraDrive		HydraulicDrive*		sercos-Drive			
	Interpolation Drive	Interpolation Control	Interpolation Drive	Interpolation Control				
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

Axes

Function block	Real axes					Virtual axes	Encode/ link* axis	Controller axis*
	IndraDrive		HydraulicDrive*		sercos- Drive			
	Interpolation Drive	Interpolation Control	Interpolation Drive	Interpolation Control				
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
MB_SetPositionControlMode (5)	X	X	–	–	–	–	–	–
MC_AddAxisToGroup * (4)	X	X	X	X	X	X	–	X
MC_CamIn	X	–	–	–	–	–	–	–
MC_CamOut	X	–	–	–	–	–	–	–
MC_GearIn	X(2)	X(2)	–	X	X	X	–	X
MC_GearOut	X(2)	X(2)	–	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(2)	X	X	X	X	–	X
MC_Power	X(2)	X(2)	X	X	X	–	–	X
MC_RemoveAxisFromGroup * (4)	X	X	X	X	X	X	–	X
MC_Reset	X(2)	X(2)	X	X	X	X	–	X
MC_Stop	X(2)	X(2)	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	–	–	–	–

Axes

Function block	Real axes					Virtual axes	Encode/ link* axis	Controller axis*
	IndraDrive		HydraulicDrive*		sercos- Drive			
	Interpo- lation Drive	Interpo- lation Control	Interpo- lation Drive	Interpo- lation Control				
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

- * Only IndraMotion MLC
 (1) Not MPx03
 (2) Also with OpenLoop
 (3) Depends on the implementation of the controller
 (4) For 13VRS and higher
 (5) For 13VRS and > MPx06
 Tab. 6-4: Function blocks in IndraLogic XLC and IndraMotion MLC

**Open loop**

Only function blocks without position control are possible for real axes without position feedback (open-loop).

Drive operating states used

Operating state	IndraDrive with interpolation in the drive	IndraDrive with interpolation in the control	SercosDrive
Torque/force control	Yes	Yes	No
Velocity control	Yes	Yes	No
Position control with cyclic command value specification	Yes	Yes	Yes
Drive-controlled positioning	Yes	No	No
Velocity synchronization with virtual master axis	Yes	No	No
Phase synchronization with virtual master axis (position synchronization with virtual master axis)	Yes	No	No
Cam table with virtual master axis (position synchronization with virtual master axis)	Yes	No	No
MotionProfile with virtual master axis (position synchronization with virtual master axis)	Yes	No	No
Direct valve control	No	No	No

Tab. 6-5: Overview on the drive operating states used

6.2.2 Error response of 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 13VRS Parameters" and [chapter 6.5.14 "Error response" on page 254](#)).

6.2.3 Creating a real axis



Axes can only be created in offline mode!

All real axes (IndraDrive and SercosDrive) can thus be created. A real axis is created according to the drag&drop principle. Drag the required device "IndraDrive" from the library and

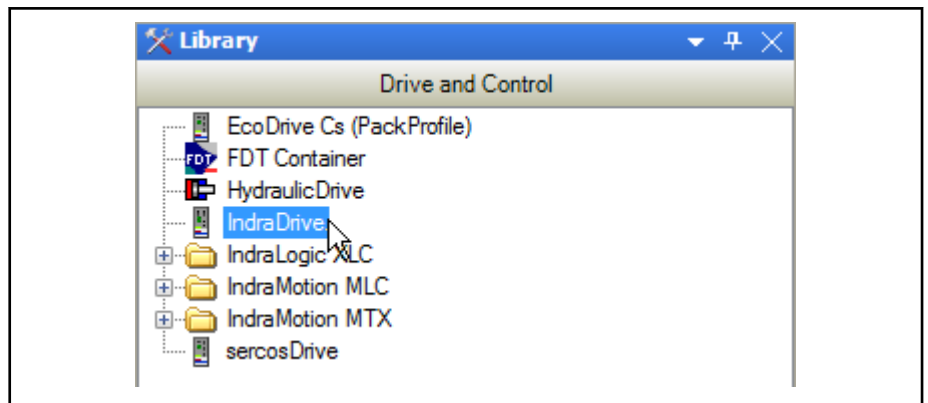


Fig. 6-1: Creating a real axis - Dragging from the library
drop it to the desired bus, in this case Sercos.

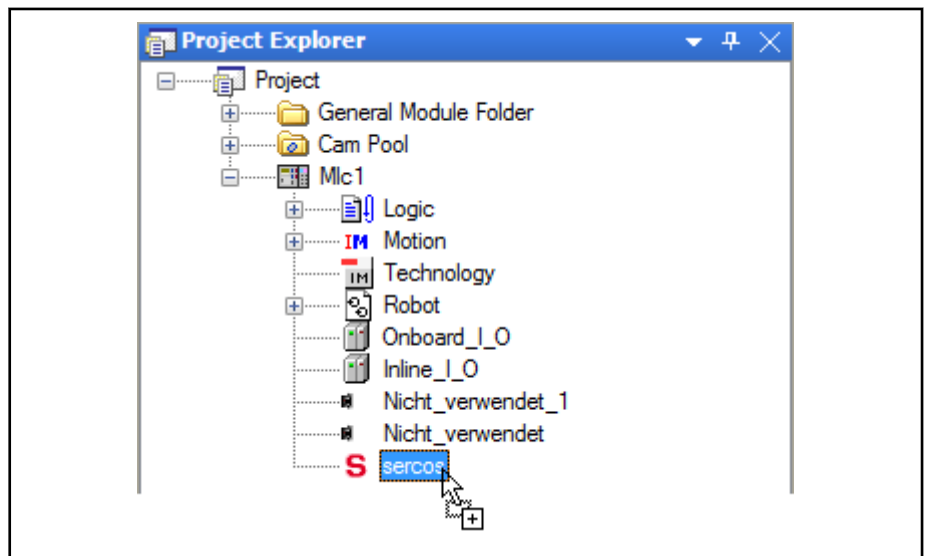


Fig. 6-2: Creating a real axis - Dropping to the Project Explorer

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.

Axes

Fig. 6-3: Creating a real axis (IndraDrive) - General Settings

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

Apply or change the preset values.

Fig. 6-4: Creating a real axis (IndraDrive) - Configuration

Configuring the real axis (axes)

The "Firmware" setting allows to select from two different software states, here MPH-07. Permitted firmwares are listed in [chapter 6.2.1 "General information" on page 208](#).

Under "Axis", the address of the drive **has to** be set in the Sercos bus (address in the drive configuration, in this case 1). Otherwise, communication with the drive is not possible.

With the "Interpolation in drive" option, the drive automatically calculates its position command values for the operating state selected, e.g. synchronization or cam. Otherwise, the control provides the drive with position command values.

If the checkbox is deselected, the control provides position command values to the drive. This is a default setting for SercosDrives and EcoDrives.

The "Closed Loop" option is used for an axis with position feedback. If the checkbox is deselected, an axis without position feedback ("open loop") 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).

Drive configuration

The drive has to be configured before starting the offline parameterization for a drive.

If "Implement configuration for offline parameterization" is selected, a dialog to enter the required data is displayed. The configuration for offline parameterization is generated from the entries.

Fig. 6-5: 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

Refer to [chapter 6.2.1 "General information" on page 208](#).

Axis assignment

To address the axis, the PLC uses the axis number and axis name (here "1" or "RA1").

Axes

Overview in the Project Explorer

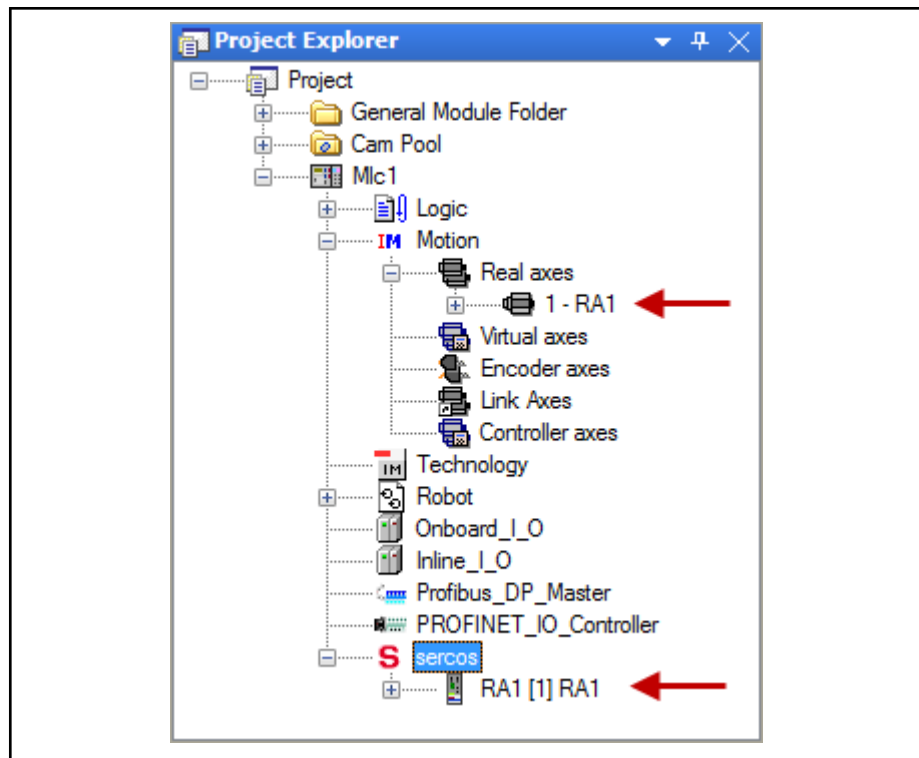


Fig. 6-6: Creating a real axis - Overview in the Project Explorer

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 [chapter 6.2.4 "Real axis – Dialogs" on page 216](#).

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 the online mode or in the offline parameterization.

S- and P-parameters are only stored in the drive and are thus directly displayed.

6.2.4 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, it is assumed that the presets cover a wide range of applications and that a modification is only required in exceptional cases.

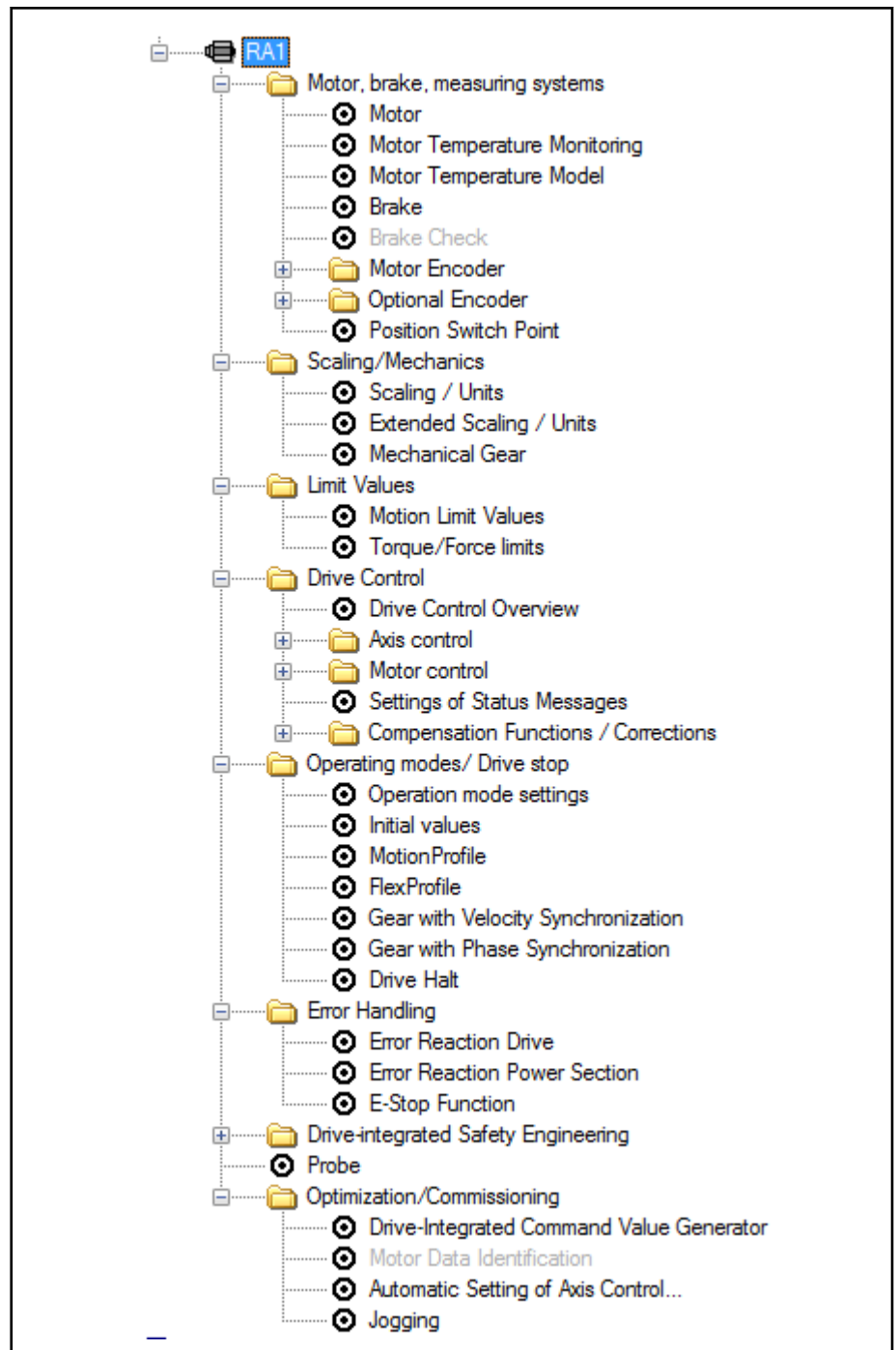


Fig. 6-7: Dialogs of a real axis (IndraDrive)



If IndraDrive drive units are disabled at "Interpolation in the Drive", EcoDrive CS and SercosDrive, the drive is controlled via PackProfile (refer to the following figure).

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

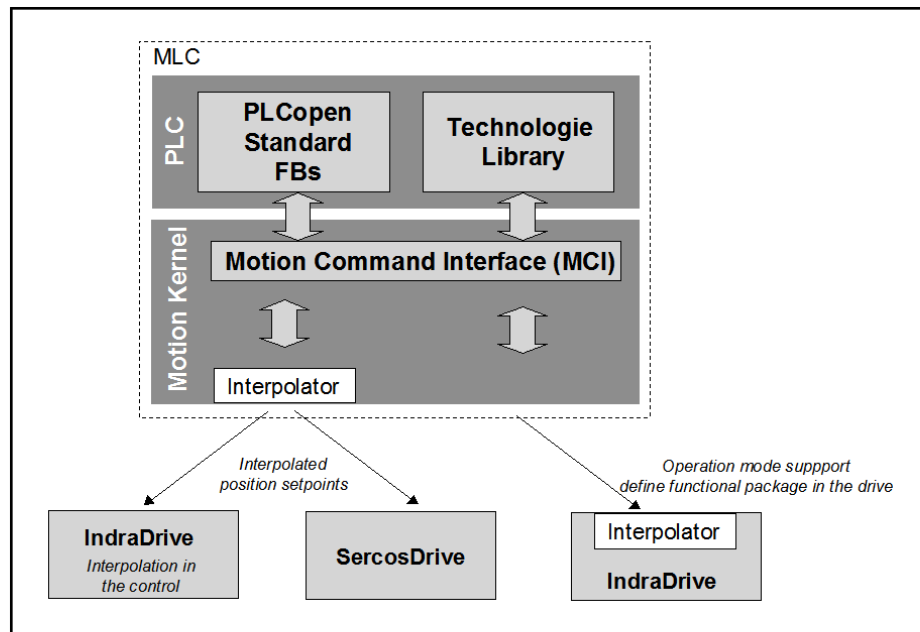


Fig. 6-8: Control principle for different drives



For a real axis without position feedback (open loop), only the operation modes without position feedback are shown.

- [chapter 6.5.3 "Motor" on page 240](#)
- [chapter 6.5.4 "Motor temperature monitoring" on page 241](#)
- [chapter 6.5.5 "Brake" on page 242](#)
- [chapter 6.5.6 "Measuring systems" on page 242](#)
- [chapter 6.5.7 "Settings: scaling/units" on page 245](#)
- [chapter 6.5.8 "Scaling/units extended" on page 246](#)
- [chapter 6.5.9 "Mechanical gear" on page 248](#)
- [chapter 6.5.10 "Closed-loop control" on page 249](#)
- [chapter 6.5.11 "Settings of status messages" on page 250](#)
- [chapter 6.5.12 "Drive halt" on page 251](#)
- [chapter 6.5.13 "Establishing dimensional reference" on page 252](#)
- [chapter 6.5.14 "Error response" on page 254](#)
- [chapter 6.5.15 "E-STOP function" on page 256](#)
- [chapter 6.5.16 "Motion limit values" on page 257](#)
- [chapter 6.5.17 "Initial values" on page 259](#)
- [chapter 6.5.18 "Compensation functions/corrections" on page 261](#)
- [chapter 6.5.19 "Drive-integrated command value generator" on page 263](#)
- [chapter 6.5.20 "Cam table" on page 263](#)
- [chapter 6.5.21 "MotionProfile" on page 269](#)
- [chapter 6.5.22 "FlexProfile" on page 275](#)
- [chapter 6.5.23 "Gear with velocity synchronization" on page 282](#)
- [chapter 6.5.24 "Gear with phase synchronization" on page 286](#)
- [chapter 6.5.26 "Drive-integrated safety engineering" on page 292](#)

- [chapter 6.5.27 "Position switching point" on page 292](#)
- [chapter 6.5.28 "Touch probe" on page 293](#)

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 original product manuals.

6.2.5 Real axis – Context menu

Overview on menu items

The context menu of a real axis contains the following menu items:

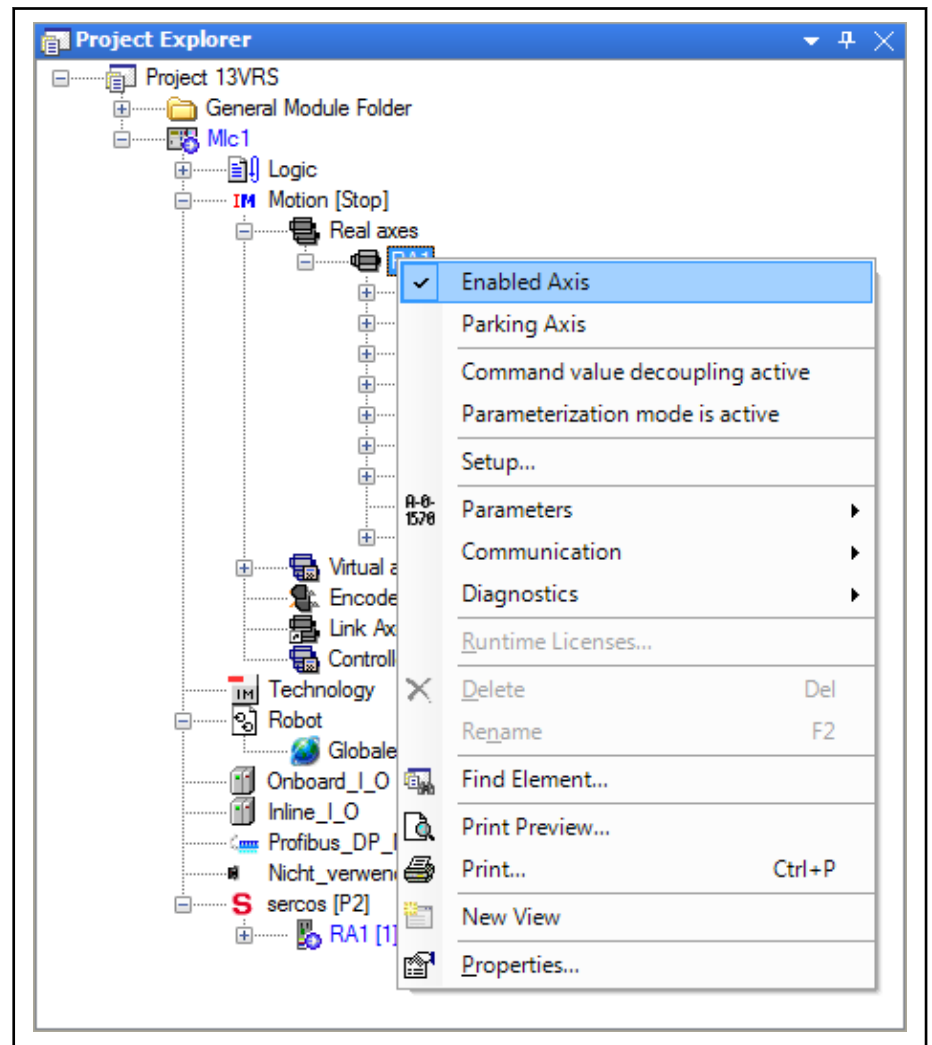


Fig. 6-9: Context menu of a real axis (IndraDrive) in online mode

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 operating states **Activated Axis/ Deactivated Axis** and **Parking Axis** of the drive using the options of the IndraDrive drive (see also [chapter 7.7 "Axis modes" on page 340](#)).
- **Command value decoupling enabled** - see [chapter 7.7 "Axis modes" on page 340](#)

Axes

- **Parameterization Mode Active** - Switch on/off the parameterization mode of the axis. For more information on the parameterization mode, refer to [chapter 7.5 "Parameterization level" on page 325](#).
- [chapter 6.5.29 "Initial Commissioning" on page 296](#) - Commissioning cycle of the real axis (IndraDrive)
- [chapter 6.5.30 "Parameters" on page 297](#)
 - [chapter "Parameter editor" on page 109](#)
 - [chapter "Parameter group" on page 111](#)
- [chapter 6.5.31 "Communication" on page 299](#)
- [chapter 6.5.32 "Diagnostics" on page 304](#)
- **Delete** - Deletes the complete axis from the project in **Offline mode**.
- **Rename** - Changes the axis name in **offline mode**.



The axis reference to the PLC also changes with the axis name. This can cause the application being compiled with errors.

- **Find Element...** - Allows scanning for elements in the project
- **Print Preview...** - Selects whether to print the configured properties, the parameters or both of the real axis There is a print preview before printing
- **Print...** - Selects whether to print the configured properties, the parameters or both of the real axis
- **New View** - Displays the axis in an individual window
- **Properties...** - Refer to [chapter "Axis properties" on page 220](#)

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.

Axis properties

The "Properties" dialog has two tabs.

"General" tab

The following settings can be changed for all axis types on the tab:

- **Name:** This name has to be unique and IEC-compliant. Otherwise, a warning is displayed



The axis reference to the PLC also changes with the axis name. This can cause the application being compiled with errors.

- **Comment:** This text is displayed as tooltip on the axis if neither a warning nor an error is present for the axis.
- **Author:** The latest editor of an axis property can be entered in this field.

"Configuration" tab

The following settings - depending on the axis type - can be changed on the tab:

- For the virtual axis:
 - **Axis number:** The axis number has to be unique. Otherwise, a warning is displayed
- For the real axis:

Axes

- Axis number(s): The axis number has to be unique. Otherwise, a warning is displayed. The axis numbers in a drive have to be unique for drives with multiple axes. Otherwise, an error is displayed.
- Firmware: The selected firmware can be changed in the selected drive family (e.g. IndraDrive) as long as the number of axes does not change.



If the derivative of the drive changes when changing the firmware (e.g. from MPB to MPH), the offline parameters of the drive might be lost.

- Sercos address(es): The Sercos address has to be unique. Otherwise, a warning is displayed. The Sercos addresses in a drive have to be unique for drives with multiple axes. Otherwise, an error is displayed.
- Axis name: The axis name can only be changed for multiple axis drives. This name has to be unique and IEC-compliant. Otherwise, a warning is displayed.
- Interpolation method: Some drives only support the interpolation in the control. In this case, this setting cannot be changed.



When changing the interpolation type of the axis, the interface to the respective drive is changed. Thus, we recommend to apply basic parameters to the drive after a change (see [chapter "Loading basic parameters" on page 297](#)).

It is recommended not to import a parameter file with S-/P-parameters of the drive with the associated different interpolation type (e.g. parameter file with "interpolation in the drive" to the drive with "interpolation in the control").

- Position feedback:
 - Checkbox enabled "Closed Loop" with position feedback (encoder)
 - or
 - Checkbox not enabled "Open Loop" without position feedback (without encoder)



Without position feedback (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).

- Configuration of the drive for the offline parameterization: Configure the offline parameterization for the drive or adjust an existing configuration.
- For the encoder axis:
 - Axis number: The axis number has to be unique. Otherwise, a warning is displayed.
 - Drive assignment

After the changes, apply the settings with **OK**.

Axes



The properties of a control can only be edited if there is no connection to the control (the control is neither online nor logged in).



After a property has been changed (e.g. axis number) and the subsequent online going of the control, messages might be displayed in the diagnostic memory of the control showing that the corresponding axis was deleted and created again. These messages show the internal procedure of a change (axis is deleted and created again with new properties) and can be ignored.

6.3 Virtual axis

6.3.1 General Information

The virtual axis is among the core functions of the IndraLogic XLC. The virtual axis 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. Virtual axes 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.

6.3.2 Creating a virtual axis (virtual master axis)

Creating the virtual axis

For 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-click or <Shift>+<F10>).

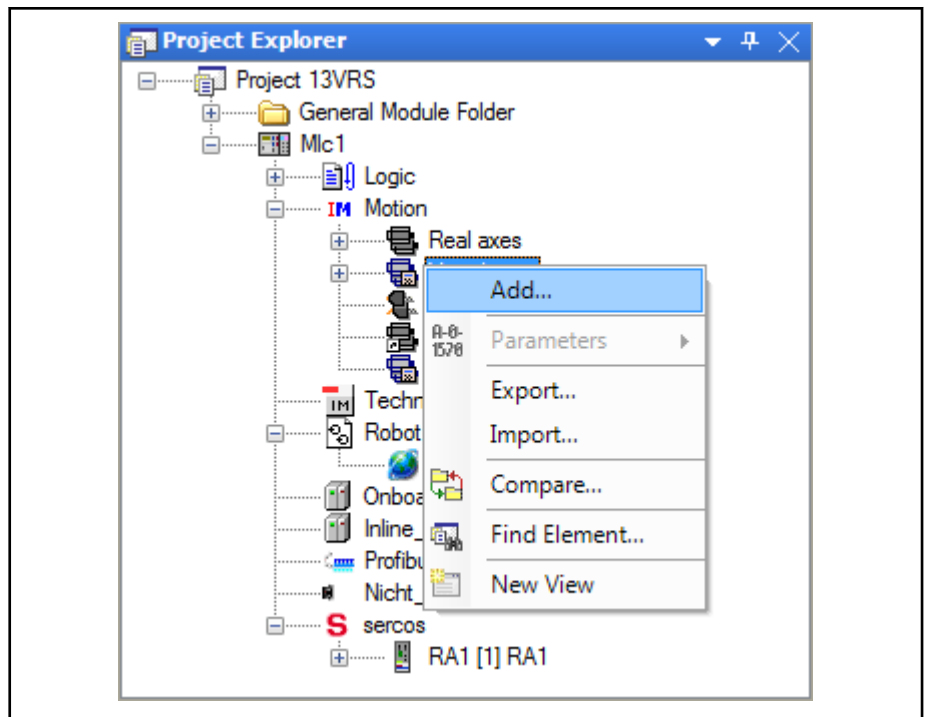


Fig. 6-10: Creating a virtual axis – Context menu

Dialogs to create a virtual axis

The creation of a virtual axis is supported by a wizard with two dialog boxes.

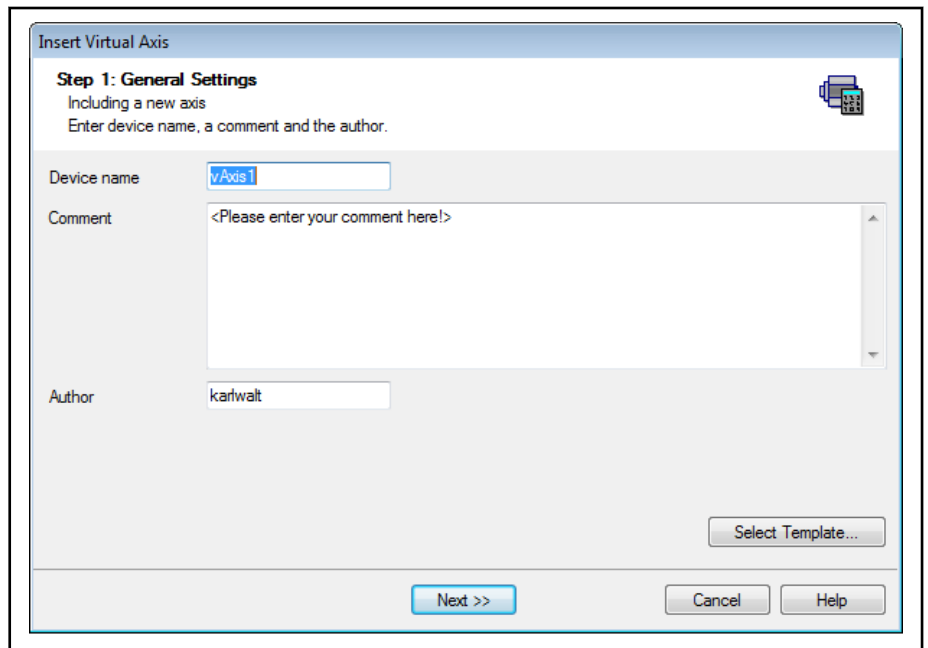


Fig. 6-11: Creating a virtual axis – General Settings

In the "General Settings" step, device name, comment on this device and author are entered. Apply or change the preset values.

Axes

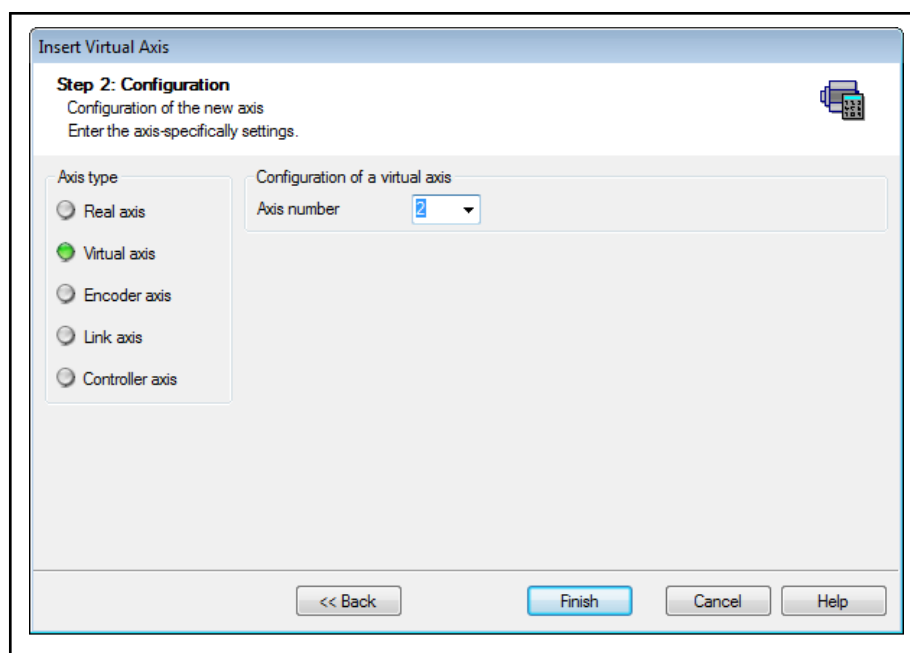


Fig. 6-12: Creating a virtual axis – Configuration

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 "2" or "vAxis1").

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 -> vAxis1** via right click, the following menu items are displayed:

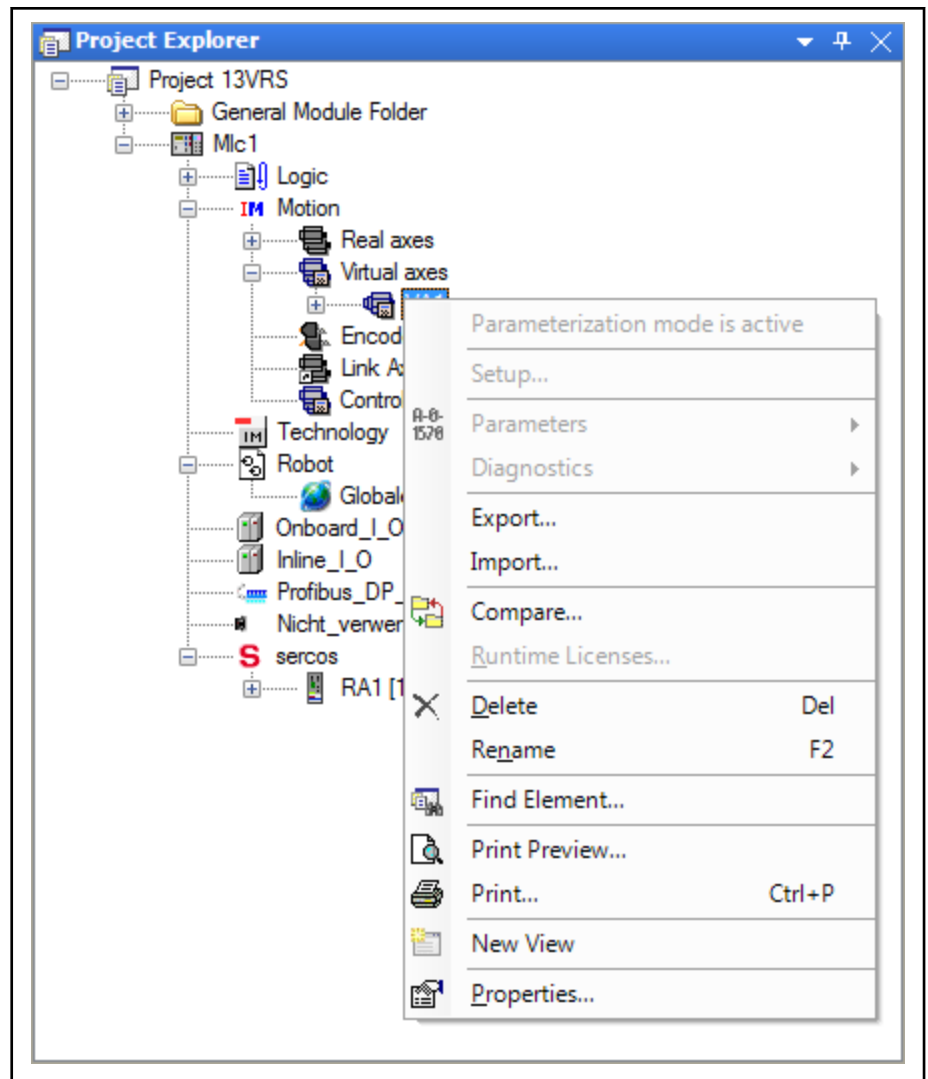


Fig. 6-13: Virtual axis - Context menu

As no connection is yet established (offline), some commands are disabled (for the complete description, see [chapter 6.3.4 "Virtual axis - Context menu" on page 226](#) or [chapter 6.3.3 "Virtual axis - Dialogs" on page 225](#)).

6.3.3 Virtual axis - Dialogs

Overview on dialogs

When opening the folder of the axis, the individual dialogs (displaying axis data and setting options for parameters) are displayed.

First, it is assumed that the presettings cover a wide range of applications and that a modification is only required in exceptional cases.

Axes

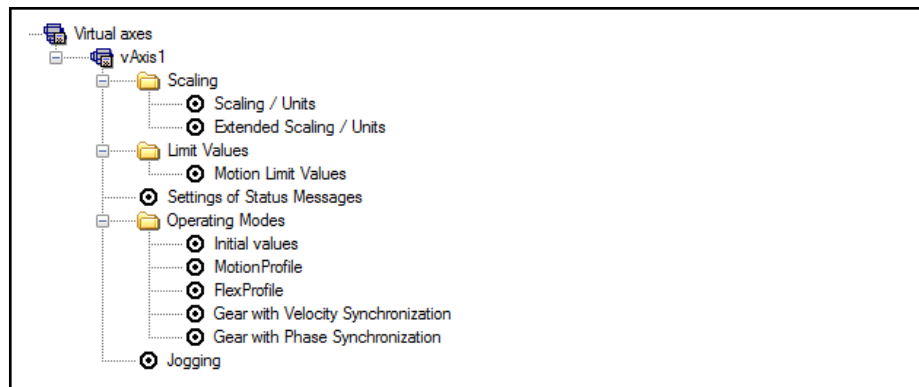


Fig. 6-14: Dialogs of a virtual axis

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:

- Scaling
 - [chapter 6.5.7 "Settings: scaling/units" on page 245](#)
 - [chapter 6.5.8 "Scaling/units extended" on page 246](#)
- Limit values
 - [chapter 6.5.16 "Motion limit values" on page 257](#)
 - [chapter 6.5.11 "Settings of status messages" on page 250](#)
- Operation modes
 - [chapter 6.5.17 "Initial values" on page 259](#)
 - [chapter 6.5.21 "MotionProfile" on page 269](#)
 - [chapter 6.5.22 "FlexProfile" on page 275](#)
 - [chapter 6.5.23 "Gear with velocity synchronization" on page 282](#)
 - [chapter 6.5.24 "Gear with phase synchronization" on page 286](#)

6.3.4 Virtual axis - Context menu

Overview on menu items

The context menu of a virtual axis in online mode contains the following menu items:

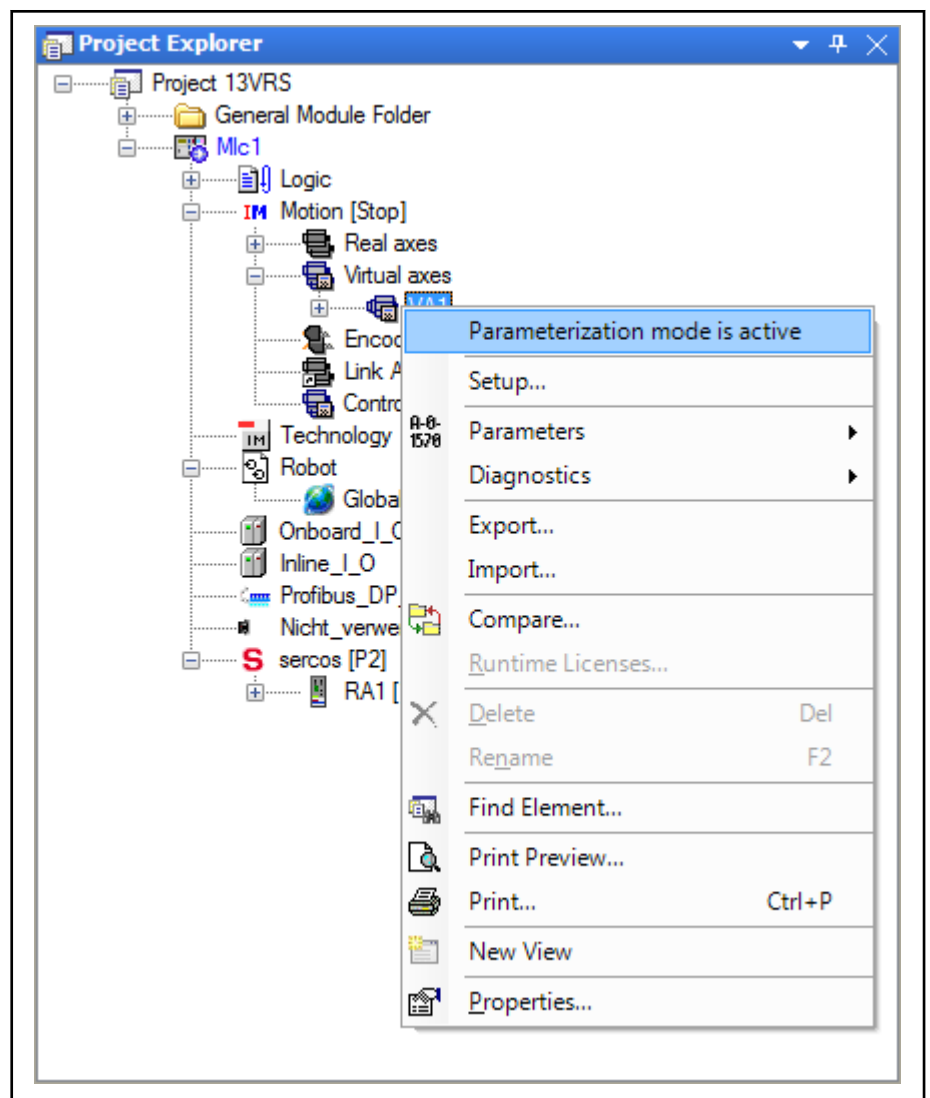


Fig. 6-15: Context menu of a virtual axis in online mode

- **Parameterization Mode Active** - Switch on/off the parameterization mode of the axis. For more information on the parameterization mode, refer to [chapter 7.7.5 "Axis in parameterization mode" on page 344](#).
- [chapter 6.5.29 "Initial Commissioning" on page 296](#) - Setup cycle
- [chapter 6.5.30 "Parameters" on page 297](#)
 - [chapter "Parameter editor" on page 109](#)
 - [chapter "Parameter group" on page 111](#)
- [chapter 4.4.7 "Diagnostics" on page 117](#)
 - [chapter "Status" on page 304](#)
 - [chapter "IDN list of invalid operating data" on page 305](#)
- **Export...** - Saves axis elements in a file. This menu item is described in [chapter 4.4.13 "Exporting" on page 148](#)
- **Import...** - Imports axis elements from a file. This menu item is described in [chapter 4.4.14 "Importing" on page 149](#)
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging" on page 151](#)

Axes

- **Runtime Licenses...** - Refer to [chapter 4.4.16 "Runtime Licenses" on page 153](#)
- **Delete** - Deletes the complete axis from the project in **Offline mode**.
- **Rename** - Changes the axis name in **offline mode**.



The axis reference to the PLC also changes with the axis name. This can cause the application being compiled with errors.

- **Find Element...** - Allows scanning for elements in the project
- **Print Preview...** - Selects whether to print the configured properties, the parameters or both of the virtual axis. There is a print preview before printing.
- **Print...** - Selects whether to print the configured properties, the parameters or both of the virtual axis.
- **New View** - Displays the axis in an individual window
- **Properties...** - Refer to [chapter "Axis properties" on page 220](#)

6.4 Encoder axis

6.4.1 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.

6.4.2 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. RA1, to supply the data required, i.e. before defining an encoder axis, the real axis RA1 has to be created first (see also [chapter 6.2.3 "Creating a real axis" on page 213](#)). Subsequently, the procedure described below is to be followed.

In the example, the axis RA1 has the drive address 1 and the axis number 1.



Without previous measuring encoder configuration (see [chapter "Configuring the measuring encoder" on page 231](#)) on the corresponding 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. Thus, enable the context menu (right-click or press <Shift>+<F10>).

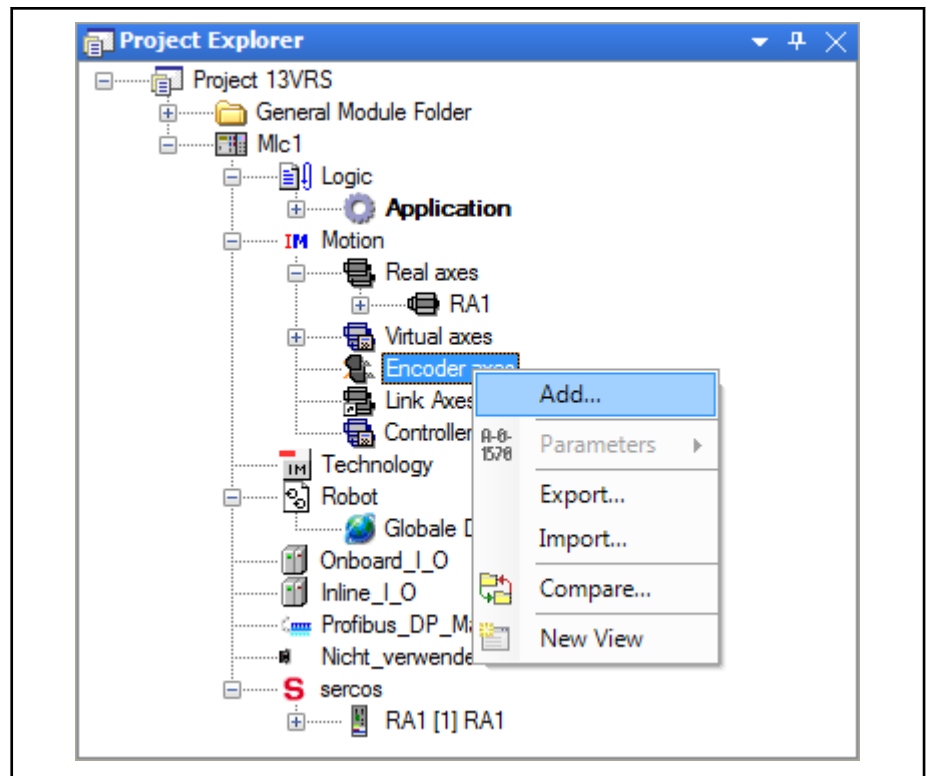


Fig. 6-16: Creating an encoder axis - Context menu

Dialogs to create an encoder axis

The creation of an encoder axis is supported by a wizard with two dialog boxes.

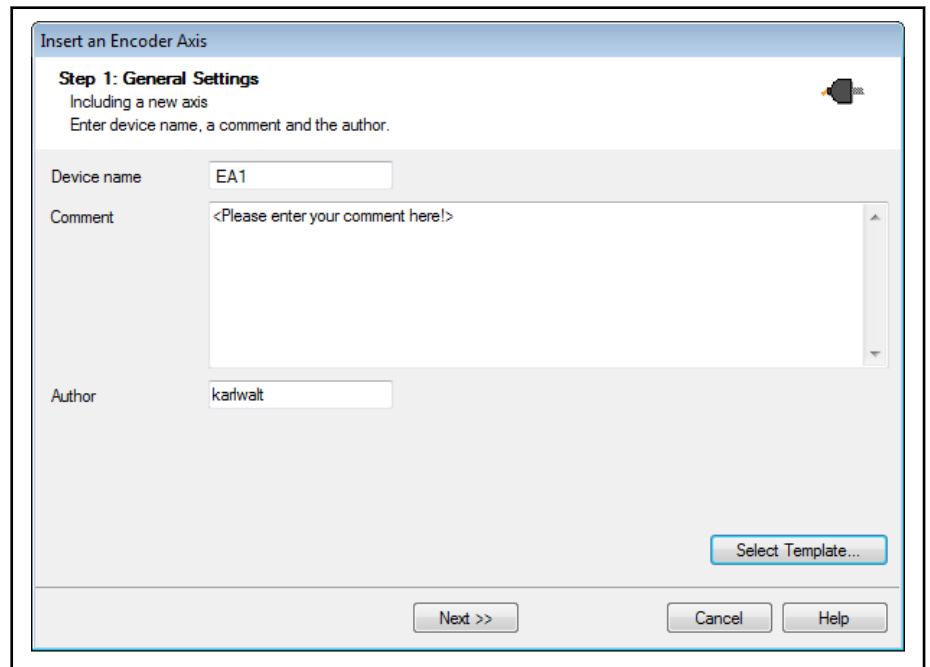


Fig. 6-17: Creating an encoder axis – General Settings

In the "General Settings" step, device name, comment on this device and author are entered. Apply or change the preset values.

Axes

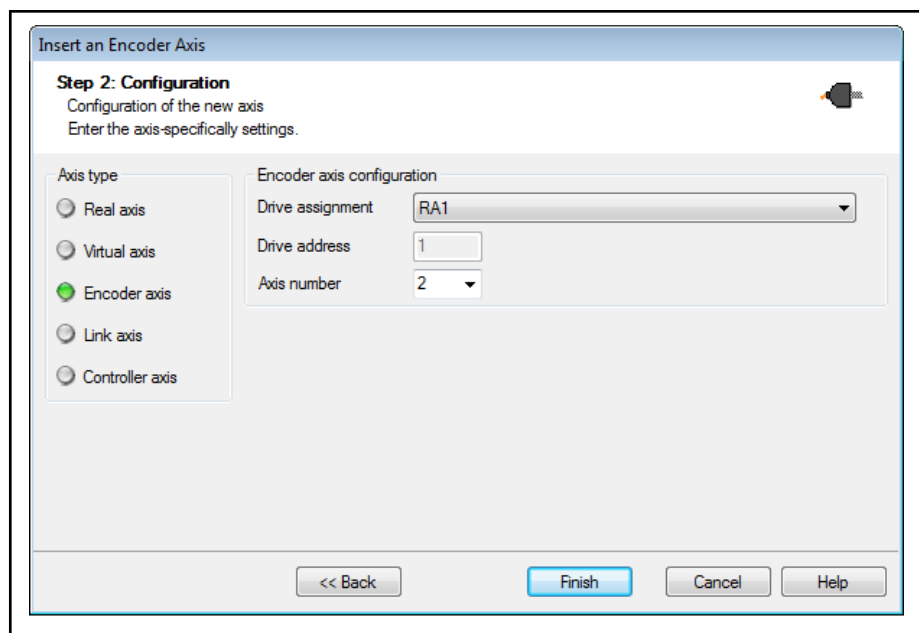


Fig. 6-18: Creating an encoder axis – Configuration

In the "configuration" step, select the drive (scroll window drive assignment: In the example RA1).

RA1 and thus its drive address 1 is selected.



Deactivating the real axis RA1, its drive, its electronics, results in the deactivation of the real encoder axis EA1, whose encoder is connected to this electronics (see also [chapter 7.7 "Axis modes" on page 340](#)).

- Axis assignment** To address the axis, the PLC uses the axis number and axis name (here "2" or "EA1").
- The encoder axis is shown only once in the Project Explorer:
- Context menu** Right-click to open the context menu on the folder **Motion** ► **Encoder Axes** ► **EA1** to display the following menu items:

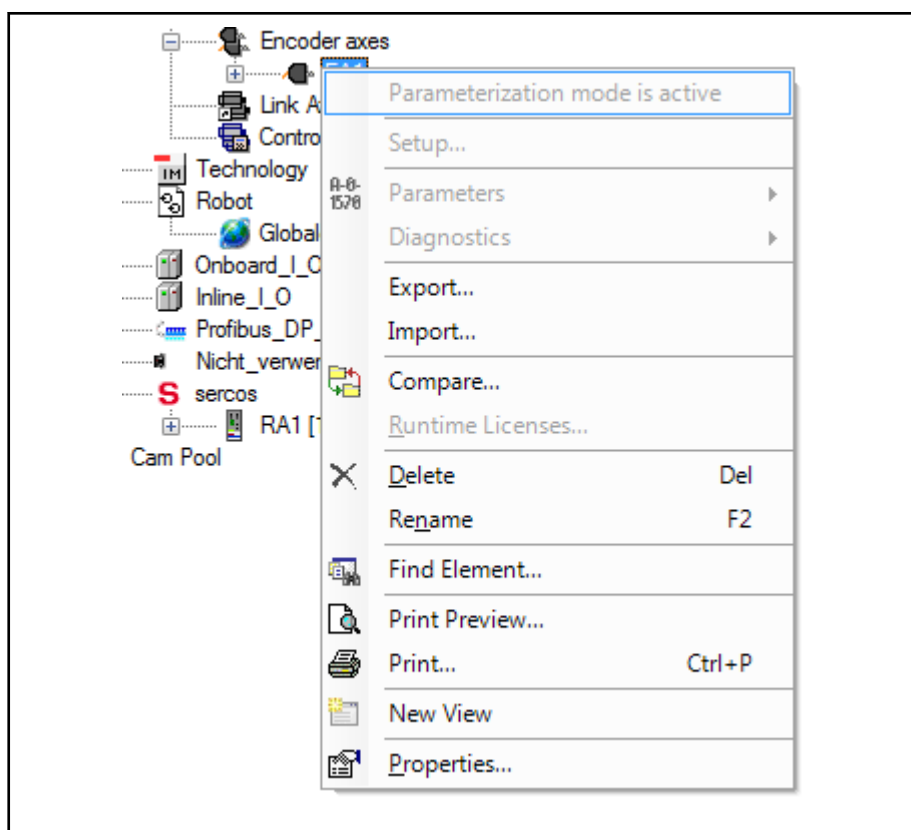


Fig. 6-19: Encoder axis - Context menu

As no connection is established (offline), many commands are disabled (for the complete description, see [chapter 6.4.4 "Encoder axis - Context menu" on page 232](#) or [chapter 6.4.3 "Encoder axis - Dialogs" on page 232](#)).

Configuring the measuring encoder

Configure the measuring encoder below the Sercos node in the real axis, whose control electronics is used by the encoder. Select the encoder and right-click to open the "Configuration" dialog.

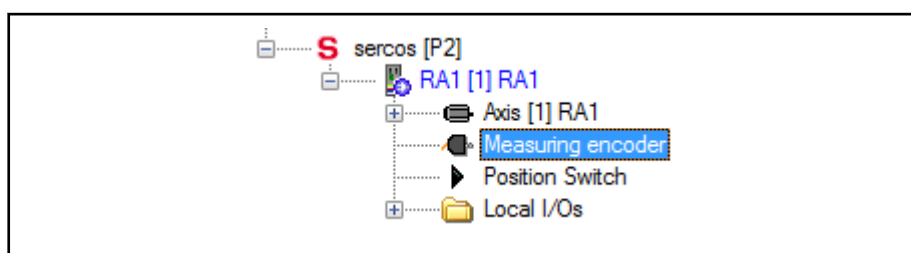


Fig. 6-20: Measuring encoder connected to the real axis RA1

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.

Axes

The measurement encoder dialog is explained in [chapter 6.5.6 "Measuring systems" on page 242](#).



For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH, MPB, MPD, MPC-07", chapter "Measuring encoder" → "Expansion package "synchronization"" (refer to [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.4.3 Encoder axis - Dialogs

Overview on dialogs

When opening the encoder axis folder, the dialogs available for this axis are displayed.

First, it is assumed that the presets cover a wide range of applications and that a modification is only required in exceptional cases.

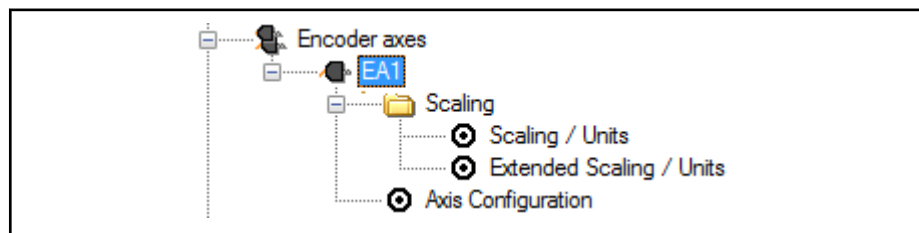


Fig. 6-21: Dialogs of an encoder axis

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:

- [chapter 6.5.7 "Settings: scaling/units" on page 245](#)
- [chapter 6.5.8 "Scaling/units extended" on page 246](#)
- [chapter 6.5.25 "Axis configuration" on page 290](#)

6.4.4 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:

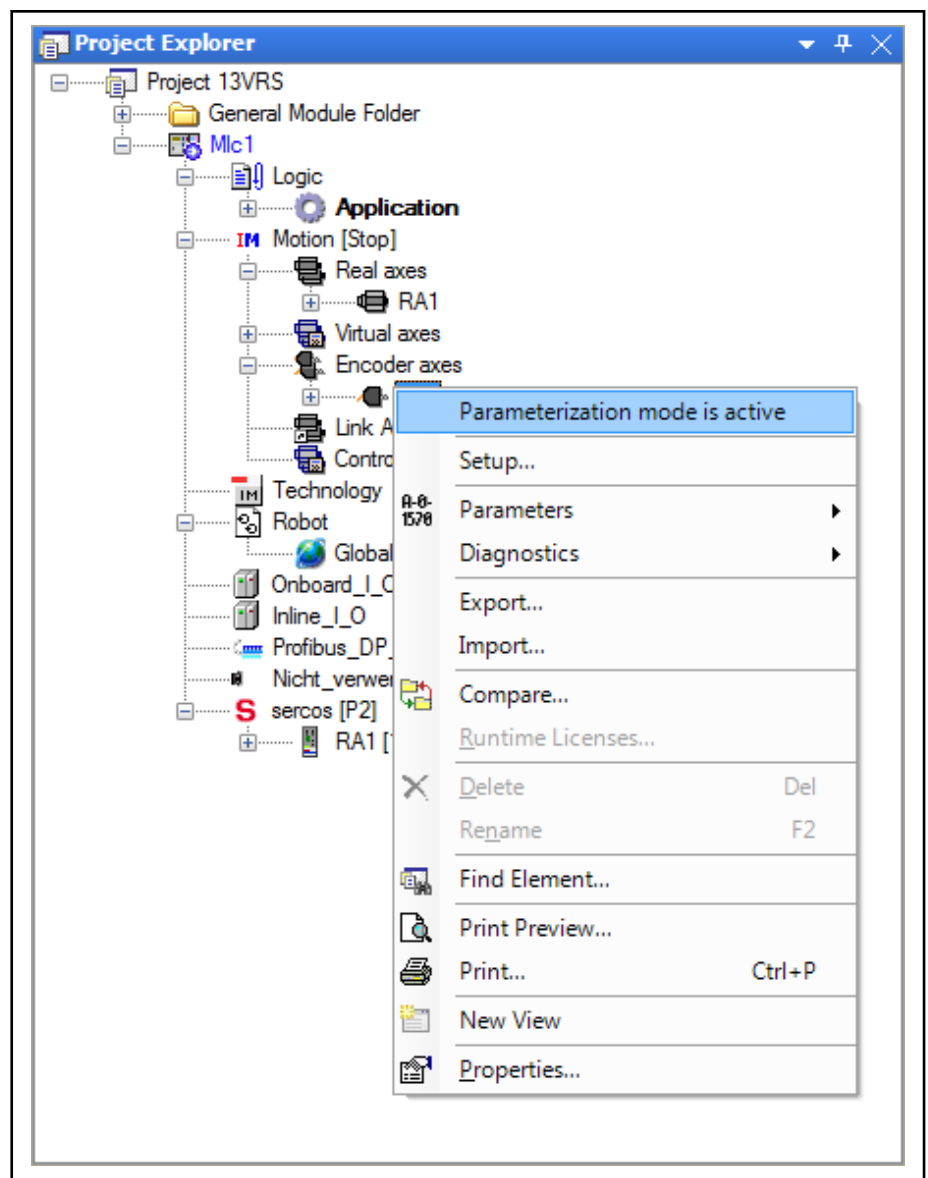


Fig. 6-22: Context menu of an encoder axis in online mode

- **Parameterization Mode Active** - Switch on/off the parameterization mode of the axis. For more information on the parameterization mode, refer to [chapter 7.7.5 "Axis in parameterization mode" on page 344](#).
- [chapter 6.5.29 "Initial Commissioning" on page 296](#) - Commissioning cycle of the encoder axis
- [chapter 6.5.30 "Parameters" on page 297](#)
 - [chapter "Parameter editor" on page 109](#)
 - [chapter "Parameter group" on page 111](#)
 - [chapter "Finding parameters" on page 113](#)
 - [chapter "Loading basic parameters" on page 297](#)
 - [chapter "Parameters - Export" on page 113](#)
 - [chapter "Parameters - Import" on page 115](#)
- Diagnostics
 - [chapter "Status" on page 304](#)

Axes

– [chapter "IDN list of invalid operating data" on page 305](#)

- **Export...** - Saves axis elements in a file. This menu item is described in [chapter 4.4.13 "Exporting" on page 148](#)
- **Import...** - Imports axis elements from a file. This menu item is described in [chapter 4.4.14 "Importing" on page 149](#)
- **Compare...** - Compares and merges objects, see [chapter 4.4.15 "Comparing/merging" on page 151](#)
- **Runtime Licenses...** - Refer to [chapter 4.4.16 "Runtime Licenses" on page 153](#)
- **Delete** - Deletes the complete axis from the project in **Offline mode**.
- **Rename** - Changes the axis name in **offline mode**.



The axis reference to the PLC also changes with the axis name. This can cause the application being compiled with errors.

- **Find Element...** - Allows scanning for elements in the project
- **New View** - Displays the axis in an individual window
- **Properties...** - Refer to [chapter "Axis properties" on page 220](#)

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 disabled are provided for export.

6.5 Dialogs

6.5.1 Overview on dialogs

When opening an axis folder, the dialogs for that axis are displayed below the axis node. These dialogs allow accessing status displays and axis settings. Most of the dialogs are only available in online mode or during offline parameterization.

A part of the dialogs is only used for Robot-Control V1.

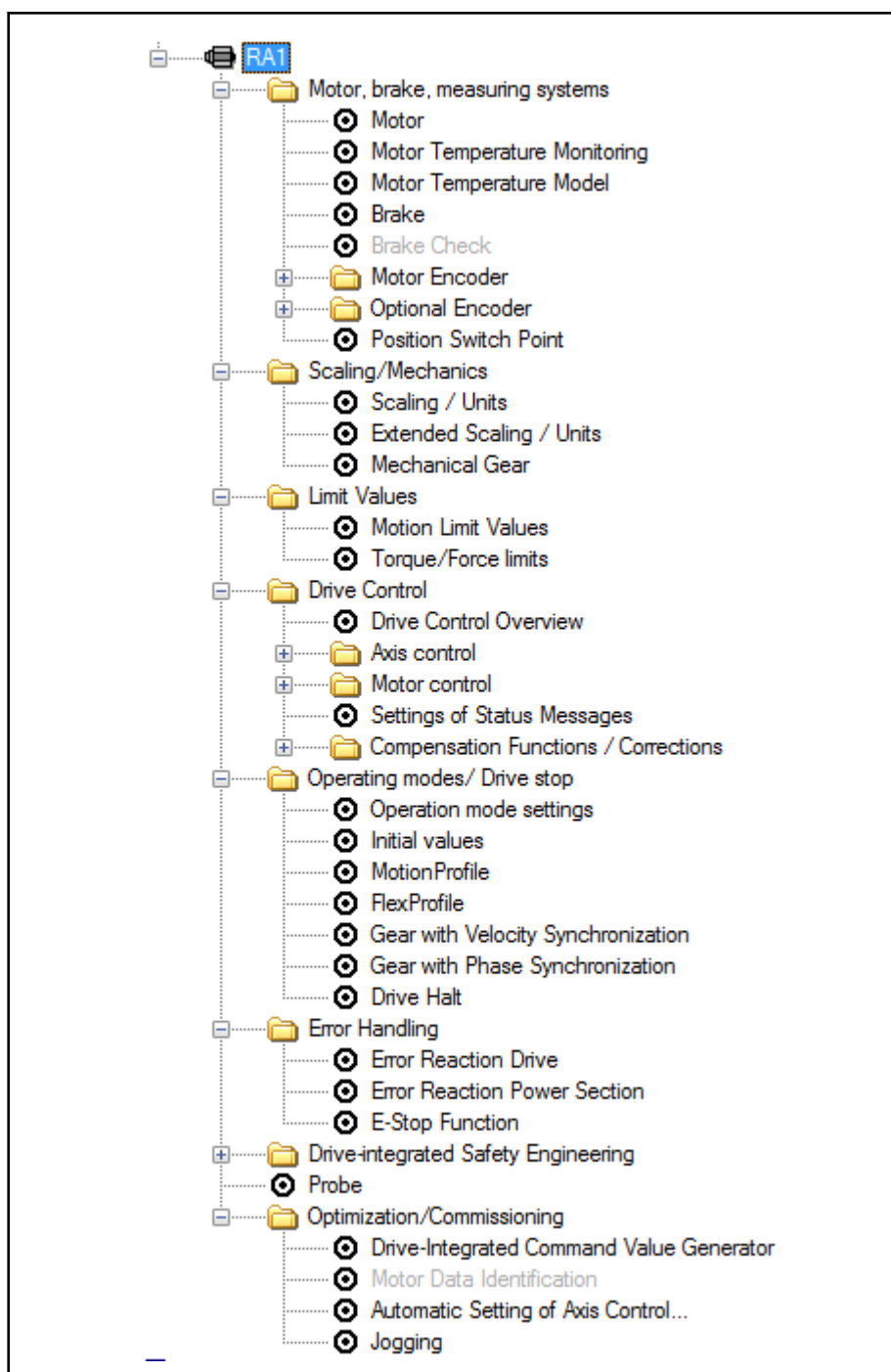


Fig. 6-23: Dialogs of a real axis (IndraDrive)

The presets already cover many use cases. A modification is only required in exceptional cases.

Depending on the axis type, different dialog scopes are available.

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	SercosDrive	VA	EncA
Operation mode settings	x	x	x		
Motor	x	x			

Axes

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	SercosDrive	VA	EncA
Motor temperature monitoring	x	x			
Brake	x	x			
Measuring systems	x	x			
Setting: Scaling/units	x	x	x	x	x
Extended scaling/units	x	x	x	x	x
Mechanical gear	x	x			
Closed-loop control	x	x			
Status message settings	x	x	x	x	
Drive Halt (AH)	x	x			
Establishing the position data reference	x	x			
Error reaction	x	x			
E-Stop function	x	x			
Motion limit values	x	x	x	x	
Initial values	x	x	x	x	
Compensation functions/corrections	x	x			
Drive-integrated command value generator	x	x			
Cam table	x				
MotionProfile		x	x	x	
FlexProfile		x	x	x	
Gear with velocity synchronization	x	x	x	x	
Gear with phase synchronization	x	x	x	x	
Axis configuration					x
Drive-integrated safety engineering	x	x			
Position switching point	x	x			
Touch probe	x	x			

VA
EncA
IPO

Virtual axis
Encoder axis
Interpolation

Tab. 6-6:

Dialogs of different axis types



If IndraDrive drive units are disabled at "Interpolation in the Drive", EcoDrive CS and SercosDrive, the drive is controlled via PackProfile (refer to the following figure).

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.

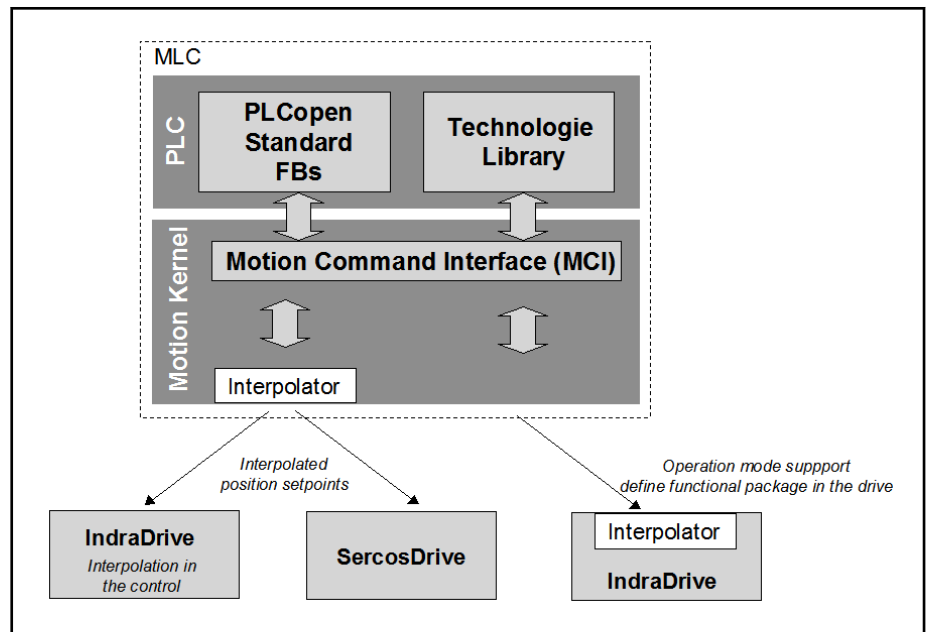


Fig. 6-24: Control principle for different drives

- [chapter 6.5.2 "Operation mode settings" on page 238](#)
- [chapter 6.5.3 "Motor" on page 240](#)
- [chapter 6.5.4 "Motor temperature monitoring" on page 241](#)
- [chapter 6.5.5 "Brake" on page 242](#)
- [chapter 6.5.6 "Measuring systems" on page 242](#)
- [chapter 6.5.7 "Settings: scaling/units" on page 245](#)
- [chapter 6.5.8 "Scaling/units extended" on page 246](#)
- [chapter 6.5.9 "Mechanical gear" on page 248](#)
- [chapter 6.5.10 "Closed-loop control" on page 249 \(reference to the drive documentation\)](#)
- [chapter 6.5.11 "Settings of status messages" on page 250](#)
- [chapter 6.5.12 "Drive halt" on page 251](#)
- [chapter 6.5.13 "Establishing dimensional reference" on page 252](#)
- [chapter 6.5.14 "Error response" on page 254](#)
- [chapter 6.5.15 "E-STOP function" on page 256](#)
- [chapter 6.5.16 "Motion limit values" on page 257](#)
- [chapter 6.5.17 "Initial values" on page 259](#)
- [chapter 6.5.18 "Compensation functions/corrections" on page 261 \(reference to the drive documentation\)](#)
- [chapter 6.5.19 "Drive-integrated command value generator" on page 263 \(reference to the drive documentation\)](#)
- [chapter 6.5.20 "Cam table" on page 263](#)
- [chapter 6.5.21 "MotionProfile" on page 269](#)
- [chapter 6.5.22 "FlexProfile" on page 275](#)
- [chapter 6.5.23 "Gear with velocity synchronization" on page 282](#)
- [chapter 6.5.24 "Gear with phase synchronization" on page 286](#)
- [chapter 6.5.26 "Drive-integrated safety engineering" on page 292](#)

Axes

- [chapter 6.5.27 "Position switching point" on page 292](#)
- [chapter 6.5.28 "Touch probe" on page 293](#)

The values in the dialogs are partially calculated from multiple parameters. If a clear assignment exists, it is given.



Dialogs exclusively referring to internal drive data are only described by a reference to the original product manuals.

Open the following dialogs via the context menu of the drive:

Dialog	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	SercosDrive	VA	EncA
chapter 6.5.29 "Initial Commissioning" on page 296	x	x	x	x	x
chapter 6.5.30 "Parameters" on page 297	x	x	x	x	x
chapter 6.5.31 "Communication" on page 299	x	x	x		
chapter 6.5.32 "Diagnostics" on page 304	x	x	x	x	x

VA
EncA
IPO

Virtual axis
Encoder axis
Interpolation

Tab. 6-7: Dialogs of different axis types to be opened via the context menu

6.5.2 Operation mode settings



This dialog box is available for the interpolation in the drive and in the control.

Call **Axis ► Operation modes / Drive Halt ► Operation mode settings**

Target Use this dialog for different operation mode settings.

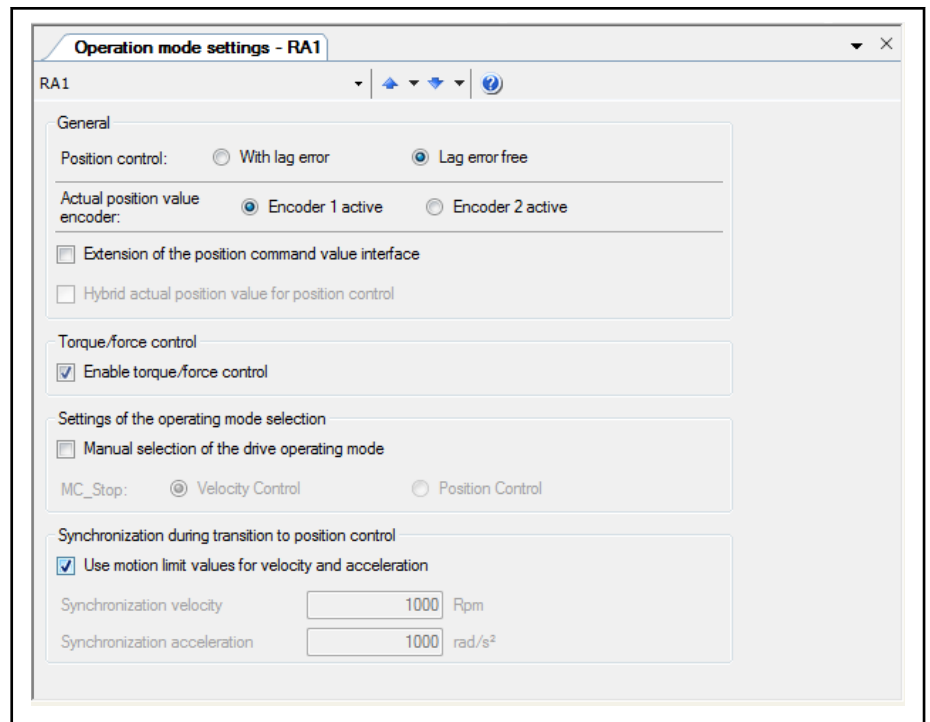


Fig. 6-25: "Operation mode settings" dialog

General information

The settings are applied on all operation mode selected via PLCopen function blocks:

- Type of position control:
In a position-controlled mode, it can be differentiated between an operation with and without lag. The lag error is the difference between the position command value and the actual position value
- Selection of the position encoder:



Only for the drive type "IndraDrive" and "SercosDrive".

This setting determines which encoder provides the actual position values

- Extension of the position control interface (for IndraDrive and Hydraulic-Drive):

For operation modes to transfer position command values to the drive ("S-0-0047, Position command value"), the accuracy of the values in the drive can be increased. The accuracy of the parameter S-0-0047 is used to increase the parameter "P-0-0100, Position command value extension". The parameter P-0-0100 is additionally configured in the master data telegram (MDT).

This is possible for all operation modes with "interpolation in the control" as well as for operation modes with "interpolation in the drive" using the parameter S-0-0047 as command value specification (positioning).

- Actual hybrid position value (encoder 2 selected):

The actual position value is generated from the motor encoder and the optional encoder. This value can be smoothed using the parameter P-0-0241 "Actual position smoothing time constant for hybrid position control".

Axes

It is used for hybrid position control or measuring wheel mode (refer to the IndraDrive Functional Description)



Only for the drive type IndraDrive.

Torque/force control

This setting removes the torque/force command value from the cyclic Sercos telegram configuration (MDT).

This is required for the HydraulicDrive to operate the axis with the so-called "alternating control". That means that the axis is in position control due to a PLCopen function block and that the torque command value is additionally specified or controlled (not with a PLCopen function block, but separately via UserCmdData).

Settings of the operation mode selection of the drive

This setting can be used (for an IndraDrive with interpolation in the control) to set for the PLCopen function block MC_Stop whether the drive moves in a position-controlled or speed-controlled operation mode.

Make the settings here for a position-controlled stop (with the function block MC_Stop).

For default applications - especially with a high inertia - speed-controlled stopping is preferred.

Synchronization at position control transition

Here, user-specific values for velocity and acceleration are set to allow a user-specific transition from velocity to position control (for an IndraDrive with interpolation in the control).

The respective limit values for velocity and acceleration are set for a fast transition when changing from velocity to position control.

Involved parameter

- A-0-0007, Axis configuration

6.5.3 Motor



This dialog is available for IndraDrive for interpolation in the drive and in the control.

The screenshot shows a software dialog titled "Motor - Axis [1] RA1". It contains several input fields and a radio button group. The "Type" field is set to "MKD025B-144-KG0-KN". The "Functional principle" field is set to "synchronous, w/o reluctance torque". The "Design" field is set to "Rotary". The "Cooling type" section has five radio buttons: "Non-ventilated" (selected), "Ventilated", "Liquid-cooled", "Improved cooling", and "Thermal effect via motor flange". The "Data in encoder memory" section has a "Type" field set to "MKD025B-144-KG0-KN" and a "Version of data structure" field set to "1".

Fig. 6-26: "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, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Motor, axis mechanics, measuring systems" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.4 Motor temperature monitoring



This dialog is available for IndraDrive for interpolation in the drive and in the control.

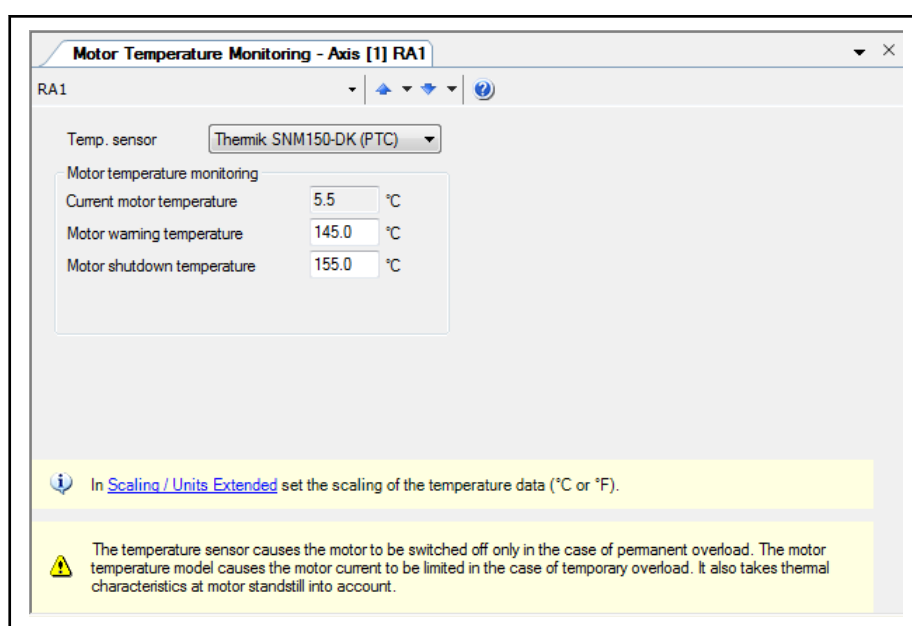


Fig. 6-27: "Motor Temperature Monitoring" dialog

The dialog sets the sensor type for the motor temperature monitoring. Furthermore, it informs on the thermal motor data.

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	
Motor warning temperature	S-0-0201	
Motor shutdown temperature	S-0-0204	
Thermal time constant of motor	P-0-4035	
Thermal time constant of winding	P-0-4034	
Thermal short-time overload of winding	P-0-4037	

Axes

6.5.5 Brake



This dialog is available for IndraDrive for interpolation in the drive and in the control.

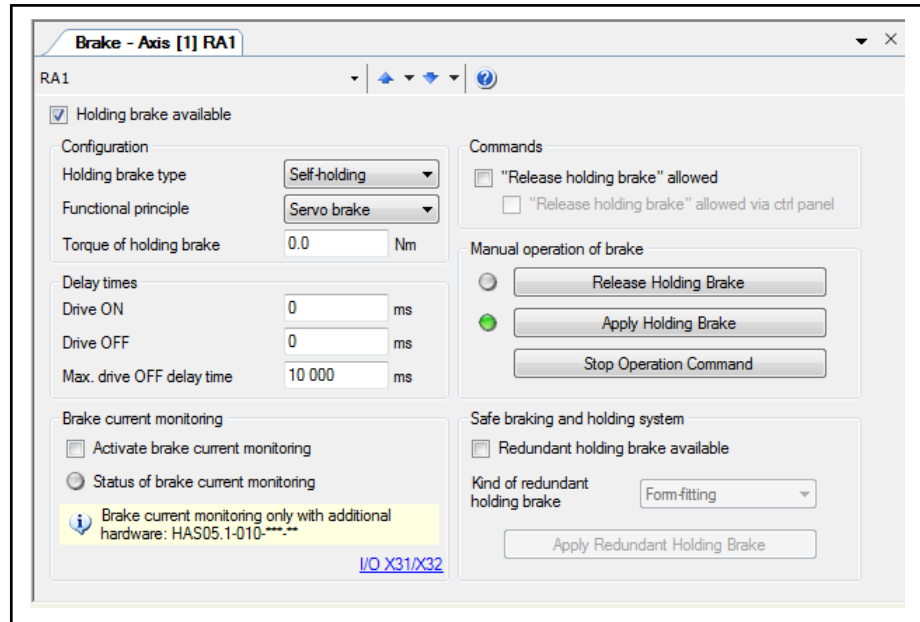


Fig. 6-28: "Brake" dialog

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
Nominal load of holding system		P-0-0547
Waiting time of the drive ON		S-0-0206
Waiting time of the drive OFF		S-0-0207
Maximum drive off delay time		S-0-0273

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Motor, axis mechanics, measuring systems" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.6 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

Axes

- [chapter "Settings of Optional Encoder" on page 244](#), the encoder is located at the load to compensate, e.g. for disturbance variables of the gear
- [chapter "Configuring the measuring encoder" on page 231](#), 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.

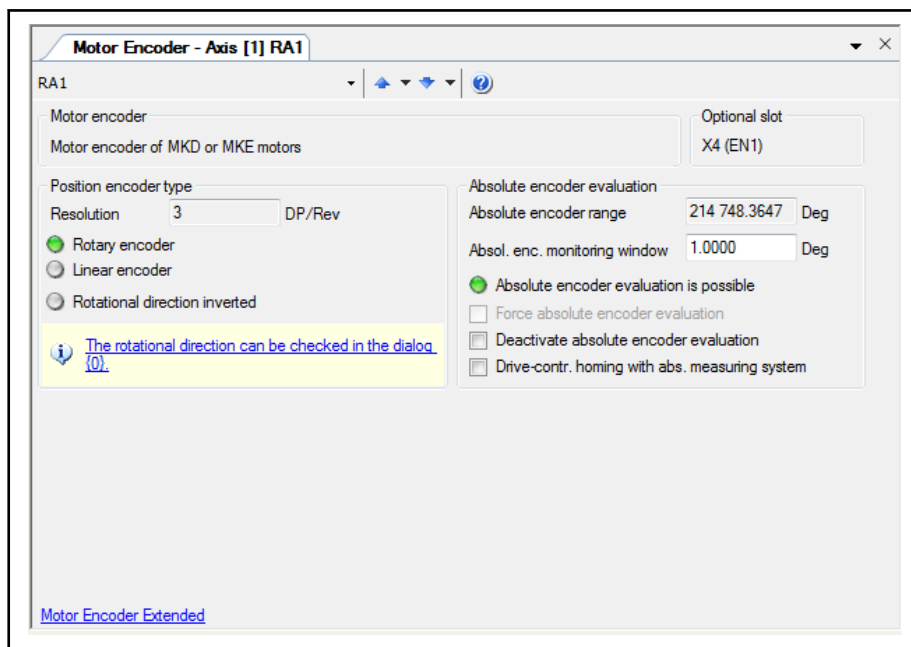


Fig. 6-29: "Motor Encoder" dialog

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Position encoder type 1	S-0-0277	
Encoder 1 resolution	S-0-0116	
Maximum traversing range	S-0-0278	
Multiplication 1 (motor encoder)	S-0-0256	
Position data format, internal	P-0-0129	
Gear 1 motor side (motor encoder)	P-0-0121	
Gear 1 encoder side (motor encoder)	P-0-0122	
Transmission ratio	(P-0-0122)/(P-0-0121)	
Position encoder type 1	S-0-0277	

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Motor, axis mechanics, measuring systems" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Axes

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:

- [chapter "Settings, Motor Encoder" on page 242](#) the encoder is located on the motor shaft itself
- Optional encoder. This encoder is located on the load side to compensate, e.g. for disturbance variables of the gear
- [chapter "Configuring the measuring encoder" on page 231](#), 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.

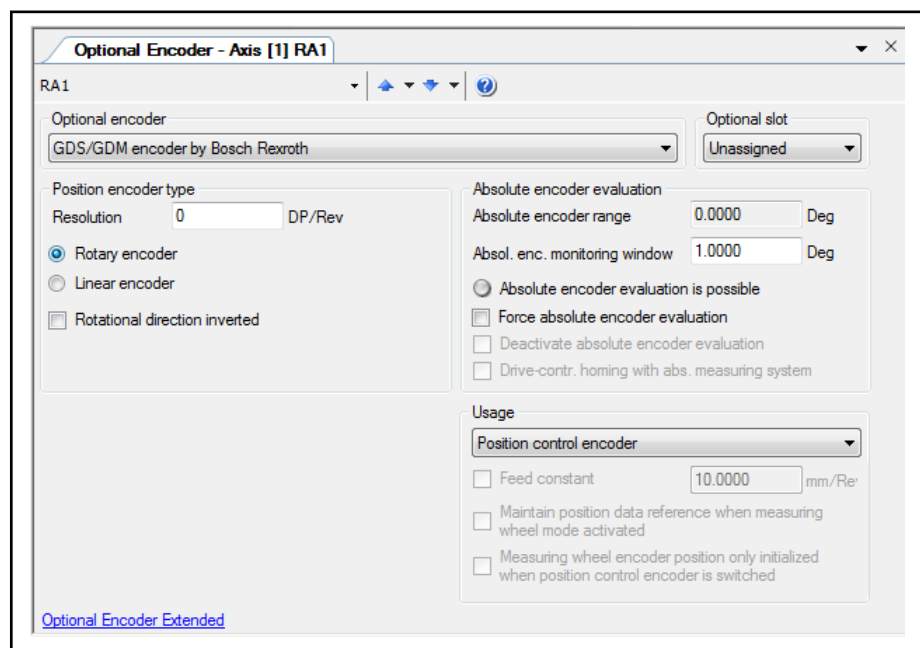


Fig. 6-30: "Optional Encoder" dialog

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Assignment optional encoder → optional slot		P-0-0078
Encoder 2 resolution		S-0-0117
Maximum traversing range		S-0-0278
Multiplication 2 (optional encoder)		S-0-0057
Position data format, internal		P-0-0129
Gear 2 load side (optional encoder)		P-0-0124
Gear 2 encoder side (optional encoder)		P-0-0125

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)
Transmission ratio	(P-0-0125)/(P-0-0124)	
Position encoder type 2	S-0-0115	

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Motor, axis mechanics, measuring systems" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.7 Settings: scaling/units



This dialog is available for all axis types.

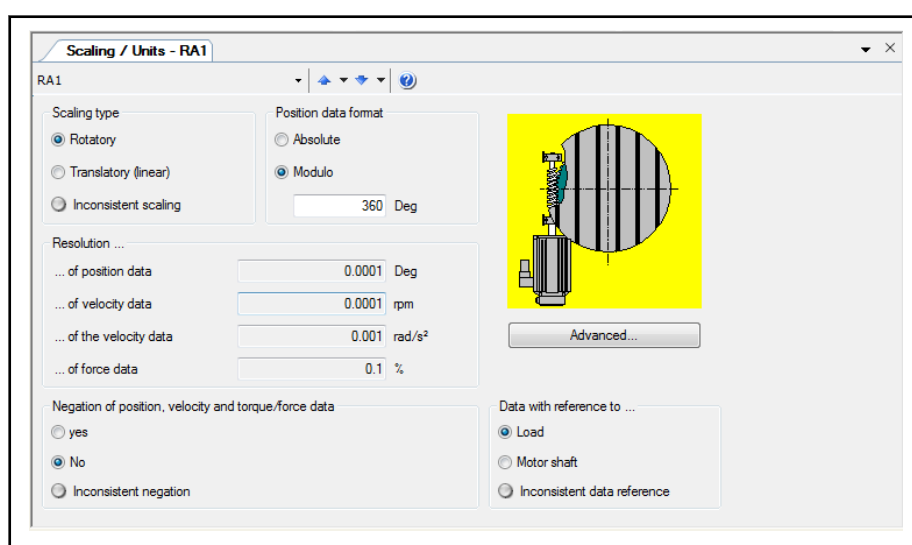


Fig. 6-31: "Scaling/Units" dialog (in the example, settings of the real axis)

The "Scaling/Units" dialog of settings derives the presets 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

Tab. 6-8: 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" (for a description, refer to <http://de.wikipedia.org>).

Max. traversing velocity = (0.5 x modulo [degree or mm]) / cycle time [s].

Axes

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
8	3750
16	1875

Tab. 6-9: 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)	SercosDrive	Virtual axis	Encoder axis
Scaling type for position data "A-0-0059"	x	x	x	x	x
Scaling type for velocity "A-0-0056"	x	x	x	x	x
Scaling type for acceleration "A-0-0053"	x	x	x	x	x
Scaling type for torque/force data "A-0-0050"	x	x			
Modulo value "A-0-0045"	x	x	x	x	
Feed distance of the slave drive "A-0-0046"	x	x	x	x	x

Tab. 6-10: Parameters for the different scaling types

Detailed data can be entered into [chapter 6.5.8 "Scaling/units extended"](#) on [page 246](#).

6.5.8 Scaling/units extended



This dialog is available for all axis types.

The dialog box supplements data to [chapter 6.5.7 "Settings: scaling/units"](#) on [page 245](#) in detail.

If the axis is an IndraDrive, the following dialog is displayed:

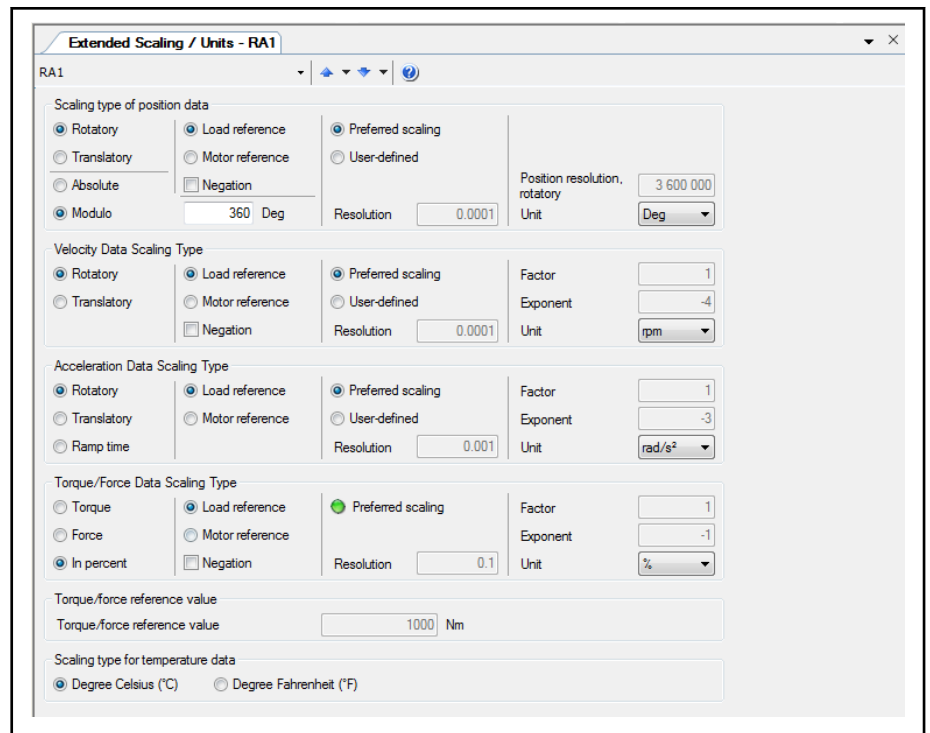


Fig. 6-32: "Scaling/Units Extended" dialog for IndraDrive axes

The dialog of the other axis types provides respectively less setting options.

The following parameters are analyzed:

Setting	IndraDrive (IPO drive)	IndraDrive (IPO control)	Sercos Drive	Virtual axis	Encoder axis
Position					
Scaling type for position data "A-0-0059"	x	x	x	x	x
Scaling factor for translatory position data "A-0-0060"	x	x	x		
Scaling exponent for translatory position data "A-0-0061"	x	x	x		
Position resolution, rotary "A-0-0062"	x	x	x		
Position polarities "A-0-0029"	x	x	x	x	
Modulo value "A-0-0045"	x	x	x	x	
Feed distance of the slave drive "A-0-0046"	x	x	x	x	x
Velocity					
Scaling type for velocity "A-0-0056"	x	x	x	x	x
Scaling factor for velocity data "A-0-0057"	x	x	x		
Scaling exponent for velocity data "A-0-0058"	x	x	x		
Velocity polarity "A-0-0048"	x	x			
Acceleration					
Scaling type for acceleration "A-0-0053"	x	x	x	x	x
Scaling factor for acceleration data "A-0-0054"	x	x			

Axes

Setting	IndraDrive (IPO drive)	IndraDrive (IPO control)	Sercos Drive	Virtual axis	Encoder axis
Scaling exponent for acceleration data "A-0-0055"	x	x			
Force/torque					
Scaling type for torque/force data "A-0-0050"	x	x			
Scaling type for torque/force data "A-0-0051"	x	x			
Scaling exponent for torque/force data "A-0-0052"	x	x			
Temperature					
Scaling type for temperature data "A-0-0063"	x	x			

User-defined scaling A user-defined scaling (selection with scaling factor and exponent) can be set via the scaling parameters (bit 3) for force, acceleration, velocity and position.

1. A-0-0050 "Scaling type for torque/force data"
2. A-0-0053 "Scaling type for acceleration data"
3. A-0-0056 "Scaling type for velocity data"
4. A-0-0059 "Scaling type for position data"

Depending on the axis type, a user-defined scaling is not always supported for all sizes.

User-defined scaling	IndraDrive (IPO drive)	IndraDrive (IPO control)	Sercos Drive	Virtual axis	Encoder axis
Position	x				
Velocity	x				
Acceleration	x				
Force¹⁾	x	x			

1) For drive firmware MPX08 or MPX17 and higher

6.5.9 Mechanical gear



This dialog is available for IndraDrive for interpolation in the drive and in the control.

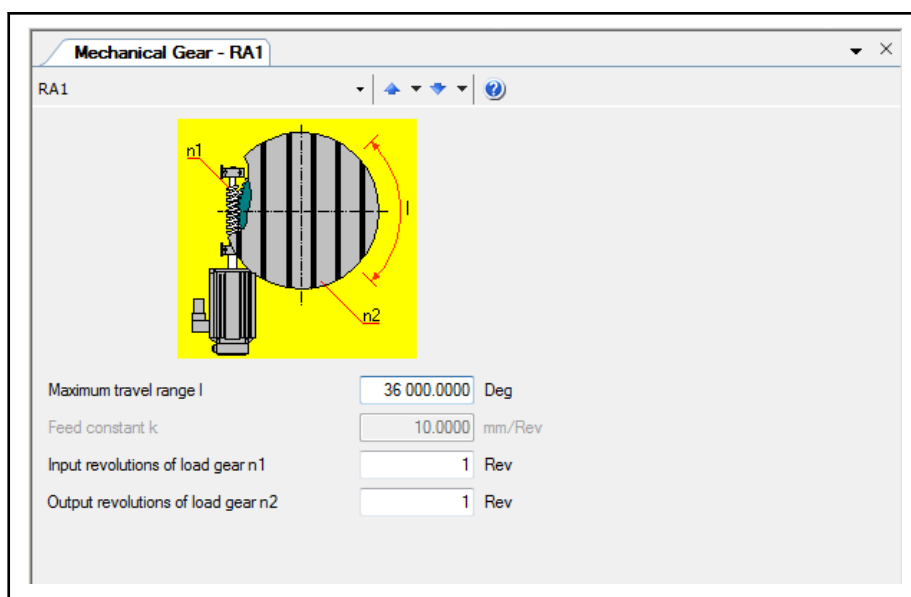


Fig. 6-33: "Mechanical Gear" dialog

The data 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 typ. value 10.00 mm/rev.)

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Motor, axis mechanics, measuring systems" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.10 Closed-loop control



This dialog box is available for the interpolation in the drive and in the control.

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Overview on the drive control" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Support is provided by the implemented online help.

Axes

6.5.11 Settings of status messages



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual axis.

The dialog box evaluates the status of the axis (A-0-0021).

The following messages depend on whether the respective value is in its window.

- "AA-0-0021, Standstill (bit 5)"
- "AA-0-0021, Actual velocity = velocity command value (bit 4)"
- "AA-0-0021, In position (bit 6)"
- "A-0-0021, Synchronous position (bit 7)" and
- "A-0-0021, Synchronous velocity" likewise (bit 7)

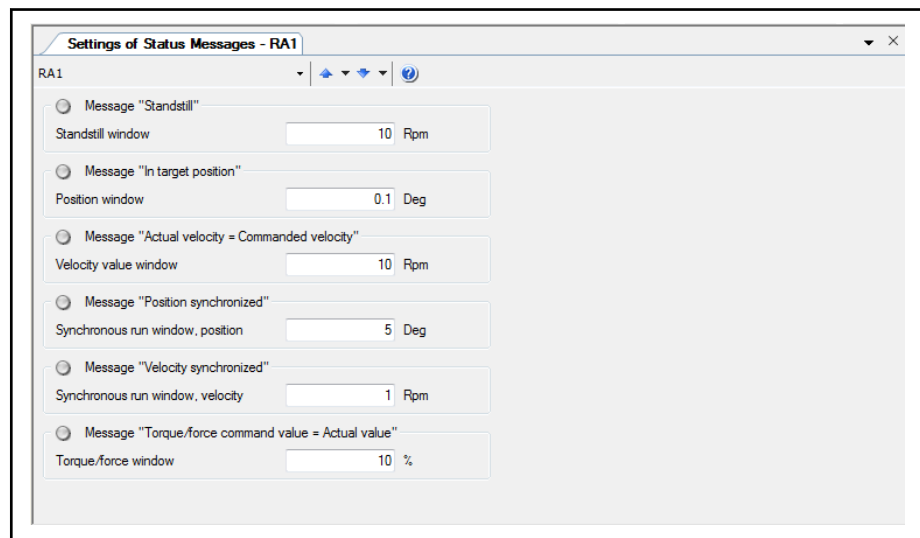


Fig. 6-34: "Settings of Status Messages" dialog

The values are filed in the following parameters:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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.

6.5.12 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.

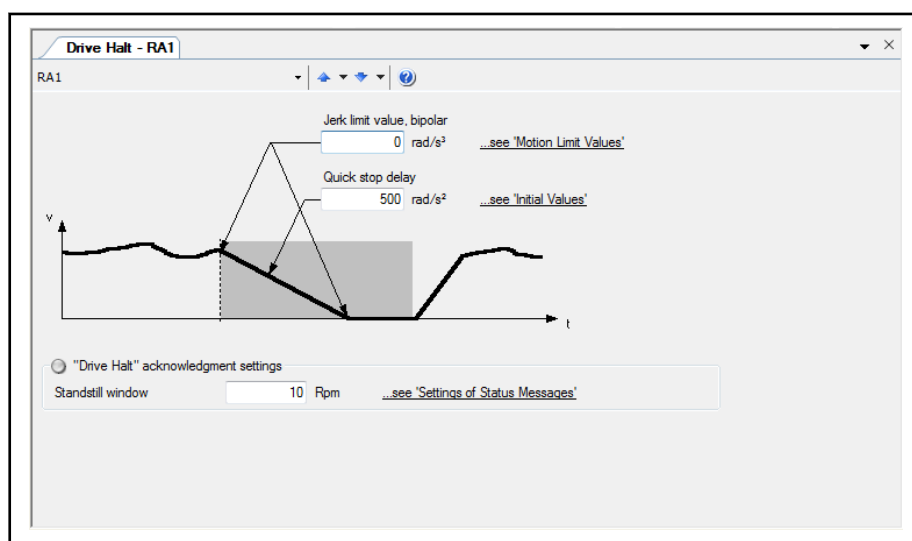


Fig. 6-35: "Drive Halt" dialog

The individual parameters are pre-adjusted in different dialogs:

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Jerk limit value, bipolar chapter 6.5.16 "Motion limit values" on page 257	A-0-0036	
Rapid stop deceleration chapter 6.5.17 "Initial values" on page 259	A-0-0224	

Axes

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
Standstill window chapter 6.5.11 "Settings of status messages" on page 250	A-0-0222	
Axis status fig. 6-83 "Status dialog – Axis status of a real axis" on page 304	A-0-0021 (Bit 15)	

6.5.13 Establishing dimensional reference

Motor encoder



This dialog box is available for the interpolation in the drive and in the control.

This dialog is used to set the dimensional reference of the motor encoder.

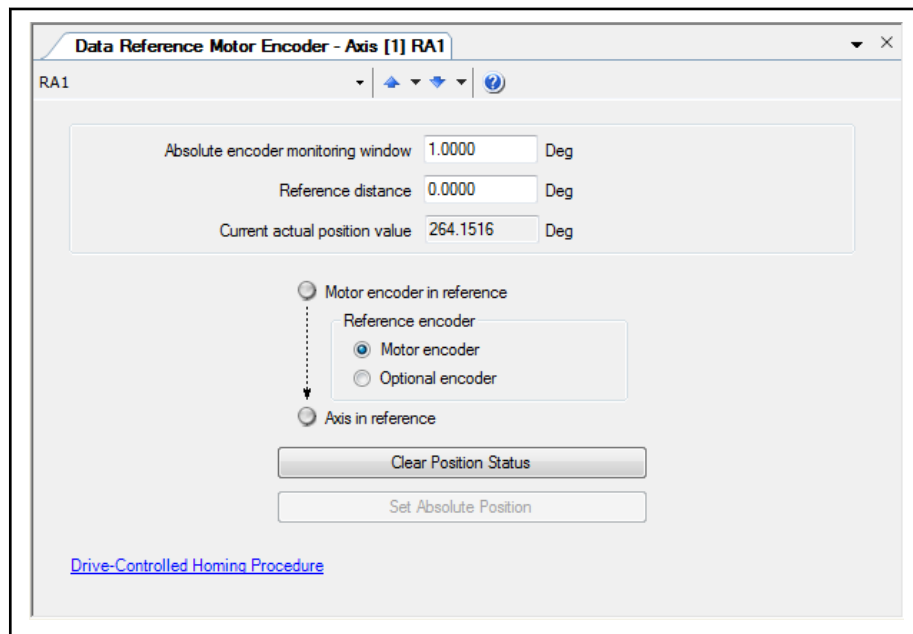


Fig. 6-36: "Data Reference Motor Encoder" dialog

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 entered into parameter S-0-0052 is applied at the reference position as actual position value.

S-0-0051, Actual position value encoder 1

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).

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. the actual position value 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. 6-37: "Data Reference Optional Encoder" dialog

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 entered into parameter S-0-0053 is applied 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).

Axes

**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. the actual position value is set to its initial value.

Furthermore, the following parameters are involved:

- S-0-0403, Actual position value status
- S-0-0147, Homing parameter

6.5.14 Error response**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.

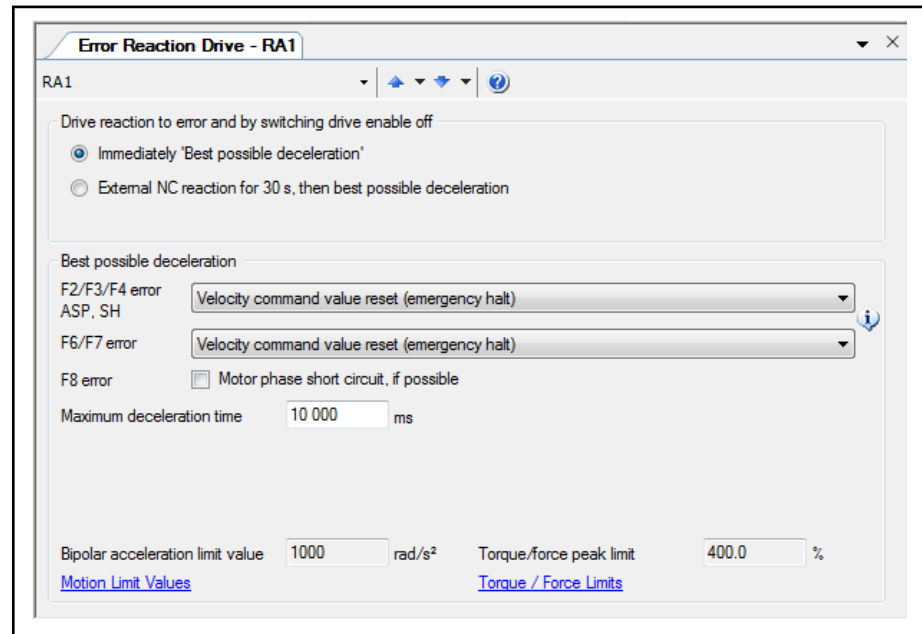


Fig. 6-38: "Error Reaction Drive" dialog

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.

NOTICE

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 "30 s NC external NC reaction, then best possible deceleration to standstill" (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:

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 function block. The slaves axis remains synchronized even in case of an error and delays with the master axis. After these 30 s, the best possible deceleration starts on the slave axis (the synchronous link is canceled in standstill). After eliminating the error cause, the error is deleted by the "MB_Command" (C-0-1030) function blocks and the PLCopen Motion function block 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 effective 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 errors" 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.

Axes

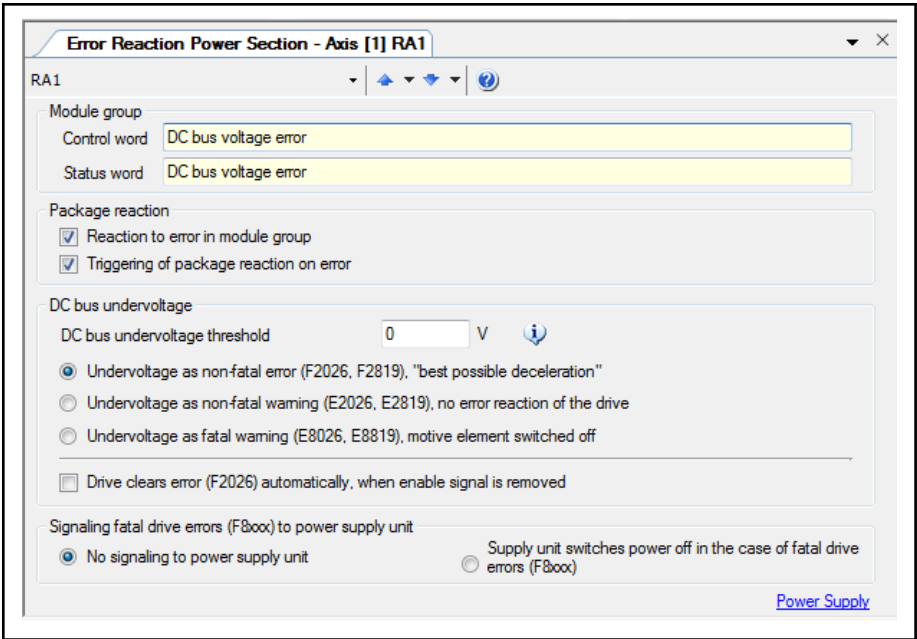


Fig. 6-39: "Error Reaction Power Section" dialog

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)!

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!

6.5.15 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. 6-40: "E-Stop function" dialog

By activating "P-0-0008, E-Stop input (Bit 0 = 1)" and allocating the bit 0 state of "P-0-0223, E-Stop input" to a digital input, the drive is initiated to execute the response defined to stop the drive defined via P-0-0008 if there is 0 V at the E-Stop.

"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 changed via the dialog box located behind the "Digital I/O 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	
Digital I/Os, assignment list	P-0-0223	
Digital I/Os, bit numbers	P-0-0301	



The input is always "0" active!

For details, refer to "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "E-STOP function" (see [chapter 1.4 "Required and supplementing documents" on page 18](#)).

6.5.16 Motion limit values



This dialog box is available for the interpolation in the drive and in the control.

Axes

The dialog provides an overview on the preset motion limit values of the real axis.

Fig. 6-41: "Motion Limit Values" dialog

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.

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	EncA
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				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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 value monitoring is enabled, "Actual position value (A-0-0100)" indicates the state of the axis and the adjustable "A-0-0030, Positive position limit value", "A-0-0031, Negative position limit value" show the admissible 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	
Errors	F6116030	F6116029

Tab. 6-11: 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-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	

Tab. 6-12: 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.

6.5.17 Initial values



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual axis.

Axes

The dialog box displays the initial values used for the PLCopen function blocks. Starting point is the state after reboot:

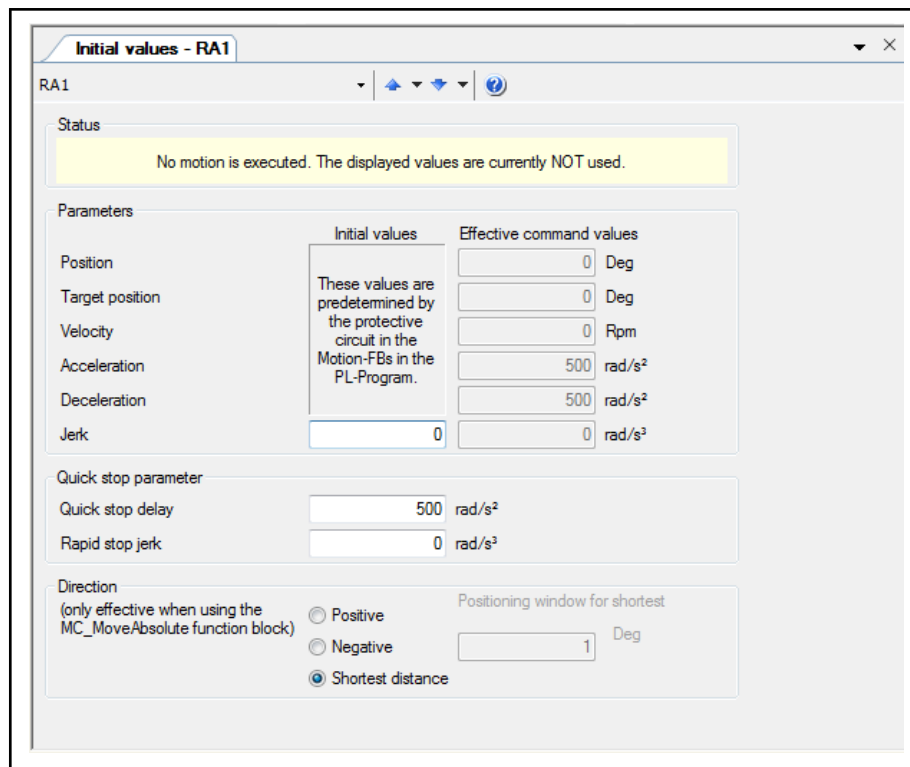


Fig. 6-42: Initial values after a reboot

The following parameters are evaluated:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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.

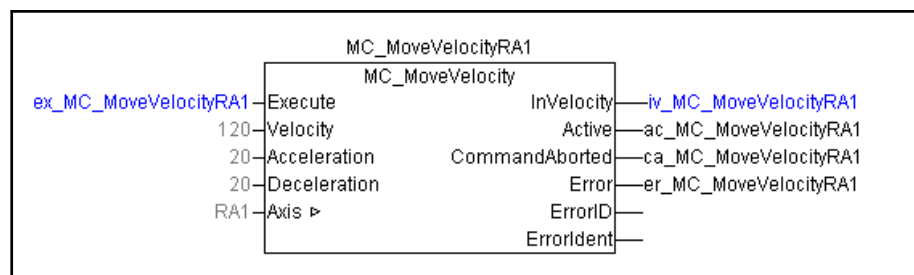


Fig. 6-43: Active PLCopen function block: MC_MoveVelocity

Fig. 6-44: Command values based on the initial values and the function block

The velocity (120 min⁻¹), acceleration (20 rad/s²) and deceleration (20 rad/s²) values were applied by the function block.

The previous position and distance values were kept. They would change in case of an active MC_MoveRelative for example.

6.5.18 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 regard-

Axes

ing 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.

The basics of the "Compensation functions, corrections" described with the corresponding products "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Compensation functions, corrections" (refer to [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

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.

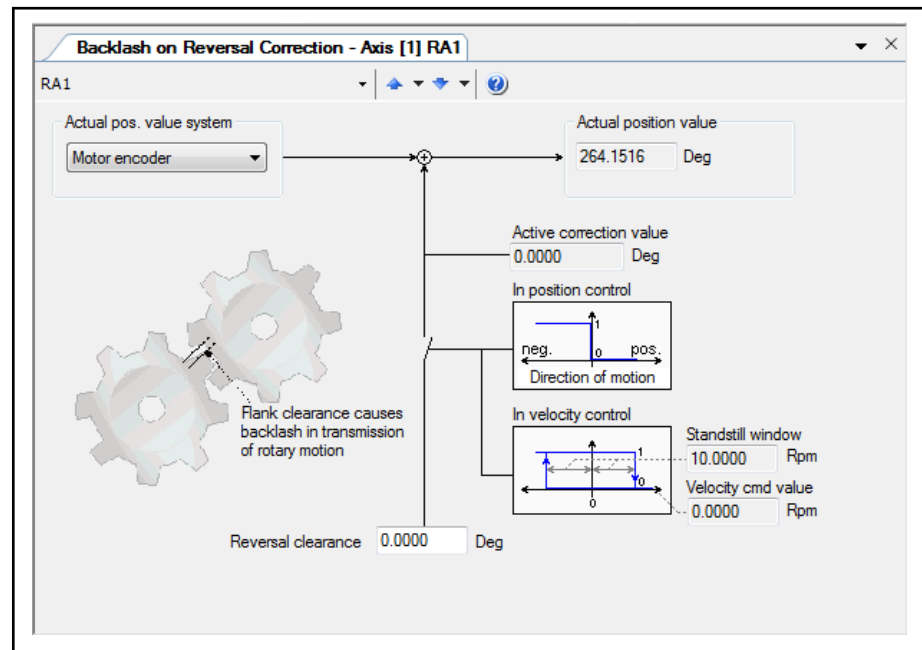


Fig. 6-45: "Backlash on reversal correction" dialog

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.

6.5.19 Drive-integrated command value generator



This dialog is available for IndraDrive for interpolation in the drive and in the control.

The basics of the "drive-integrated command value generators" described with the corresponding products "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Drive-integrated command value generators" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Support is provided by the implemented online help.

6.5.20 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".

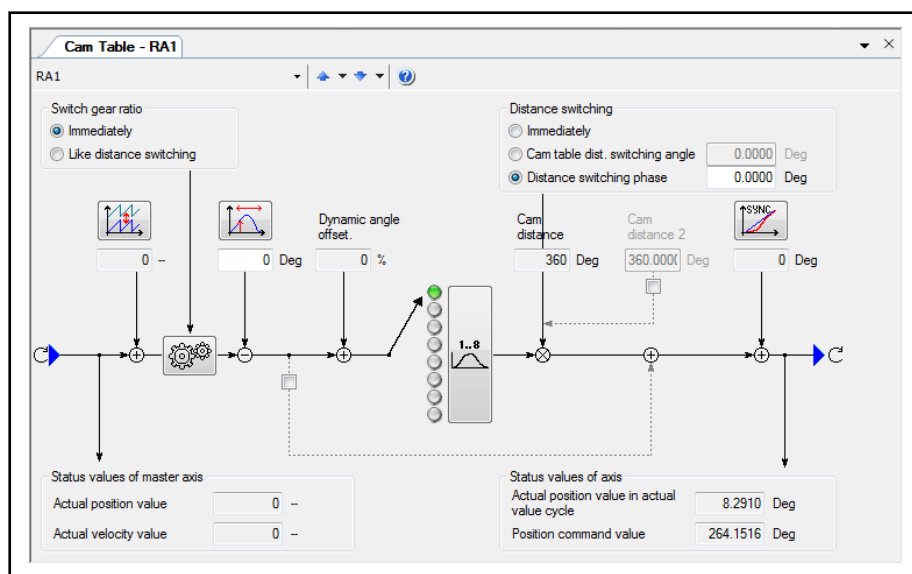


Fig. 6-46: "Cam Table" dialog

Parameters used:

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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				
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 IndraLogic XLC.

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 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.

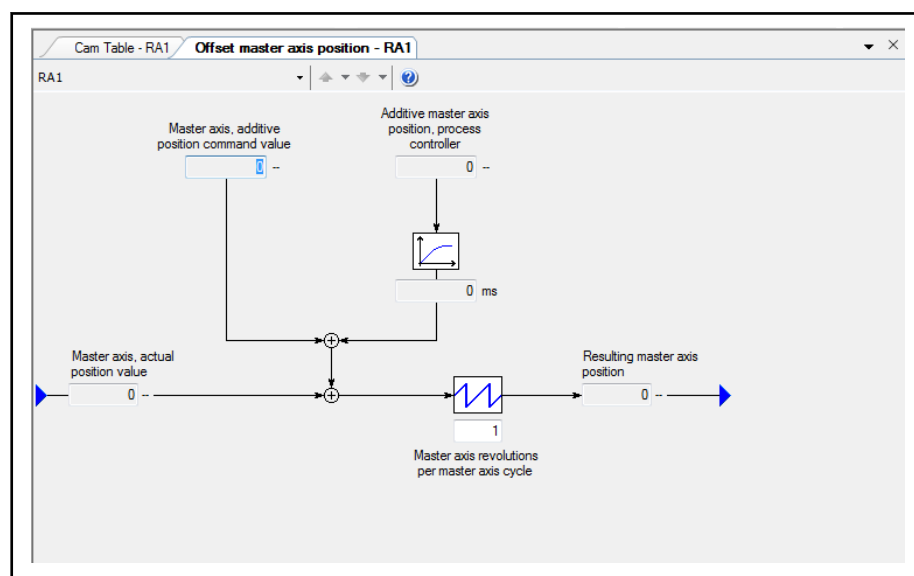


Fig. 6-47: "Offset Master Axis Position" dialog in the "Cam Table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

Axes

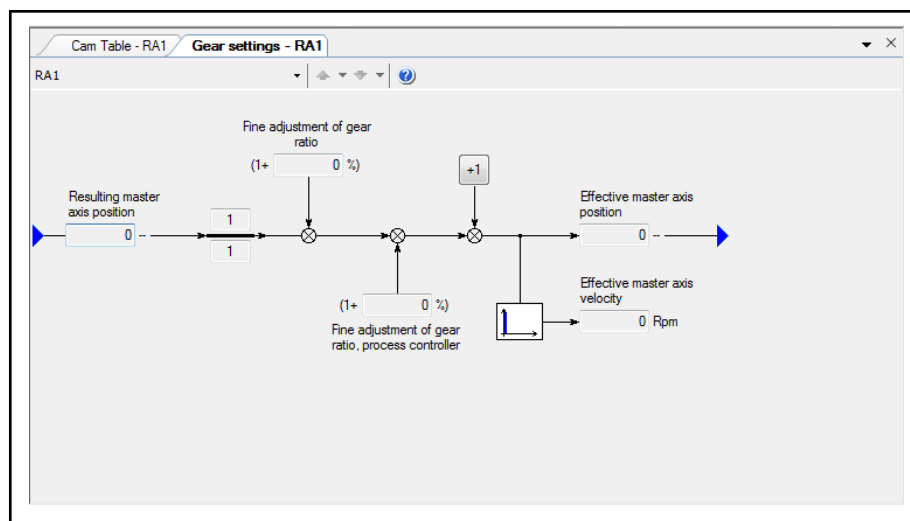


Fig. 6-48: "Gear Settings" dialog in the "Cam Table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

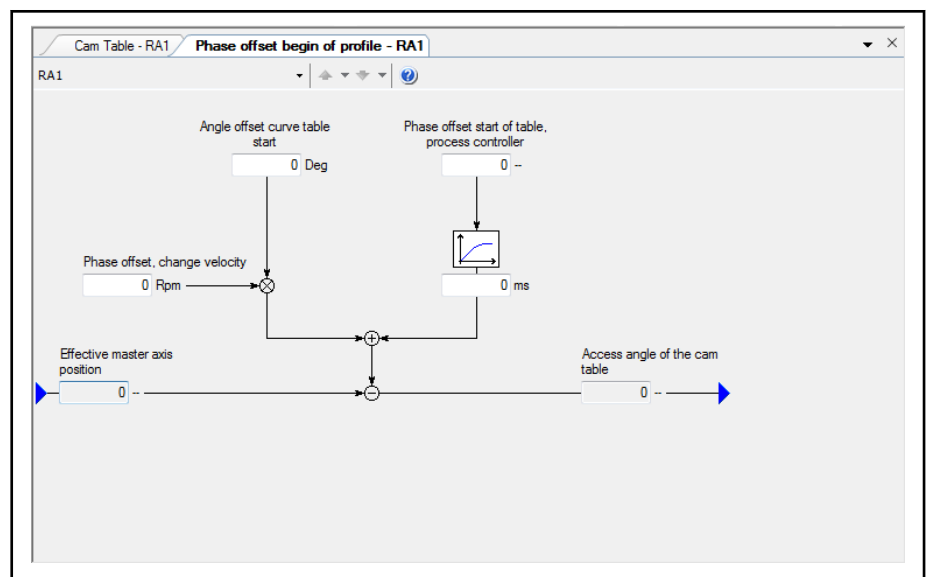


Fig. 6-49: Dialog "Phase offset begin of profile" in the "Cam Table"

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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 IndraLogic XLC.

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 cam table to be edited is selected in the "MC_CamIn" Motion function block. Furthermore, for each cam table, either linear and cubic spline interpolation can be selected.

Cam tables can be loaded from a parameter file via the  icon.

Cam tables can be stored as parameter file via the  icon.

Axes

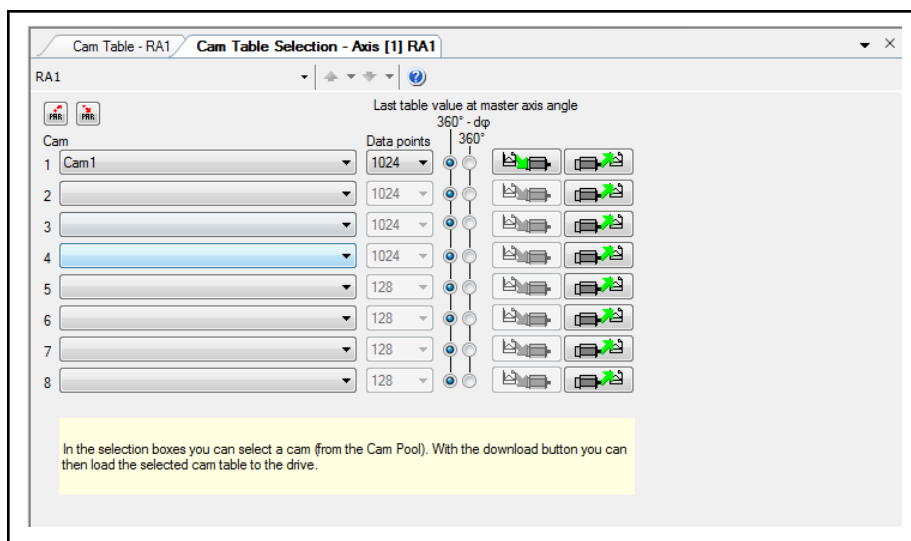


Fig. 6-50: "Cam Table Selection" dialog in the "Cam Table" dialog

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".

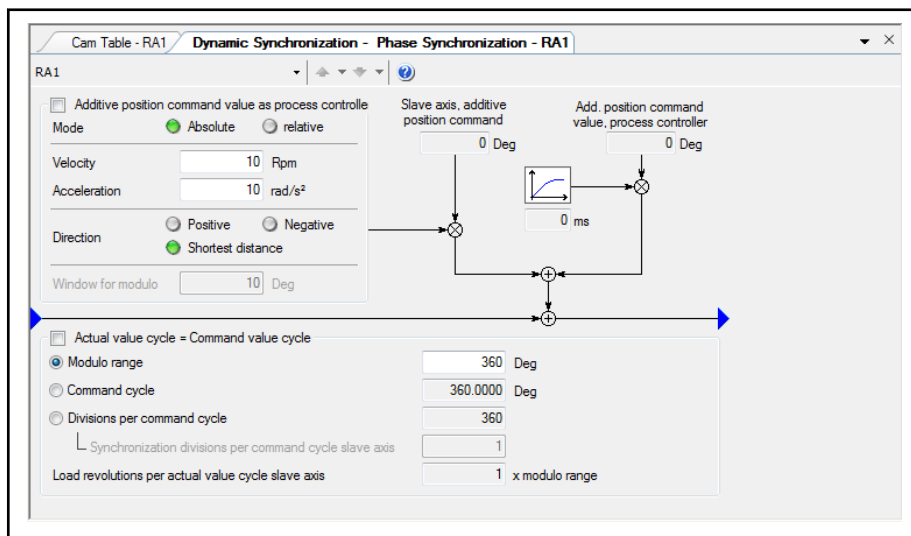


Fig. 6-51: "Dynamic Synchronization - Phase Synchronization" dialog in the "Cam Table" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
Synchronization mode	P-0-0155	A-0-2794			
ELS, synchronization mode (SercosDrive and virtual axes)	A-0-2794				
ELS, synchronization velocity	A-0-2790				

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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				

6.5.21 MotionProfile

General information



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual 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".

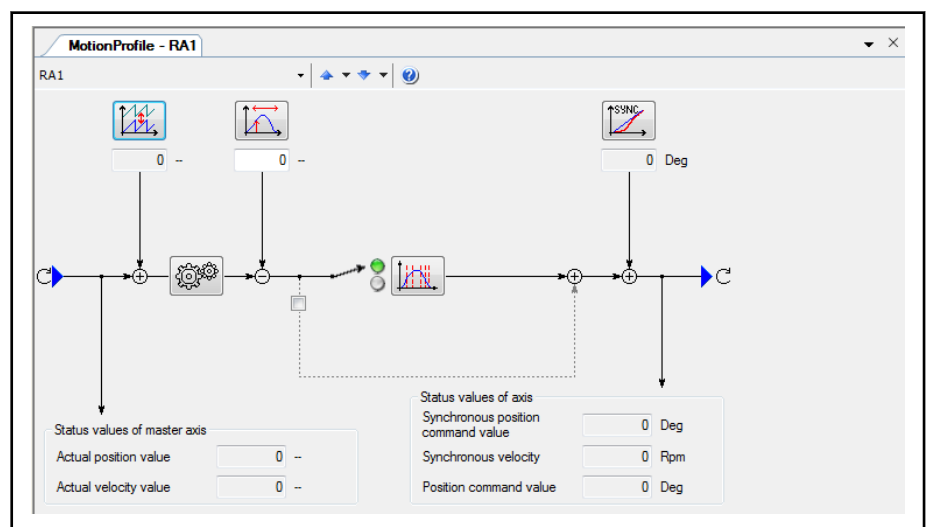


Fig. 6-52: "MotionProfile" dialog

Parameters used:

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
ELS, master axis, axis number			A-0-2701		
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".

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 IndraLogic XLC.

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.

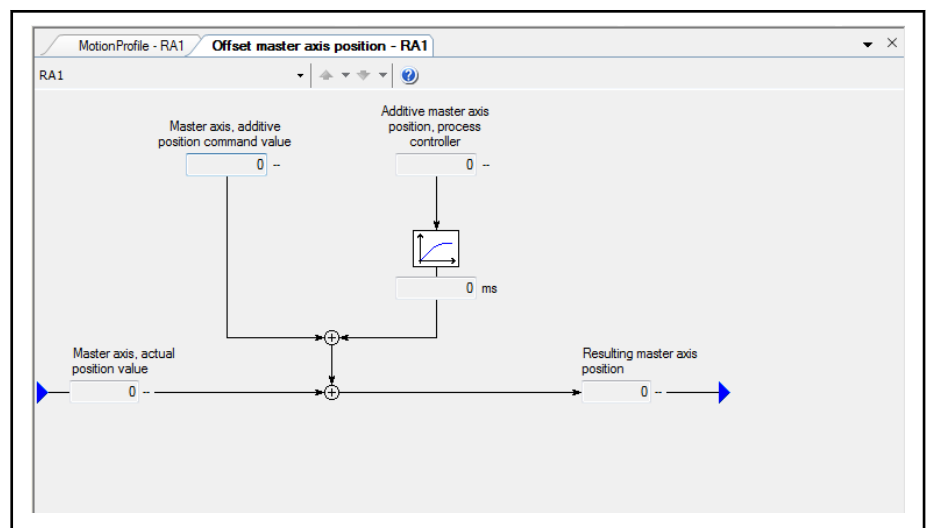


Fig. 6-53: "Offset Master Axis Position" dialog in the "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		
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".

Axes

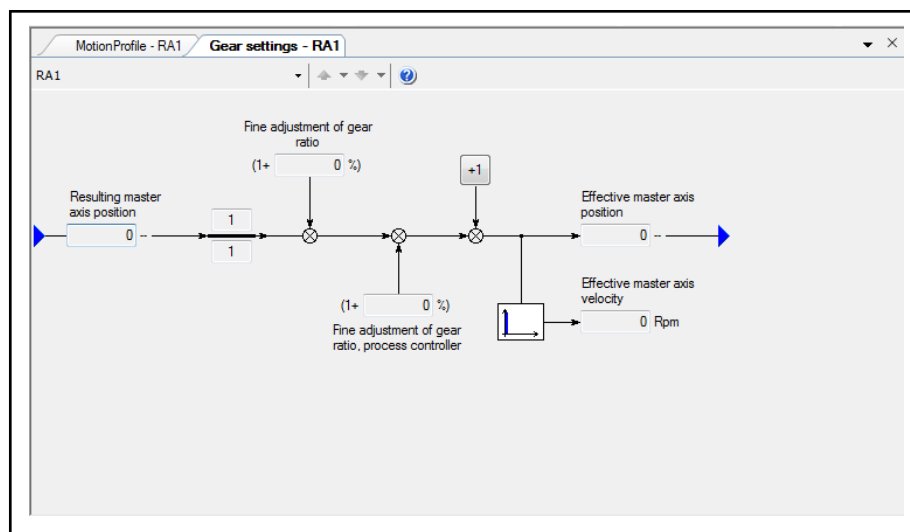


Fig. 6-54: "Gear Settings" dialog in the "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

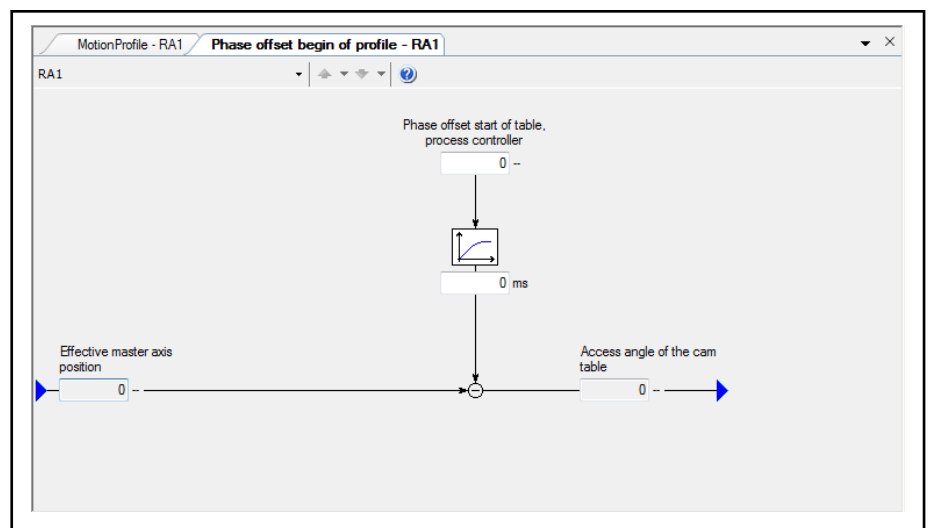


Fig. 6-55: Dialog "Phase offset begin of profile" in the "MotionProfile"

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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 IndraLogic XLC.

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.

Axes

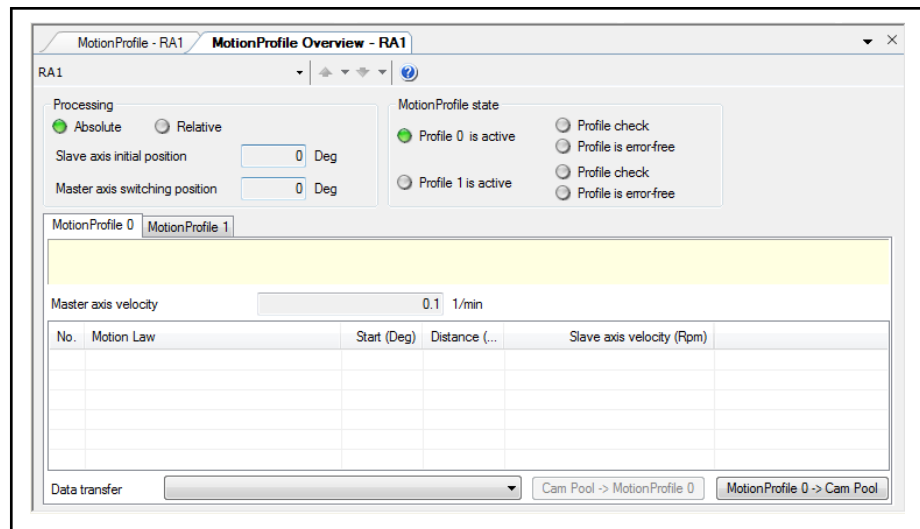


Fig. 6-56: "MotionProfile Overview" dialog in the "MotionProfile" dialog
Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		
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".

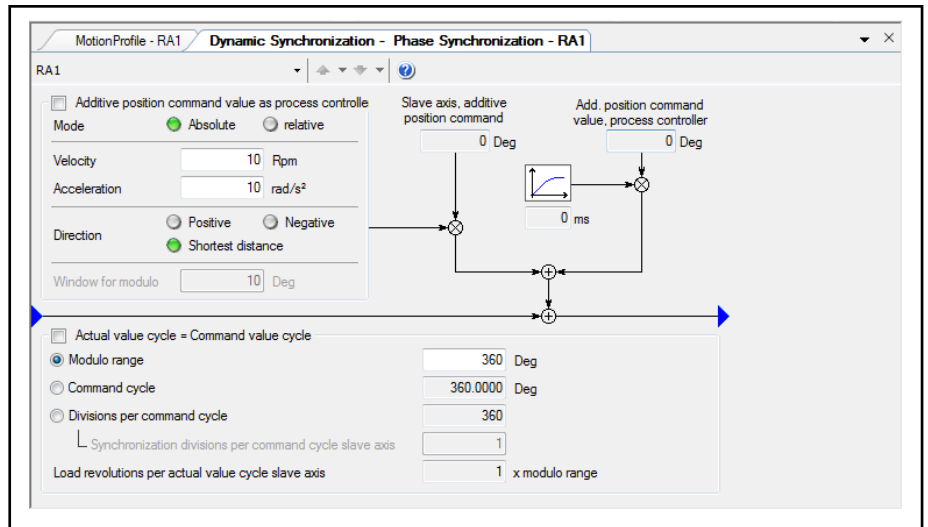


Fig. 6-57: "Dynamic Synchronization - Phase Synchronization" dialog in the "MotionProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		
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			

6.5.22 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 switch-

Axes

ing 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 common application of the FlexProfile is described in "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Libraries" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

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".

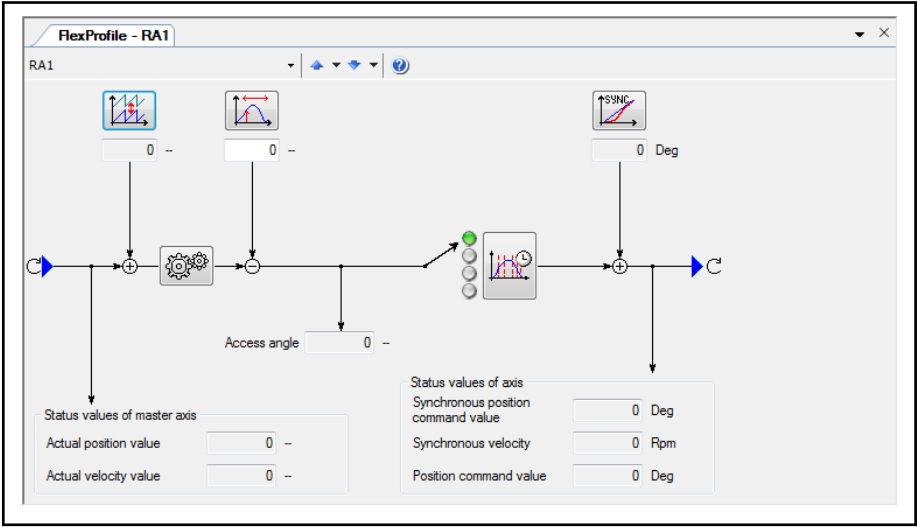


Fig. 6-58: "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
Scaling type for position data			A-0-0059		
ELS, master axis, axis number			A-0-2701		
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 velocity" as well as "A-0-2620, ELS, phase offset start of table, process controller".
FlexProfile overview	Selection and download of a FlexProfile to a 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.

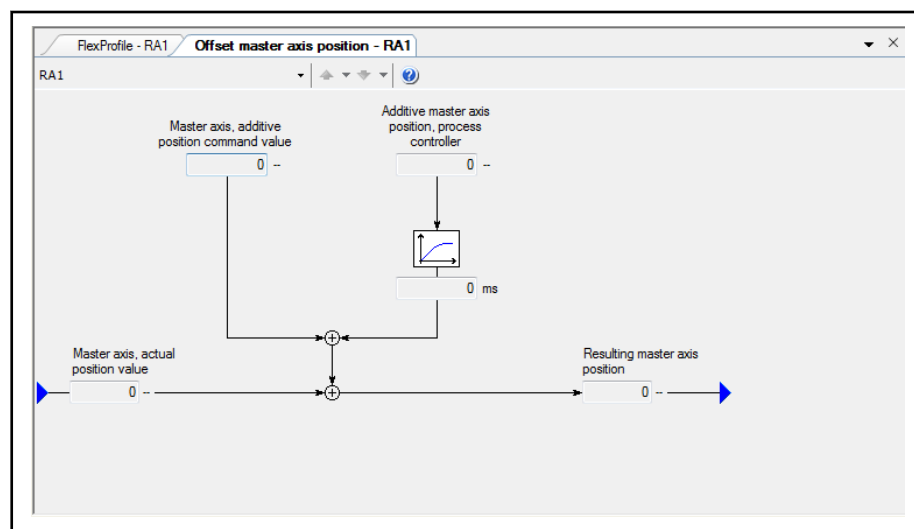


Fig. 6-59: "Offset Master Axis Position" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

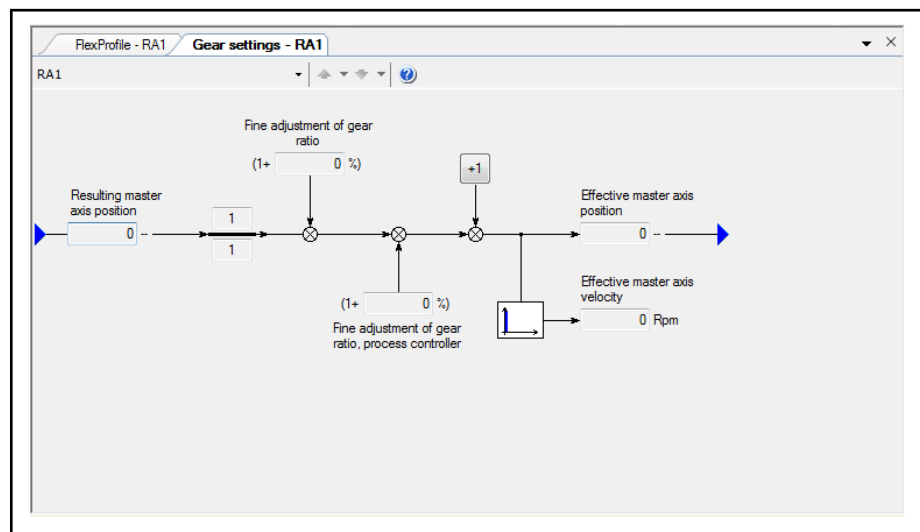


Fig. 6-60: "Gear Settings" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

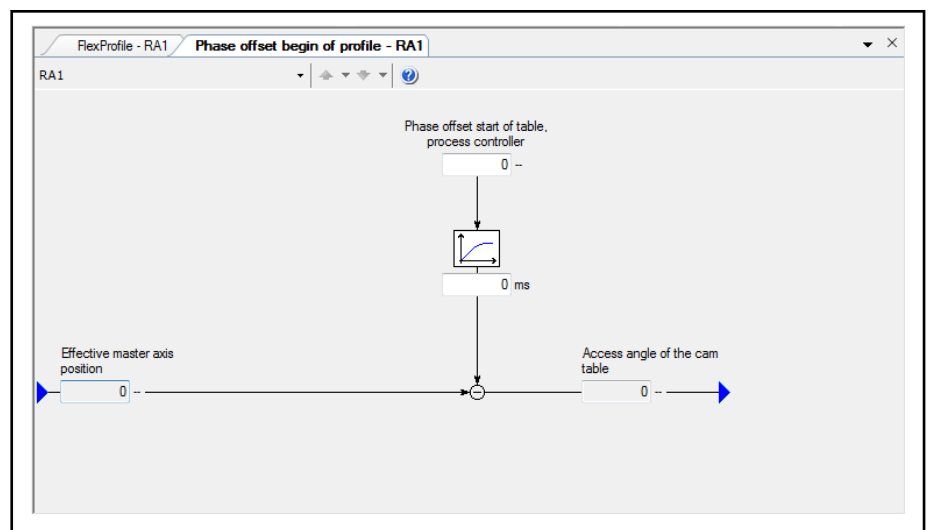


Fig. 6-61: Dialog "Phase offset begin of profile" in the "FlexProfile"

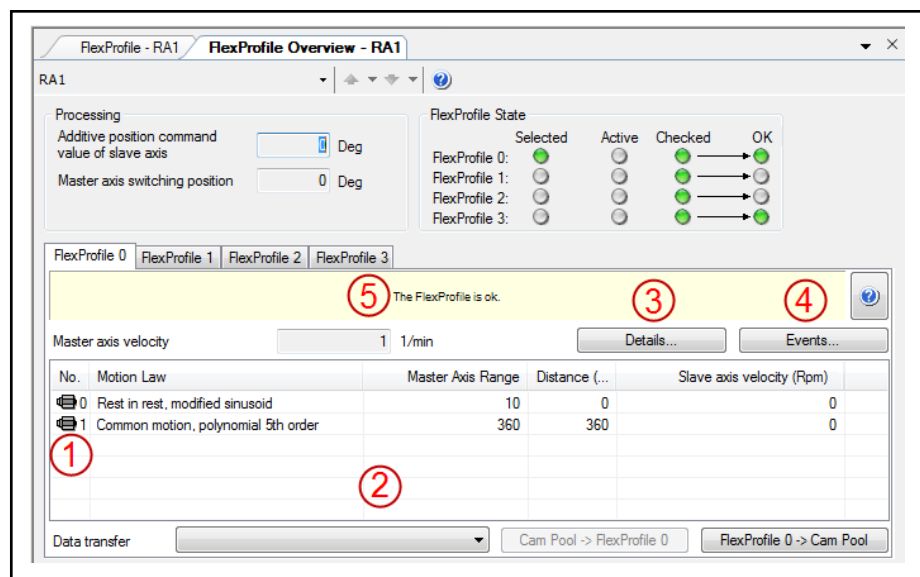
Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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

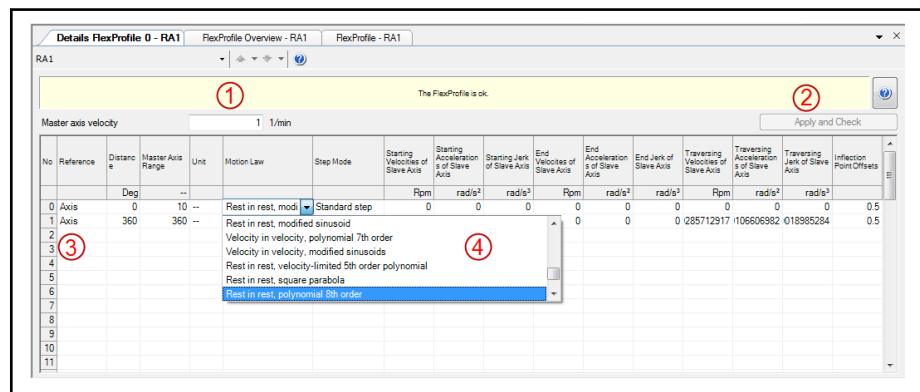
Selection and download of a FlexProfile to the axis. For each axis, up to four FlexProfiles can be stored. The FlexProfile to be processed is selected in the ML_FlexProfile Motion function block.

Axes



- ① Symbolic representation of the segment type (axis or time)
- ② Configuration display
- ③ FlexProfile configurator
- ④ Event configurator
- ⑤ Status display

Fig. 6-62: Dialog "FlexProfile overview" in "FlexProfile" dialog



- 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. 6-63: Configuration of FlexProfile

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		
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".

Axes

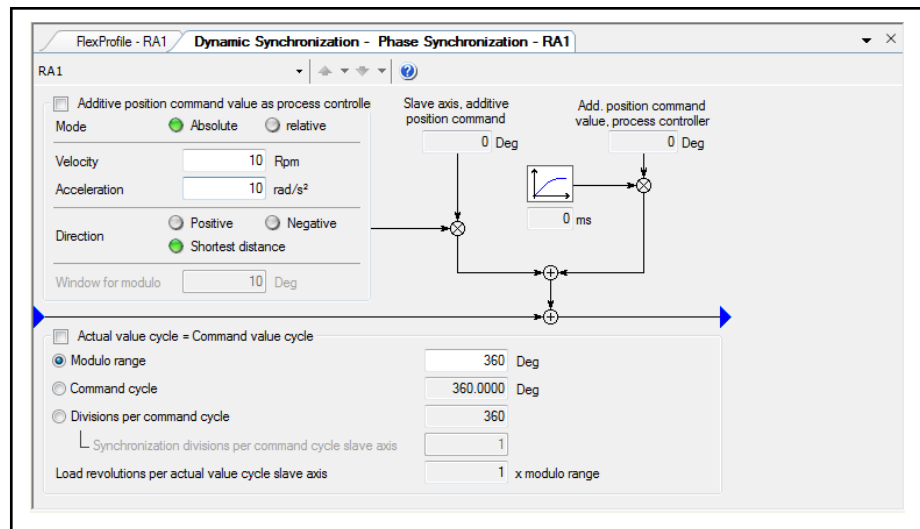


Fig. 6-64: "Dynamic Synchronization – Phase Synchronization" dialog in the "FlexProfile" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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		
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			

6.5.23 Gear with velocity synchronization

General information



This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual 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".

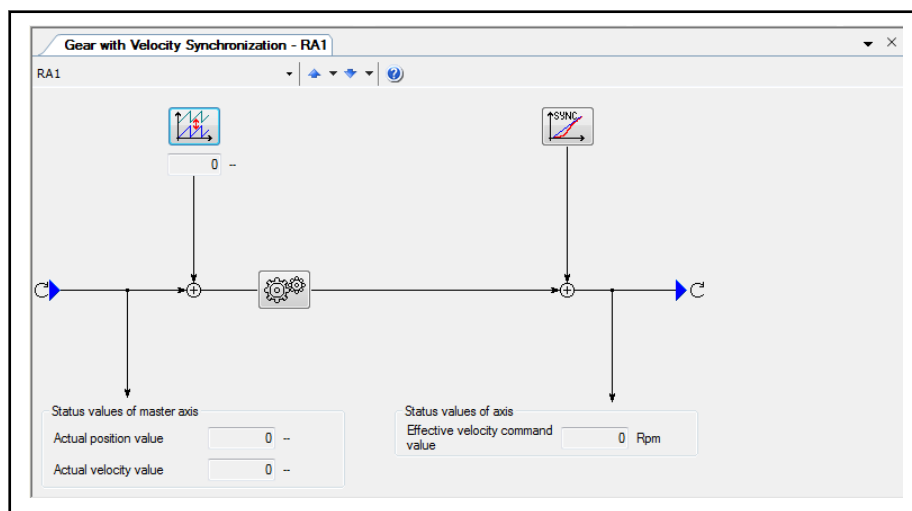


Fig. 6-65: "Gear with Velocity Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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			
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 MC_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

Axes

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".

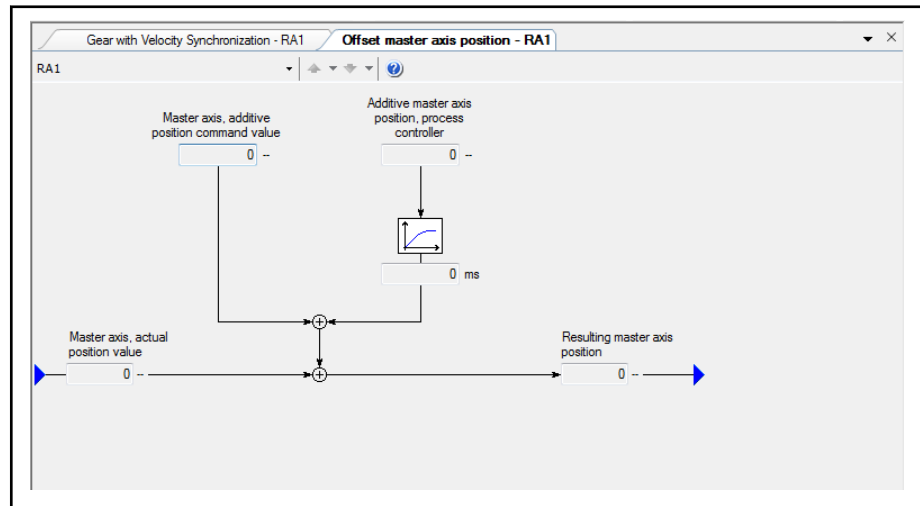


Fig. 6-66: Dialog "Offset master axis position" in the dialog "Gear with Velocity Synchronization"

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

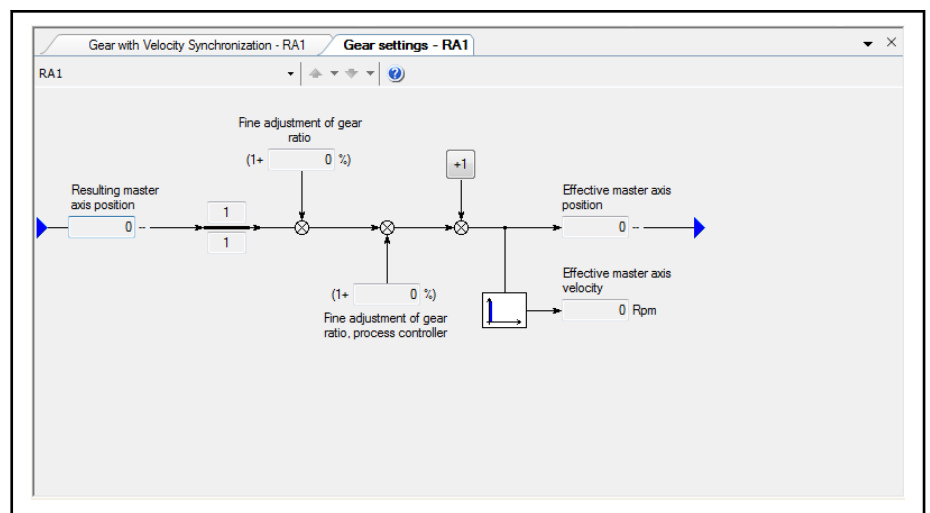


Fig. 6-67: "Gear Settings" dialog in the "Gear with Velocity Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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"

Axes

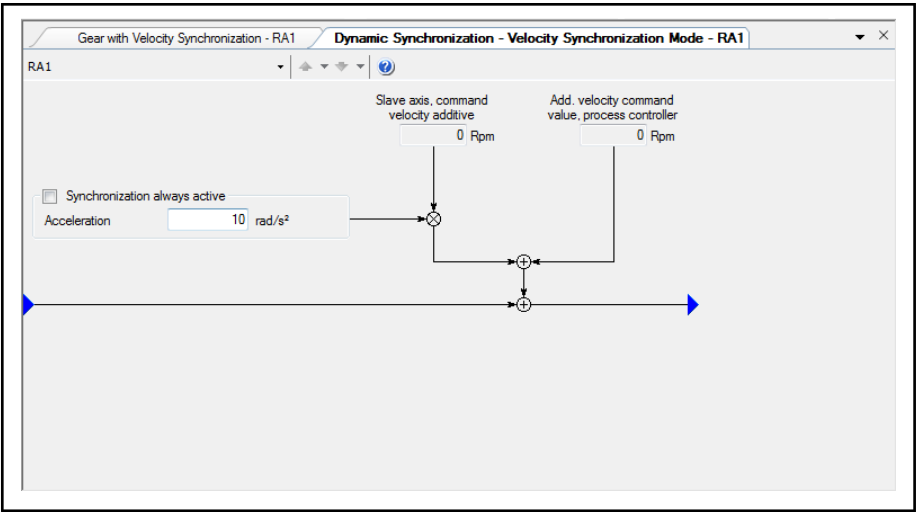


Fig. 6-68: *Dialog "Dynamic Synchronization" in the dialog "Gear with Velocity Synchronization"*

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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, process controller	A-0-2615				

6.5.24 Gear with phase synchronization

General information

 This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual axis.

If a real axis operates as slave axis in the MB_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".

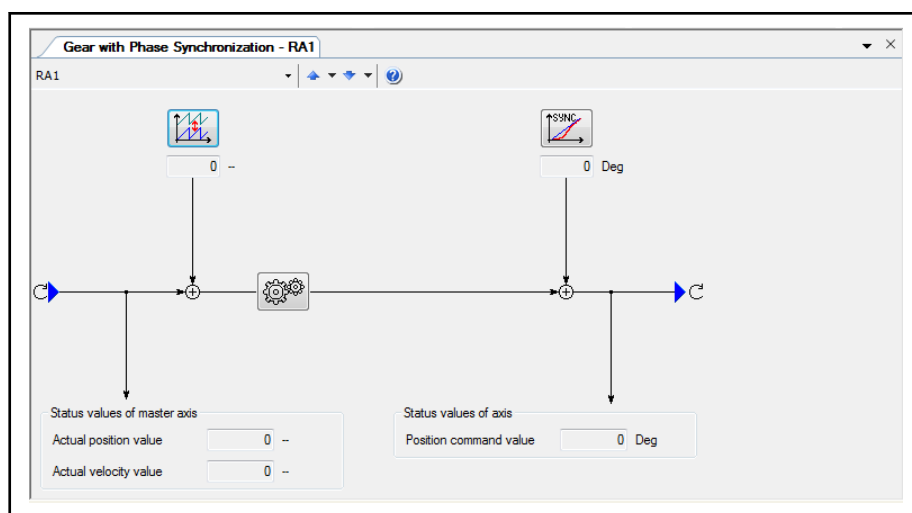


Fig. 6-69: "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

Axes

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.

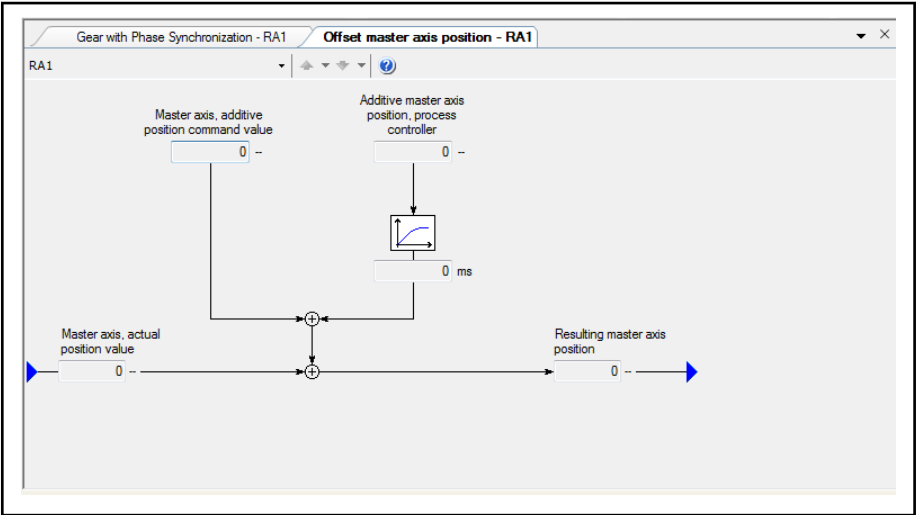


Fig. 6-70: Dialog "Offset master axis position" in the dialog "Gear with Phase Synchronization"

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

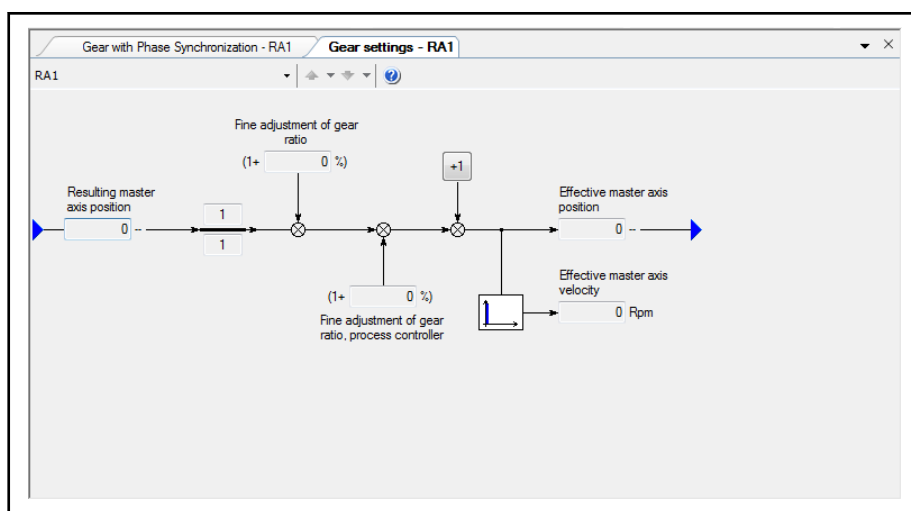


Fig. 6-71: "Gear Settings" dialog in the "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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".

Axes

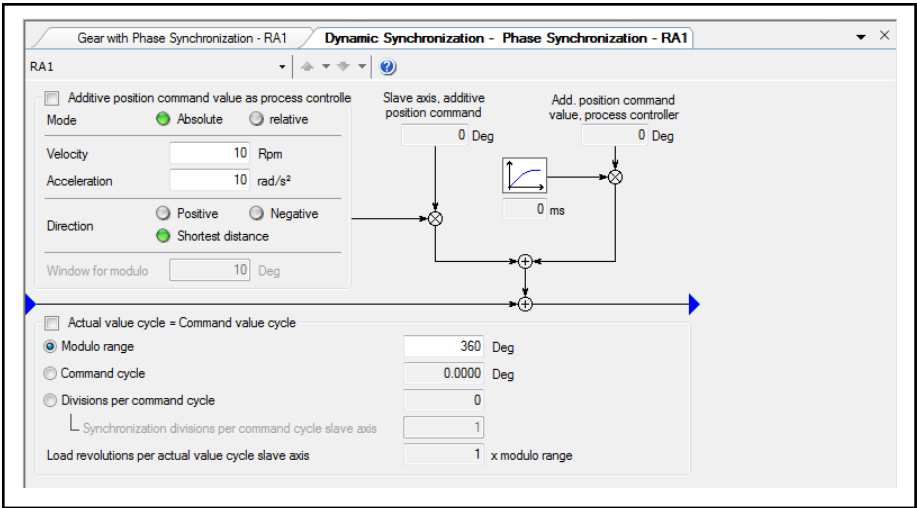


Fig. 6-72: "Dynamic synchronization" in the "Gear with Phase Synchronization" dialog

Parameters used:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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				

6.5.25 Axis configuration

 This dialog is available for encoder and link axes.

The encoder axis and link axis are configured via the following dialog.

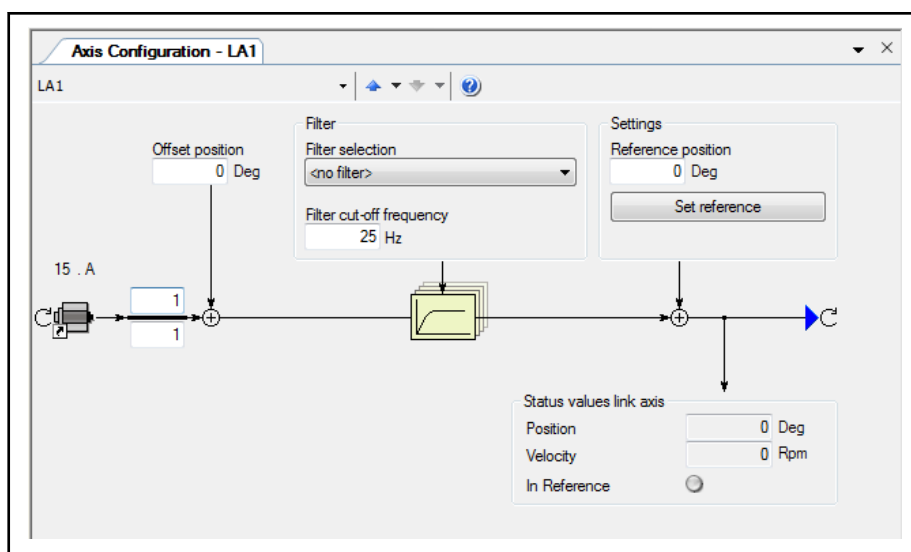


Fig. 6-73: Dialog "Axis Configuration"

The following parameters are analyzed:

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
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 encoder					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".

Axes

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

Tab. 6-13: Filter type specification

The parameter "A-0-2805, Master axis encoder, filter cutoff frequency" to store the filter cutoff frequency completes the component.

In the end, the position value can be set to the reference position via 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

6.5.26 Drive-integrated safety engineering



This dialog is available for IndraDrive for interpolation in the drive and in the control.

This dialog is only available if the drive is provided with the option module "Safety" or "S4" or "S5". The 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 [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.27 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.

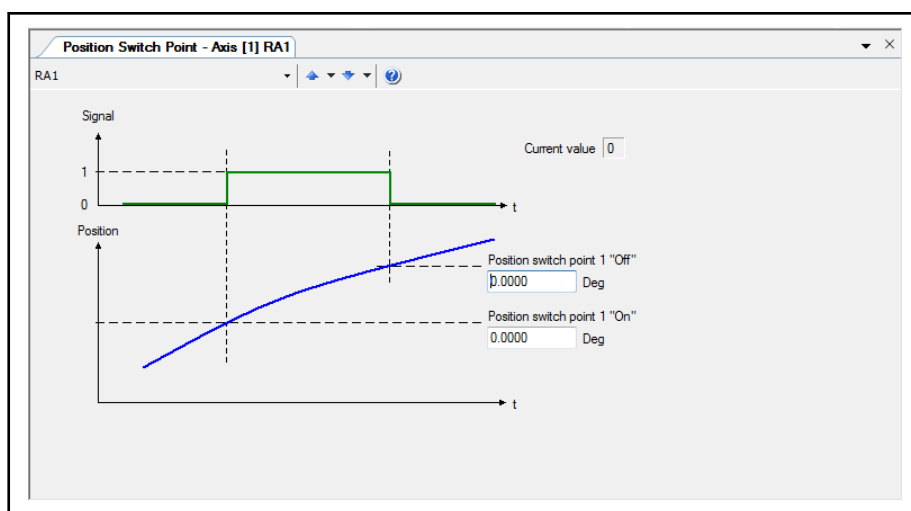


Fig. 6-74: "Position Switch Point" dialog

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 status "1"; outside the two positions, the virtual switch is in status "0".

The value can be read from the parameter "S-0-0059, Position switching point parameter".

The basics of the "position switching points" of the "dynamic programmable limit switch" are described with the corresponding products "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Drive-integrated command value generators" (see [chapter 1.4 "Required and supplementing documents" on page 18](#)).

Support is provided by the implemented online help.

6.5.28 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.

Axes

A measurement can be triggered by positive and/or negative touch probe signal edges.

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).

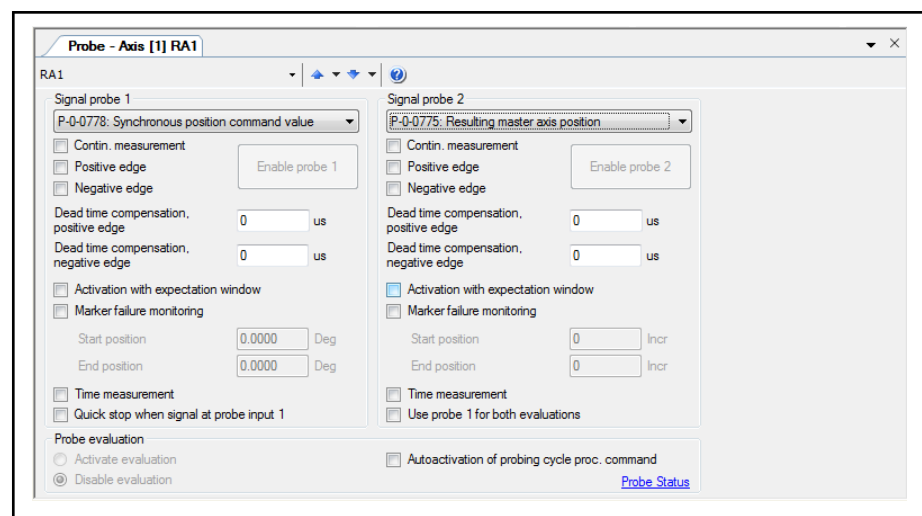


Fig. 6-75: Dialog "Touch Probe"

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	
Starting position active	P-0-0200	

Setting	IndraDrive (interpolation in the drive)	IndraDrive (interpolation in the control)
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.

When disabling, the command in the drive is deleted and, at the same time, the command execution is interrupted.



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.

Enabling 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 range of an axis or shaft within which probe signal edges lead to measured values.

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 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 run (exceeding 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 can be displayed. 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.

Axes

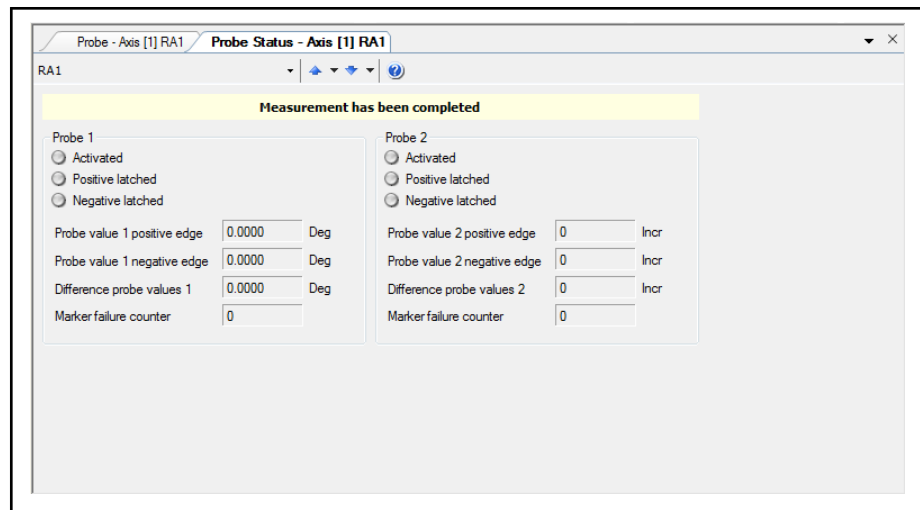


Fig. 6-76: "Touch Probe Status" dialog in the "Touch Probe" dialog

For more touch probe function basics, refer to the corresponding products "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Touch probe function" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Support is provided by the implemented online help.

6.5.29 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)	SercosDrive	VA	EncA
chapter "Loading basic parameters" on page 297	x	x	x	x	x
chapter "Settings, Motor Encoder" on page 242	x	x	x		
chapter "Settings of Optional Encoder" on page 244	x	x	x		x
chapter 6.5.7 "Settings: scaling/units" on page 245	x	x	x	x	
chapter 6.5.8 "Scaling/units extended" on page 246	x	x	x		
chapter 6.5.9 "Mechanical gear" on page 248	x	x	x		
chapter 6.5.5 "Brake" on page 242	x	x	x		
chapter 6.5.16 "Motion limit values" on page 257	x	x	x		
chapter 6.5.17 "Initial values" on page 259	x	x	x	x	
chapter 6.5.15 "E-STOP function" on page 256	x	x	x		
chapter 6.5.31 "Communication" on page 299	x	x	x		

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	SercosDrive	VA	EncA
chapter 6.5.11 "Settings of status messages" on page 250	x	x	x	x	
chapter 6.5.25 "Axis configuration" on page 290					x



Data changed while editing in the dialog is directly applied to the control/drive, irrespective of whether a subsequent date is still changed or the dialog is exited with **Next** or **Cancel**.

6.5.30 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
 - Load the basic parameters for the drive via the context menu items "Setup" or "Parameters"⇒"Load Basic Parameters..."
 - A-0-1000, command: Load basic parameters of the axis: Sets the required A-parameters for the drive to default
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
 - Load the basic parameters for the drive via the context menu items "Setup" or "Parameters"⇒"Load Basic Parameters..."
 - 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 a new axis (axis was deleted before)



This procedure is not recommended!

- Dragging drive to the Project Explorer

Axes

- Going online
 - "Load basic parameter" is not executed
- All old parameter assignments are applied

Drive password - Change password

IndraDrive control units allow to protect parameter values from unintended or unauthorized password change.

Based on a default or known password, this menu item allows to enter a new changed customer password and to confirm it with the second input.

Regarding the write protection, there are three groups of writable parameters:

- The parameters that are always write-protected, such as motor parameters, hardware identification parameters, encoder parameters, error memory, etc. ("Management 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 used to adapt the drive to the axis after their definition
- All remaining writable parameters which are not included in one of the mentioned groupings. The parameters are not write-protected

Password types:

The drive firmware allows to activate or deactivate the write protection for parameter values by 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 basics about the "loading, saving and backing up parameters" are described product-specifically in the manuals "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" or "Rexroth IndraDrive MPx 17 Functions", chapter "Loading, saving and backing up parameters" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Support is provided by the implemented online help.

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 basics about the "loading, saving and backing up parameters" are described product-specifically in the manuals "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" or "Rexroth IndraDrive MPx 17 Functions", chapter "Loading, saving and backing up parameters" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

6.5.31 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 in MDT. The AT is the response telegram. Actual values are received in AT. 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.

Fig. 6-77: "Cyclic Sercos Data Channel" dialog

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":

Axes

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 $\Rightarrow$ axis 1, MDT data container B, interpreted as an INT.

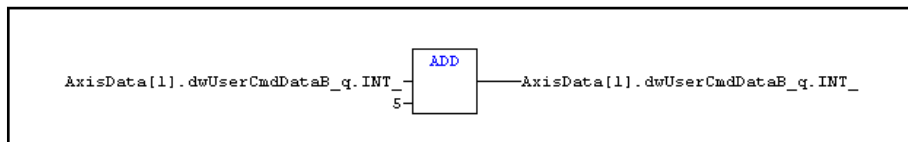


Fig. 6-78: Using the cyclic Sercos data channel for writing

Another option from the PLC program is the use of the function blocks and functions for the cyclic data exchange (MB_GetCyclicParameterHandle, MB_ReadCyclicParameter, MB_ReadCyclicRealParameter of the PLC library ML_TechBase).

The following parameters are used to exchange data:

Write access

- 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

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 μ s) and is therefore to be avoided!

Signal status word



This dialog is available for IndraDrive for interpolation in the drive and in the control.

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 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.

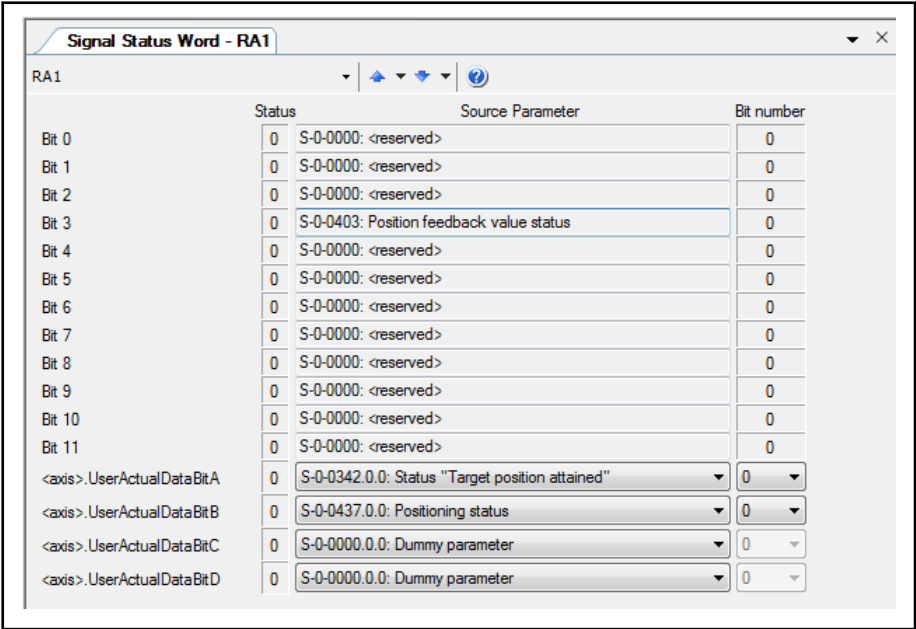


Fig. 6-79: "Signal Status Word" dialog

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 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⇒ axis 1, data container A, WORD.



In order to receive a Boolean value, the functional (WORD) variable has to be masked with 0x0001:

Axes

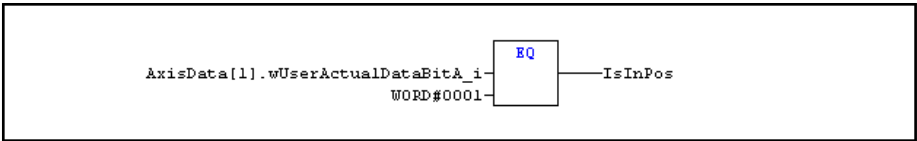


Fig. 6-80: Using the signal status word for reading

Another option from the PLC program is the use of the function blocks and functions for the cyclic data exchange (MB_GetCyclicParameterHandle, MB_ReadCyclicParameter, MB_ReadCyclicRealParameter of the PLC library ML_TechBase).

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

Tab. 6-14: 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.

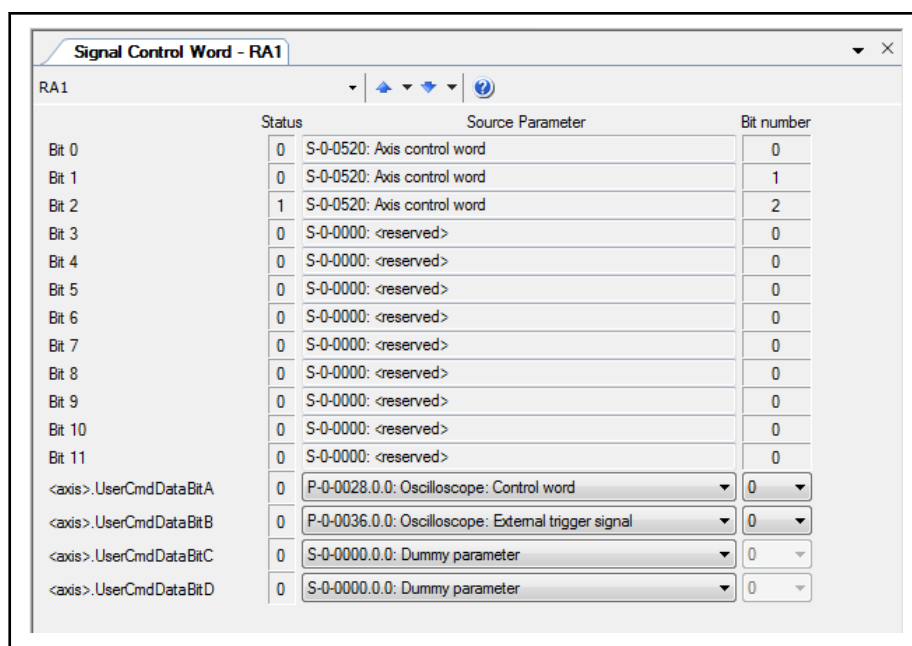


Fig. 6-81: "Signal Control Word" dialog

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 ⇒ 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):

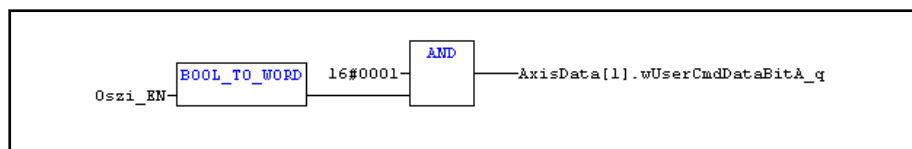


Fig. 6-82: Using the signal control word for writing

Another option from the PLC program is the use of the function blocks and functions for the cyclic data exchange (MB_GetCyclicParameterHandle, MB_ReadCyclicParameter, MB_ReadCyclicRealParameter of the PLC library ML_TechBase).

Axes

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

Tab. 6-15: 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 µs) and is therefore to be avoided!

6.5.32 Diagnostics
Status



This dialog is available for all axis types.

This dialog shows the status information on the respective axis. In the given example, this is a real axis in an endless motion. The window is split in two, the upper half contains general information, whereas the lower half contains detailed supplements.

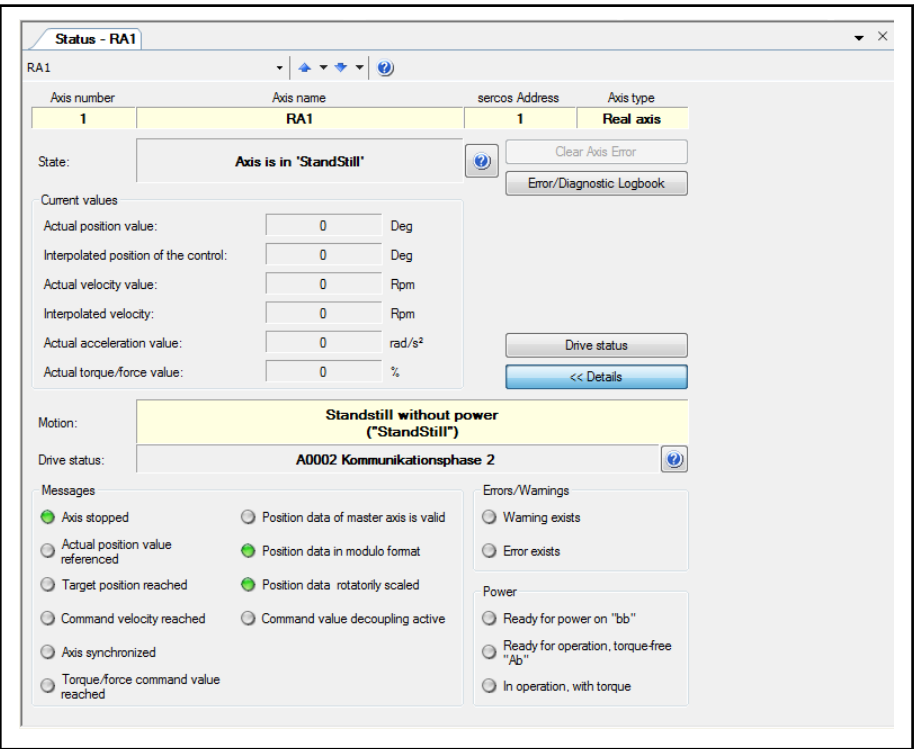


Fig. 6-83: "Status" dialog – Axis status of a real axis

To identify the axis:

Axes

Setting	IndraDrive (IPO in the drive)	IndraDrive (IPO in the control)	Sercos- Drive	VA	EncA
Axis ID	x				
Logic axis number	A-0-0001				
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				
Drive address, here: Sercos address of the drive	A-0-0005				A-0-0005
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).



Double-click on the axis in the "Device status" of the control to open the "Axis status" dialog.

IDN list of invalid operating data



This dialog is available for IndraDrive for interpolation in the drive and in the control.

Axes

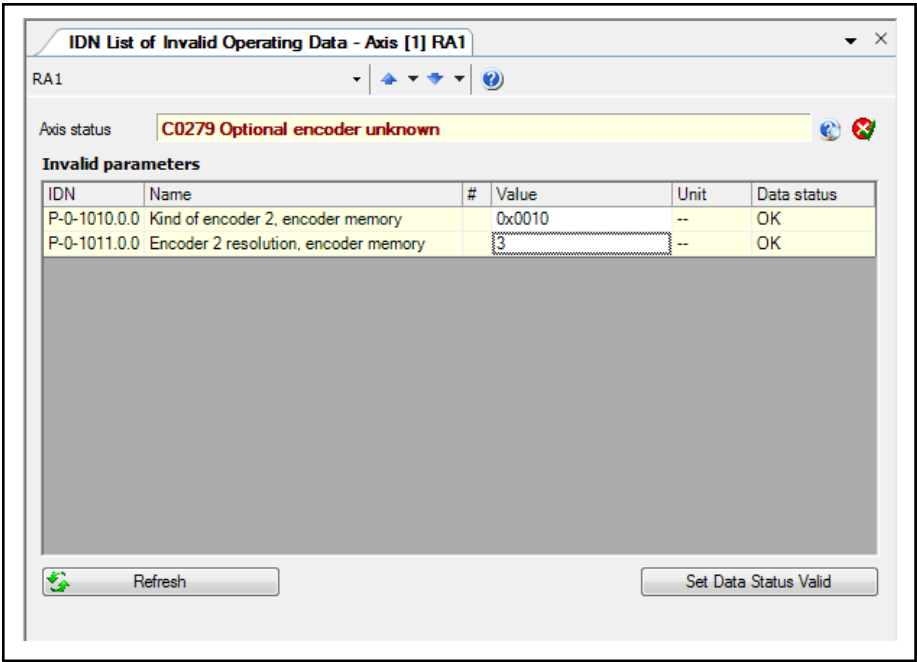


Fig. 6-84: "IDN List of Invalid Operating Data" dialog

When switching to phase 4, the drive parameters are checked and converted. If errors occur, the incorrect parameters are labeled as "configuration invalid or not permitted". The detected parameters are then listed in the dialog "IDN list of invalid operating data".

Using this dialog, the data status of parameters identified as invalid can be set to valid or the parameter value can be corrected.

Clearing Errors

 This dialog is available for all axis types.

Enabling the context menu item, triggers "A-0-1030, Command: Clear all axis errors" and deletes the axis error in the drive and on the control.

Error/diagnostic memory

 This dialog is available for all commanding axis types. These are: IndraDrive, SercosDrive and virtual axis.

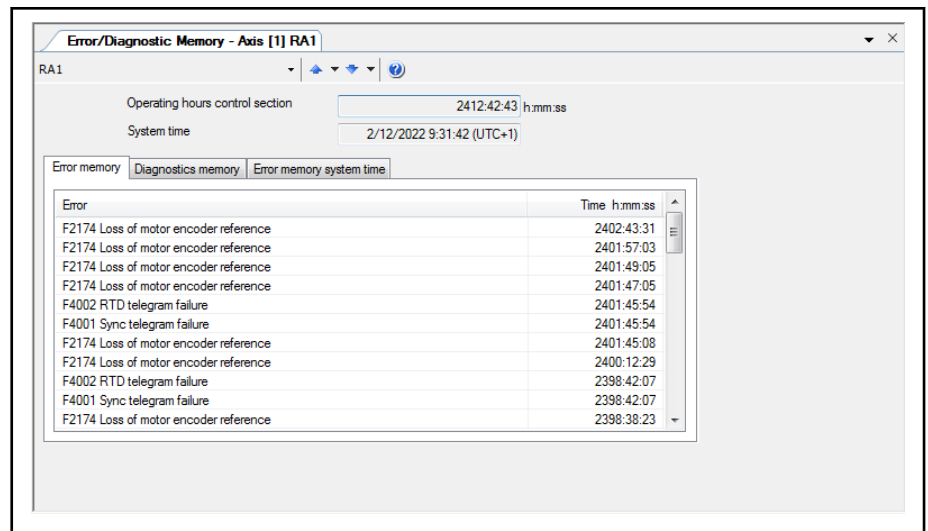


Fig. 6-85: "Error/Diagnostic Memory" dialog

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.

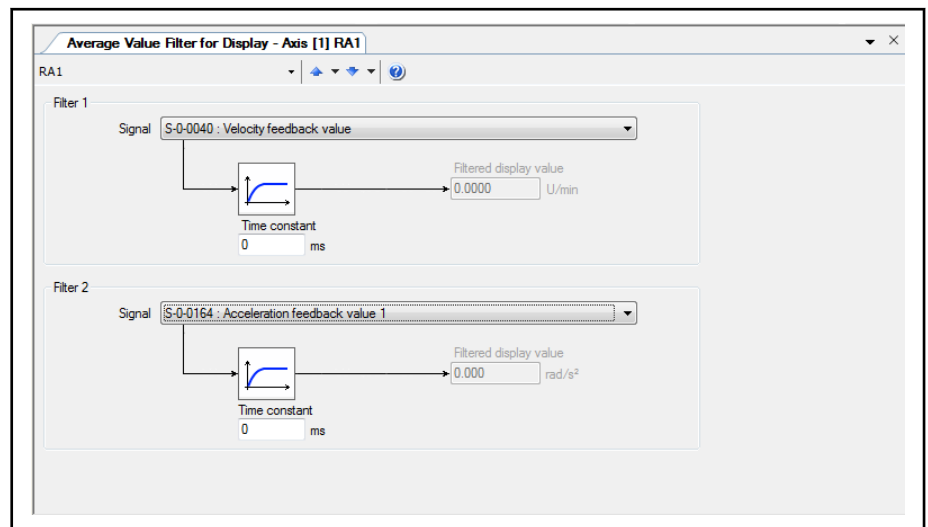


Fig. 6-86: "Average Value Filter for Display" dialog

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.

Axes



Note: 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 basics are described with the corresponding products "Rexroth IndraDrive Firmware for Drive Controllers MPH-, MPB-, MPD-, MPC-07" and "Rexroth IndraDrive MPx-17 Functions", chapter "Velocity controller (with corresponding filters)" (see [chapter 1.4 "Required and supplementing documents" on page 18](#)).

Support is provided by the implemented online help.

6.6 Drive as PLC device

6.6.1 General information

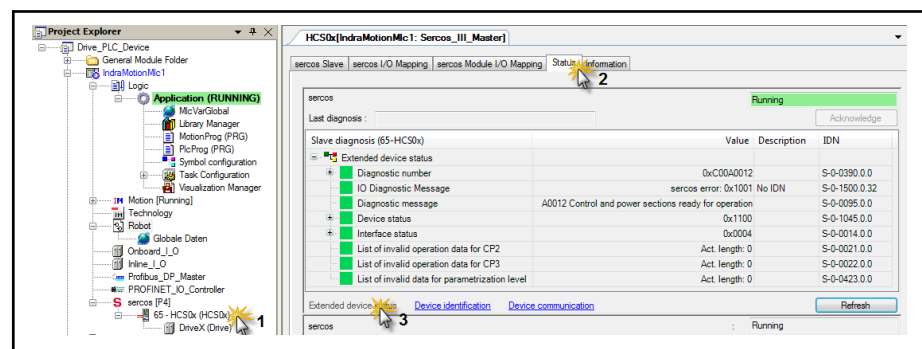
In contrast to an axis, a drive as PLC device has no motion functionality and no A-parameters. The drive is PLC-commanded and freely programmable.

The drive is created in the IndraWorks project by dragging the device from the library window ("Periphery" folder, folder for the respective field bus) to the "Field bus" node.

For Sercos, an IndraDrive single-axis device for MPx17VRS and higher is provided as HCS0x in the peripherals "Sercos". IndraDrive double-axis devices as HMD01 (being prepared, final library name not yet defined).

Drives of any manufacturers can be subsequently installed in the device database.

To configure the cyclic communication and diagnostics of the PLC devices, IndraWorks dialogs are provided.



- 1 Double-click on the Sercos device
- 2 Select "Status" tab
- 3 Click on "Extended device status"

Fig. 6-87: Example: Diagnostic dialogs on the Sercos PLC devices

For IndraDrive, configure the communication and parameterization of the drives via IndraWorks Ds. For other drives, use the manufacturer-specific tool.

6.6.2 Sercos II drive as PLC device

Sercos III drive as PLC device – Overview

The implementation of the following motion sequence is exemplarily shown in the following:

- Positioning to an absolute position
- Relative positioning (target position = current position + position specification)
- Additive positioning (target position = last target position + position specification)
- Velocity control
- Drive Halt (AH)

Any number of motion sequences and drive-related functions can be implemented. In this example, only a small part of the IndraDrive functionality is used.

The drive commissioning steps can be followed using this section. At the end, it should be possible to move the drive with the described motion sequence using the listing (see [chapter "Sercos III drive as PLC device – PLC programming" on page 314](#)). The required steps are described at the end of the respective section.



The usage as real axis has many advantages compared to the usage as PLC device with free programming described here. For usage as PLC device, advanced knowledge on the Sercos parameters used and additional programming effort is required.



The RMB_PLCOpenFieldbus library cannot be used for Sercos drives. When using PLCopen function blocks for programming, the drive has to be operated as real axis (e.g. as SercosDrive). Thus, the Motion functionalities and dialogs to parameterize in IndraWorks are available.

Creating the project

IndraWorks is installed. An IndraLogic XLC control is available. A Sercos drive is available. The corresponding device description file is installed.

1. Creating an IndraWorks project with suitable control.
2. Dragging drive from the library window (folder "Periphery", folder "Sercos") to the "Sercos" node.
3. Switching the control online, synchronizing it (download Motion configuration on the device) and switching it to P2.
4. Right-click on **Control ► Communication ► Interfaces**.

Noting down IP address of the bridge for the communications settings for the parameterization.

5. Right-click on **Drive ► Parameter Editor**.

Note down the IP address of the drive from S-0-1020 for the communications settings for the parameterization.

Sercos II drive as PLC device – Parameterization

For IndraDrive, parameterize the drives via IndraWorks Ds. For other drives, use the manufacturer-specific tool.

Axes

It has to be communicated directly via PLC and drive supply. The PC can be connected to a free Sercos port of the drive or communicate via a "Sercos III NetSwitch" (ordering no.: R911328254 HAWA HGS NS-S3-1NRT). The Windows command "route" can also route the IP communication via the control.

The IP addresses are preset by the control. The presetting depends on the control address C-0-0031 and the Sercos address if the Sercos device is provided with the corresponding parameters (class SCP_NRT or SCP_NRTPC in S-0-1000).

- Default address of the Sercos device if C-0-0031=0: 172.31.254.<SercosAddress>. Otherwise, 172.31.<C-0-0031>.<SercosAddress>
- Default address of the control via a Sercos port if C-0-0031=0: 172.31.254.254. Otherwise, 172.31.<C-0-0031>.254

For information on the IP address specification, refer to [tab. 3-4 "Automatically generated IP address and subnet mask" on page 70](#).



Instead of using IndraWorks Ds, an IndraDrive can also be dragged from the library to the main node (parallel to the control) in the IndraWorks project of the control. A previously created parameter file using IndraWorks Ds is useful for the configuration.

The cyclically applied parameters are configured at Sercos and - in contrast to other field buses - below the "Sercos" node in IndraWorks.

To implement the motion sequences given in [chapter "Sercos III drive as PLC device – Overview" on page 309](#), the following parameterization has to be made at least using IndraWorks Ds:

- Primary operation mode "Drive-controlled positioning"
- 1. Secondary mode "Velocity control"

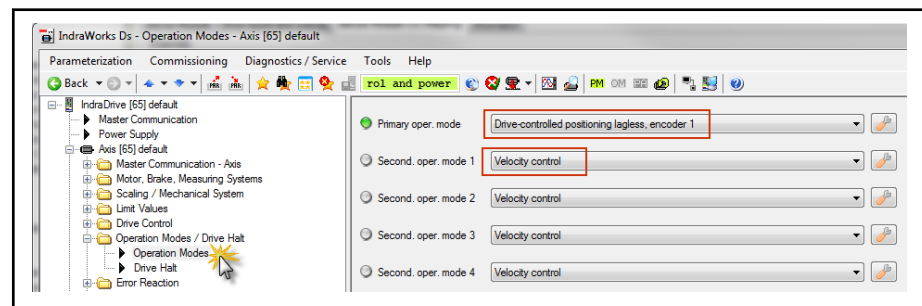


Fig. 6-88: Parameterizing operation modes using IndraWorks Ds

- Signal control word
 - Bit 0: S-0-0346, bit 0 "Applying position command value via toggling"
 - Bit 1: S-0-0346, bit 3 "Type of position command value, 0:: absolute, 1: relative (depends on bit 4)"
 - Bit 2: S-0-0346, bit 4 "Reference point for position command values, 0: Latest effective target position (S-0-0430), 1: Active actual position value (S-0-0386)"
- Signal status word:
 - Bit 0: S-0-0331, bit 0 "Message n_act = 0"
 - Bit 1: S-0-0342, bit 0 "Target position reached"
 - Bit 2: S-0-0330, bit 0 "Message n_act = n_com"

- Bit 3: S-0-0403, bit 0 "Status of the reference encoder "actual position value" (encoder 1 or 2), 0: relative, 1: referenced"

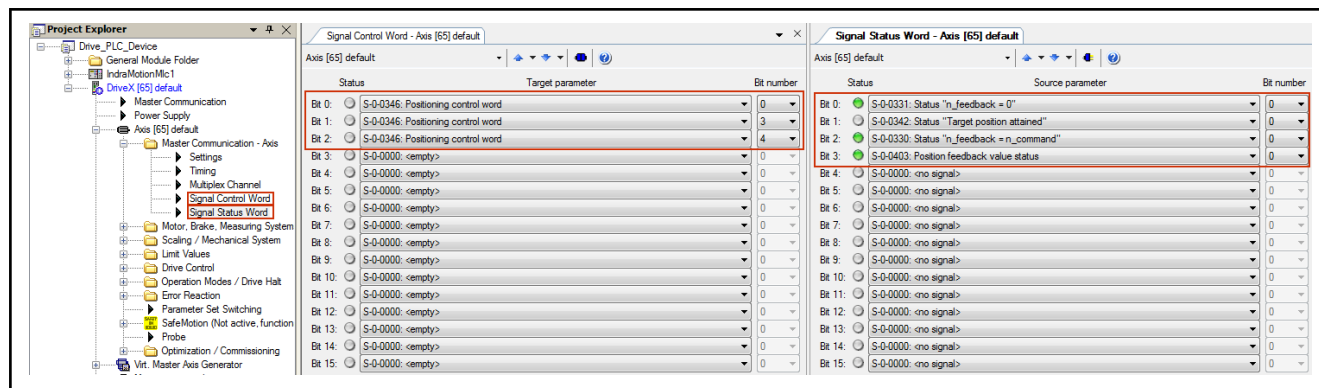


Fig. 6-89: Parameterizing signal control and signal status words using IndraWorks

Also make further parameterization of the drive, e.g. motion limit values.



To read or write parameters of a PLC device from the PLC program, use function blocks from the RIL_SERCOSIII, folder AcyclicCommunication.

Parameterizing the drive

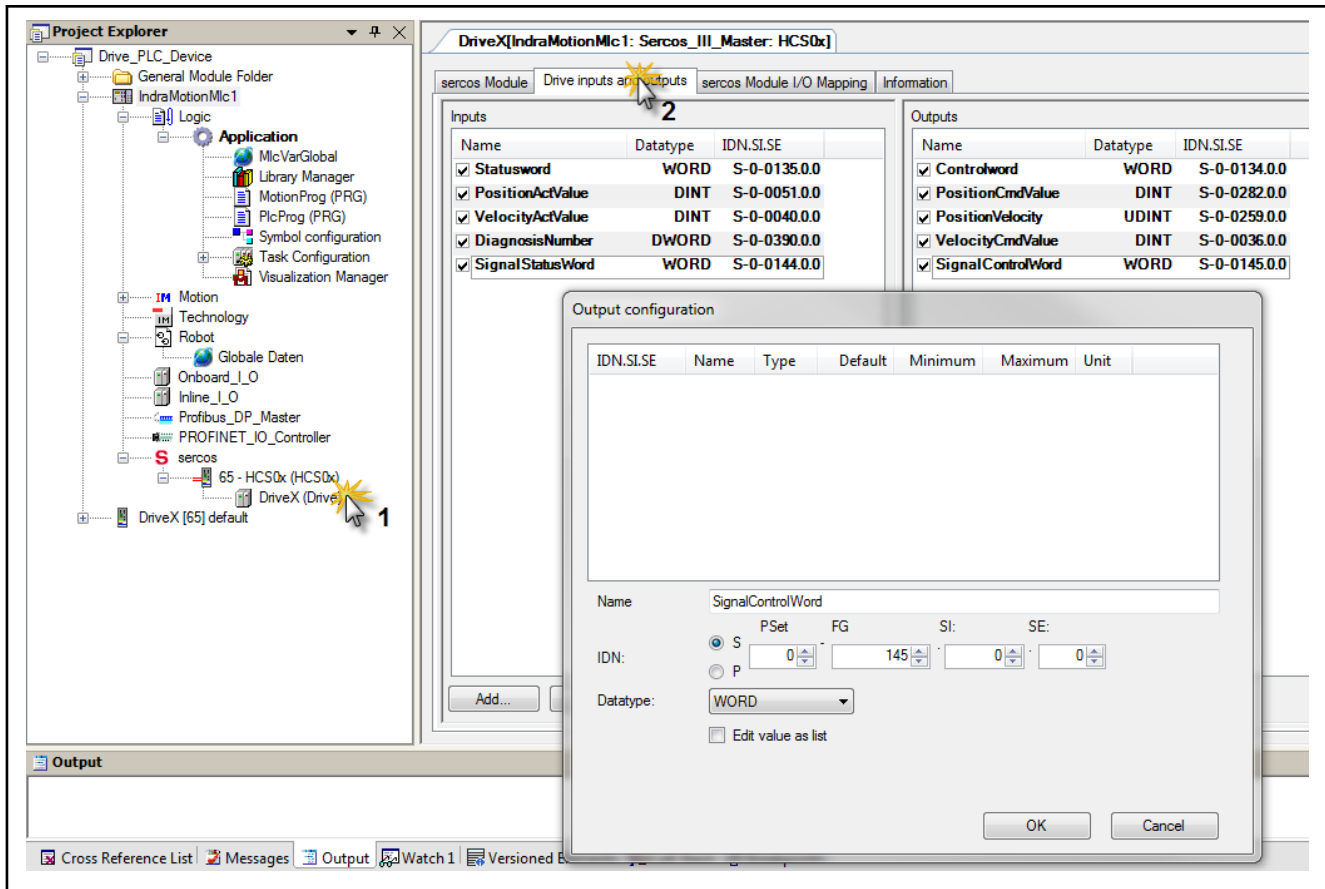
The PC is connected to an open Sercos port or via a "Sercos III NetSwitch". The IP address of the PC is configured with an address matching the control and drive IP address.

1. Enter the bridge address noted down in "Creating the project" on page 309, point 4, into the IndraWorks project as IP address of the control. Switching online should be possible again.
2. Start IndraWorks Ds. Select connection -> IP address search. Enter and search IP address of the drive (in "Creating the project" on page 309, point 5, noted down). Connect to drive.
3. Parameterizing operation modes in IndraWorks Ds as described in fig. 6-88 "Parameterizing operation modes using IndraWorks Ds" on page 310.
4. Parameterizing signal control word and signal status word in IndraWorks Ds as described in fig. 6-89 "Parameterizing signal control and signal status words using IndraWorks" on page 311.
5. Further parameterization of the drive is to be executed in IndraWorks Ds. Motion limit values, rapid stop deceleration, controller parameters, status messages, etc.

Sercos III drive as PLC device – Configuring cyclic data

Cyclically applied data is configured in IndraWorks below the "Sercos" node in the dialog "General Inputs and Outputs".

Axes



- 1 Double-click on the "Drive" node below the Sercos device
- 2 Select driver inputs and outputs

Fig. 6-90: Example configuration of cyclic parameters in the drive diagram AT (inputs) and master data telegram MDT (outputs)

In this example, the parameters required for the operation modes "Positioning" and "Velocity control" are configured.

The entered name is subsequently output in the "Channel" column in [fig. 6-91 "Assigning PLC variables to the configured parameters" on page 313](#).

Select the correct data type with the suitable length and without or with sign depending on the attribute.

Data length in the attribute	Format in the attribute	Data type in the PLC
2 bytes	BIN or HEX	WORD
4 bytes	BIN or HEX	DWORD
2 bytes	DEC_MV	INT
4 bytes	DEC_MV	DINT
2 bytes	DEC_OV	UINT
4 bytes	DEC_OV	UDINT

Tab. 6-16: Assignment: Parameter attribute <> Data type in the PLC

The parameters are assigned to the PLC variables in the "Sercos Module I/O Mapping" tab. New variables can be created or the input and output parameters can be mapped on existing PLC variables.

Variables are created in the following.

DriveX[IndraMotionMlc1: Sercos_III_Master: HCS0x]							
sercos Module		Drive inputs and outputs		sercos Module I/O Mapping		Information	
Channels							
Variable	Mapping	Channel	Address	Type	Default Value	Unit	Description
wDriveX_ControlWord		Controlword	%QW4	WORD			
diDriveX_PositionCmd...		PositionCmdValue	%QD8	DINT			
udiDriveX_PositionVel...		PositionVelocity	%QD12	UDINT			
diDriveX_VelocityCmd...		VelocityCmdValue	%QD16	DINT			
		SignalControlWord	%QW20	WORD			
boDriveX_PosTog...		Bit0	%QX20.0	BOOL	FALSE		
boDriveX_PosRelN...		Bit1	%QX20.1	BOOL	FALSE		
boDriveX_PosRelN...		Bit2	%QX20.2	BOOL	FALSE		
		Bit3	%QX20.3	BOOL	FALSE		
		Bit4	%QX20.4	BOOL	FALSE		
		Bit5	%QX20.5	BOOL	FALSE		
		Bit6	%QX20.6	BOOL	FALSE		
		Bit7	%QX20.7	BOOL	FALSE		
		Bit8	%QX21.0	BOOL	FALSE		
		Bit9	%QX21.1	BOOL	FALSE		
		Bit10	%QX21.2	BOOL	FALSE		
		Bit11	%QX21.3	BOOL	FALSE		
		Bit12	%QX21.4	BOOL	FALSE		
		Bit13	%QX21.5	BOOL	FALSE		
		Bit14	%QX21.6	BOOL	FALSE		
		Bit15	%QX21.7	BOOL	FALSE		
wDriveX_StatusWord		Statusword	%IW4	WORD			
diDriveX_ActualPosition		PositionActValue	%ID8	DINT			
diDriveX_ActualVelocity		VelocityActValue	%ID12	DINT			
dwDriveX_DiagnosisN...		DiagnosisNumber	%ID16	DWORD			
		SignalStatusWord	%IW20	WORD			
boDriveX_StandStill		Bit0	%DX20.0	BOOL	FALSE		
boDriveX_InPosition		Bit1	%DX20.1	BOOL	FALSE		
boDriveX_InVelocity		Bit2	%DX20.2	BOOL	FALSE		
boDriveX_Homed		Bit3	%DX20.3	BOOL	FALSE		
		Bit4	%DX20.4	BOOL	FALSE		
		Bit5	%DX20.5	BOOL	FALSE		
		Bit6	%DX20.6	BOOL	FALSE		
		Bit7	%DX20.7	BOOL	FALSE		
		Bit8	%DX21.0	BOOL	FALSE		

= Create new variable

= Map to existing variable

Fig. 6-91: Assigning PLC variables to the configured parameters



The variable names of the example configuration can be copied from the listing below.

Configuring cyclic data

1. Double-click on the "Drive" node of the drive in the IndraWorks project. Select the "Driver Inputs and Outputs" tab.
2. Configure the parameter S-0-0135 "Drive status". Check for attribute in the help of S-0-0135: Length: 2 bytes, format: BIN. Select Add for the inputs and fill out the fields as follows: Field name: "Status word". Field FG: "145". Data type: Select "WORD".
3. Configure the parameter S-0-0051 "Actual position value, encoder 1". Check for attribute in the help of S-0-0051: Length: 4 bytes, format: DEC_MV. Select Add for the inputs and fill out the fields as follows: Field name: "PositionActValue". Field FG: "51". Data type: Select "DINT".
4. Enter more parameters until all parameters are configured as shown in [fig. 6-90 "Example configuration of cyclic parameters in the drive diagram AT \(inputs\) and master data telegram MDT \(outputs\)" on page 312.](#)

Axes

5. Go to the "Sercos Module I/O Mapping" tab and enter the PLC variables into the "Variable" column as shown in [fig. 6-91 "Assigning PLC variables to the configured parameters" on page 313](#).
6. Loading a PLC program Switching to "P4" should be possible via IndraWorks now.

Sercos III drive as PLC device – PLC programming

The configured PLC variables of the cyclically configured parameters can now be freely connected in the PLC.

It does not have to be programmed in a specific task. Only in case of specific requirements, as for Gantry axes, the specifications have to be made in the motion-synchronous or Sercos-synchronous task.

Program example

This sequence was implemented in the following example:

- Start in MODE_OFF = AB. Set the variable "OpMode" to 20 (ABS_POS_MODE) to start the sequence
- ABS_POS_MODE = AF, absolute positioning with 4 rpm to 42 degrees
- REL_POS_MODE = AF, relative positioning with 2 rpm by 142 degrees in clockwise direction
- ADD_POS_MODE = AF, relative positioning with 3 rpm by 42 degrees in anti-clockwise direction
- VEL_MODE = AF, velocity control with 42 rpm for 5 seconds
- MODE_AH = AH, stopping with parameterized "deceleration rapid stop" S-0-0372
- End again in Mode_OFF = AB. Switch-off controller enabled when standstill is reached

By writing one of the values REL_POS_MODE, ADD_POS_MODE or VEL_MODE to the "OpMode" variable, the sequence can also be started at these positions.

PLC programming

1. Open "PlcProg" in the IndraWorks project.
2. Copy the first program code "Declaration" below the declaration part.
3. Copy the second program code "Implementation" below the implementation part. Do not copy page break or delete it subsequently.
4. Compile. load and start PLC program.
5. Adding performance bb ->Ab at the drive. Check reference (boDriveX_Homed in the PLC project).
6. Set breakpoints at the transitions to the next operation mode (comment "// set breakpoint here to check") to check the correct processing.
7. Set the "OpMode" variable to 20. This starts the sequence and the first motion to the breakpoint can be executed.
8. Restart the PLC program up to the next breakpoint and check the motion sequence. Repeat until all steps are processed.
9. Set the "OpMode" variable to 30, 40 or 50. This starts the sequence of the respective operation mode.
10. For the final program implementation, sections can be copied and used from this program.

Declaration

```
PROGRAM PlcProg
VAR
  OpMode:INT:=MODE_DRV_OFF; // selection operation mode and state machine
  fbTON: TON;
  rActPosition: REAL; // actual position as real
  rActVelocity: REAL; // actual velocity as real
  boError: BOOL; // error flag from status word
  boWarning: BOOL; // warning flag from status word
END_VAR

VAR CONSTANT
  // operation modes and state machine
  MODE_DRV_OFF :INT:=10;
  ABS_POS_MODE :INT:=20; ABS_POS_MODE_1 :INT:=21; ABS_POS_MODE_2 :INT:=22;
  REL_POS_MODE :INT:=30; REL_POS_MODE_1 :INT:=31; REL_POS_MODE_2 :INT:=32;
  ADD_POS_MODE :INT:=40; ADD_POS_MODE_1 :INT:=41; ADD_POS_MODE_2 :INT:=42;
  VEL_MODE :INT:=50; VEL_MODE_1 :INT:=51;
  MODE_DRV_AH :INT:=60;
  // control word S-0-0134
  CTRL_WORD_OFF:WORD :=16#0000;
  CTRL_WORD_AH:WORD :=16#C000;
  CTRL_AF_PRIMARY_OP_MODE:WORD :=16#E000;
  CTRL_AF_SECONDARY_OP_MODE:WORD:=16#E100;
  // status word S-0-0135
  MASK_STATUS_AF_OP_MODE:WORD :=16#C700;
  STATUS_AF_PRIMARY_OP_MODE:WORD :=16#C000;
  STATUS_AF_SECONDARY_OP_MODE:WORD:=16#C100;
  // Weighting
  WEIGHTING_FACTOR_POS: DINT:=10000; // S-0-0078 = -4
  WEIGHTING_FACTOR_VEL: DINT:=10000; // S-0-0046 = -4
END_VAR
```

Implementation

```
CASE OpMode OF
  MODE_DRV_OFF:
    // Ab, switched OFF
    wDriveX_ControlWord := CTRL_WORD_OFF; // all OFF...

  ABS_POS_MODE:
    // start absolute positioning to 42°
    boDriveX_PosRelNotAbs:=FALSE; // S-0-0346.3 0=absolute Pos, 1=relative Pos
    //boDriveX_PosRelNotAdd; // S-0-0346.4 0=additive (last target), 1=rel. (actual pos.)
    udiDriveX_PositionVelocity:=DINT_TO_UDINT(4*WEIGHTING_FACTOR_VEL); // 4 rpm * 10000
    diDriveX_PositionCmdValue:=42*WEIGHTING_FACTOR_POS; // 42 degree * 10000
    wDriveX_ControlWord := CTRL_AF_PRIMARY_OP_MODE; // AF and primary operation mode
    boDriveX_PosToggel := NOT boDriveX_PosToggel; // start positioning
    OpMode:=ABS_POS_MODE_1;

  ABS_POS_MODE_1:
    // check acknowledge power AND homed
    IF (wDriveX_StatusWord AND MASK_STATUS_AF_OP_MODE) = STATUS_AF_PRIMARY_OP_MODE
    AND boDriveX_Homed = TRUE
    THEN
      IF boDriveX_InPosition = FALSE THEN // in position can be set from last command
        OpMode:=ABS_POS_MODE_2; // next
      END_IF
    END_IF

  ABS_POS_MODE_2:
    // check in position
    IF boDriveX_InPosition THEN
      // here we should be at 42°
      OpMode := REL_POS_MODE; // set breakpoint here to check
    END_IF

  REL_POS_MODE:
    // start relative positioning 142° clockwise
    boDriveX_PosRelNotAbs:=TRUE; // S-0-0346.3 0=absolute Pos, 1=relative Pos
    boDriveX_PosRelNotAdd:=TRUE; // S-0-0346.4 0=additive (last target), 1=rel. (actual pos.)
    udiDriveX_PositionVelocity:=DINT_TO_UDINT(2*WEIGHTING_FACTOR_POS); // 2 rpm
    diDriveX_PositionCmdValue:=142*WEIGHTING_FACTOR_VEL; // 142 degree * 10000
    wDriveX_ControlWord := CTRL_AF_PRIMARY_OP_MODE; // AF and primary operation mode
    boDriveX_PosToggel := NOT boDriveX_PosToggel; // start positioning
    OpMode:=REL_POS_MODE_1;

  REL_POS_MODE_1:
    // check acknowledge power
```

Axes

```

IF (wDriveX_StatusWord AND MASK_STATUS_AF_OP_MODE) = STATUS_AF_PRIMARY_OP_MODE THEN
  IF boDriveX_InPosition = FALSE THEN // in position can be set from last command
    OpMode:=REL_POS_MODE_2; // next
  END_IF
END_IF

REL_POS_MODE_2:
  // check in position
  IF boDriveX_InPosition THEN
    // here we should be at 42°+142°=184°
    OpMode := ADD_POS_MODE; // set breakpoint here to check
  END_IF

ADD_POS_MODE:
  // start additive positioning 42° counterclockwise
  boDriveX_PosRelNotAbs:=TRUE; // S-0-0346.3 0=absolute Pos, 1=relative Pos
  boDriveX_PosRelNotAdd:=FALSE; // S-0-0346.4 0=additive (last target), 1=rel. (actual pos.)
  udiDriveX_PositionVelocity:=DINT TO UDINT(3*WEIGHTING_FACTOR_VEL); // 3 rpm
  diDriveX_PositionCmdValue:=-42*WEIGHTING_FACTOR_POS; // -42 degree * 10000
  wDriveX_ControlWord := CTRL_AF_PRIMARY_OP_MODE; // AF and primary operation mode
  boDriveX_PosToggel := NOT boDriveX_PosToggel; // start positioning
  OpMode:=ADD_POS_MODE_1;

ADD_POS_MODE_1:
  // check acknowledge power
  IF (wDriveX_StatusWord AND MASK_STATUS_AF_OP_MODE) = STATUS_AF_PRIMARY_OP_MODE THEN
    IF boDriveX_InPosition = FALSE THEN // in position can be set from last command
      OpMode:=ADD_POS_MODE_2; // next
    END_IF
  END_IF

ADD_POS_MODE_2:
  // check in position
  IF boDriveX_InPosition THEN
    // here we should be at 42°+142°-42°=142°
    OpMode := VEL_MODE; // set breakpoint here to check
  END_IF

VEL_MODE:
  // start velocity control with 42 rpm
  diDriveX_VelocityCmdValue:=42*WEIGHTING_FACTOR_VEL; // 42 rpm * 10000
  wDriveX_ControlWord := CTRL_AF_SECONDARY_OP_MODE; // AF and secondary operation mode 1
  OpMode := VEL_MODE_1;

VEL_MODE_1:
  // check acknowledge power
  IF (wDriveX_StatusWord AND MASK_STATUS_AF_OP_MODE) = STATUS_AF_SECONDARY_OP_MODE THEN
    fbTON(IN := TRUE, PT:= T#5S);
    // 5 seconds in velocity control
    IF boDriveX_InVelocity AND fbTON.Q=TRUE THEN
      // here we should rotate with 42rpm
      OpMode := MODE_DRV_AH; // set breakpoint here to check
      fbTON(IN := FALSE);
    END_IF
  END_IF

MODE_DRV_AH:
  // stop the drive untill StandStill is reached
  wDriveX_ControlWord := CTRL_WORD_AH; // AH
  IF boDriveX_StandStill THEN
    OpMode := MODE_DRV_OFF; // set breakpoint here to check
  END_IF

END_CASE

//actual values
rActPosition := DINT_TO_REAL(diDriveX_ActualPosition)/DINT_TO_REAL(WEIGHTING_FACTOR_POS);
rActVelocity := DINT_TO_REAL(diDriveX_ActualVelocity)/DINT_TO_REAL(WEIGHTING_FACTOR_VEL);

boError:= wDriveX_StatusWord.13;
boWarning:=wDriveX_StatusWord.12;
// dwDriveX_DiagnosisNumber; // diagnosis number
// Clear error: write S-0-0099 to b0011 with IL_SIIISvcWrite()

```

6.6.3 Drive as PLC device – More field buses

Free programming

Drives can be freely programmed at all other available field bus master connections as described for Sercos III drives. In contrast to Sercos III, not the parameters S-0-0134/S-0-0135 are configured as control/status word for the

IndraDrive, but the parameters P-0-4077/P-0-4078 field bus: control/status word. Thus, some bits are already configured in the control/status word as described in [chapter "Sercos II drive as PLC device – Parameterization" on page 309](#) for Sercos in the signal control/signal status word.

RMB_PLCOpenFieldBus

Instead of programming freely, the RMB_PLCOpenFieldbus library can be used to program with the function blocks according to the PLCopen.

6.7 PowerSupply (power supply) as PLC device

6.7.1 General information

Power supply devices with a separate master communication (e.g. for IndraDrive C an HMV05, for IndraDrive Mi a KMV03) are available for the drive firmware MPx19 and higher.

An "intelligent" power supply is provided with a bus address and a parameter block to set functionalities (e.g. device control word and device status word).

For more information on the device, refer to the Functional Description "Parameter Description for Power Supplies" of the drive documentation.

Such power supplies are supported as PLC device by IndraLogic XLC.

As for "Drive as PLC device" no A-parameters are available for this device. Control and commanding is executed via PLC ([chapter 6.6 "Drive as PLC device" on page 308](#)).



For parameter backups and firmware updates (e.g. FWA-INDRV*-PSB-19Vxx), there has to be a direct connection with IndraWorks Ds.

IndraLogic XLC allows an individual parameterization on the device in the IndraWorks project tree.

The power supply is created in the IndraWorks project by dragging the device from the library window (Periphery folder, folder for the respective field bus) to the "Field Bus" node.

The individual power supplies for Sercos are located in the library of the Sercos peripherals library.

6.7.2 Power supply (PowerSupply) as PLC device, parameterization

A dialog-supported parameterization of the power supply is performed via IndraWorks Ds.

IndraLogic XLC Allows a parameterization using individual parameters.

It has to be communicated directly via PLC and power supply. The PC can be connected to a free Sercos port of the supply unit or communicate via a "Sercos III NetSwitch" (ordering no.: R911328254 HAWA HGS NS-S3-1NRT). The Windows command "route" can also route the IP communication via the control.

The IP addresses are preset by the control. The presetting depends on the control address C-0-0031 and the Sercos address if the Sercos device is provided with the corresponding parameters (class SCP_NRT or SCP_NRTPC in S-0-1000).

- Default address of the Sercos device if C-0-0031=0: 172.31.254.<SercosAddress>. Otherwise, 172.31.<C-0-0031>.<SercosAddress>
- Default address of the control via a Sercos port if C-0-0031=0: 172.31.254.254. Otherwise, 172.31.<C-0-0031>.254

Axes

For information on the IP address specification, refer to [tab. 3-4 "Automatically generated IP address and subnet mask" on page 70](#).

The cyclically applied parameters are configured at Sercos and - in contrast to other field buses - below the "Sercos" node in IndraWorks.

Set "Voltage control" as primary operation mode for the power supply. This is also the default value.

Parameterizing the power supply

1. PC with second network card or USB <-> Ethernet adapter. Set IP address of the network card or adapter to the power supply range 172.31.x.x
2. Start IndraWorks Ds. Select connection -> IP address search. Enter IP address of the drive and search. Connect to device.
3. Check operation modes (primary operation mode) in IndraWorks Ds:

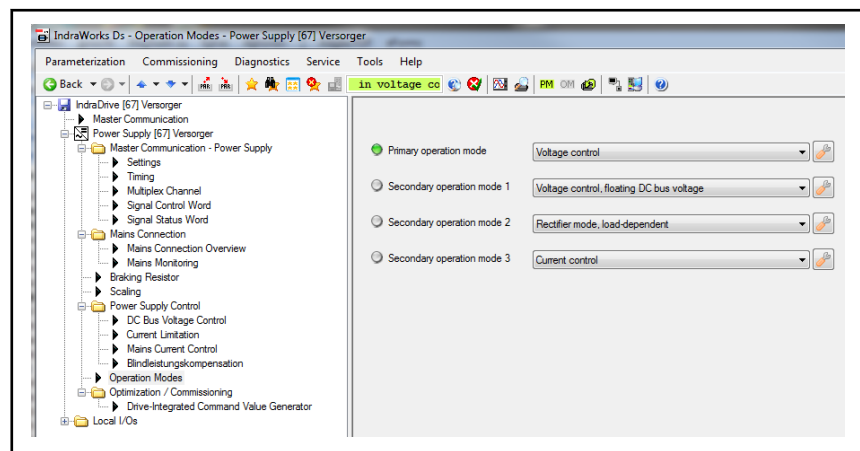
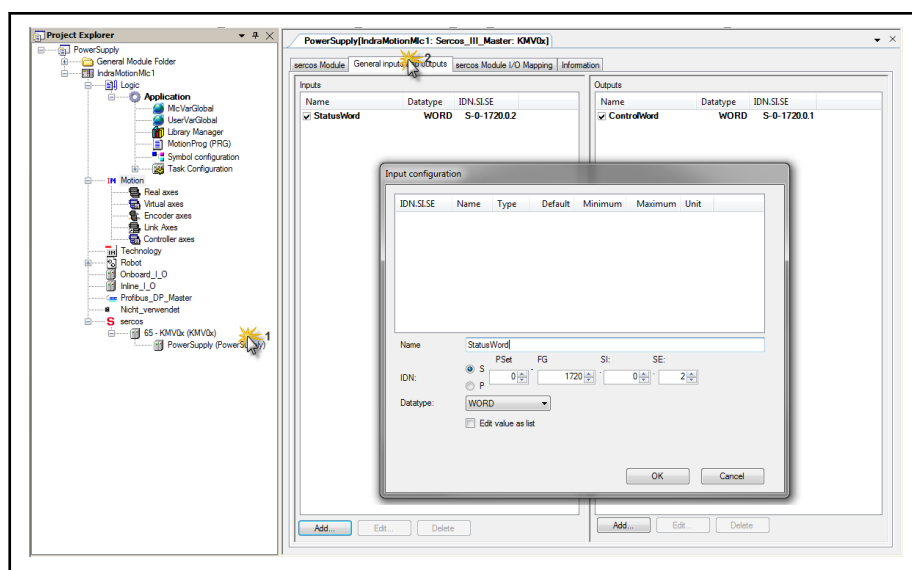


Fig. 6-92: Dialog: IndraWorks Ds, operation modes

4. If required, perform another parameterization of the power supply in IndraWorks.

6.7.3 Power supply as PLC device – Configuring cyclic data

Cyclically applied data is configured in IndraWorks below the "Sercos" node in the dialog "General Inputs and Outputs".



1 Double-click on the "PowerSupply" node below the Sercos device

2 Select driver inputs and outputs

Fig. 6-93: Example configuration of cyclic parameters in the drive diagram AT (inputs) and master data telegram MDT (outputs)

This is the minimum configuration with the status word S-0-1720.0.2 "Power supply status word" as PLC input (data type: WORD) as well as the control word S-0-1720.0.1 "Power supply control " as PLC output (data type: WORD). These parameters are required to command the power supply to retrieve the status.

The entered name is subsequently output in the "Channel" column in [fig. 6-94 "Assigning PLC variables to the configured parameters" on page 320](#).

Select the correct data type with the suitable length and without or with sign depending on the attribute.

Data length in the attribute	Format in the attribute	Data type in the PLC
2 bytes	BIN or HEX	WORD
4 bytes	BIN or HEX	DWORD
2 bytes	DEC_MV	INT
4 bytes	DEC_MV	DINT
2 bytes	DEC_OV	UINT
4 bytes	DEC_OV	UDINT

Tab. 6-17: Assignment: Parameter attribute <> Data type in the PLC

The parameters are assigned to the PLC variables in the "Sercos Module I/O Mapping" tab. New variables can be created or the input and output parameters can be mapped on existing PLC variables.

Variables are created in the following:

Axes

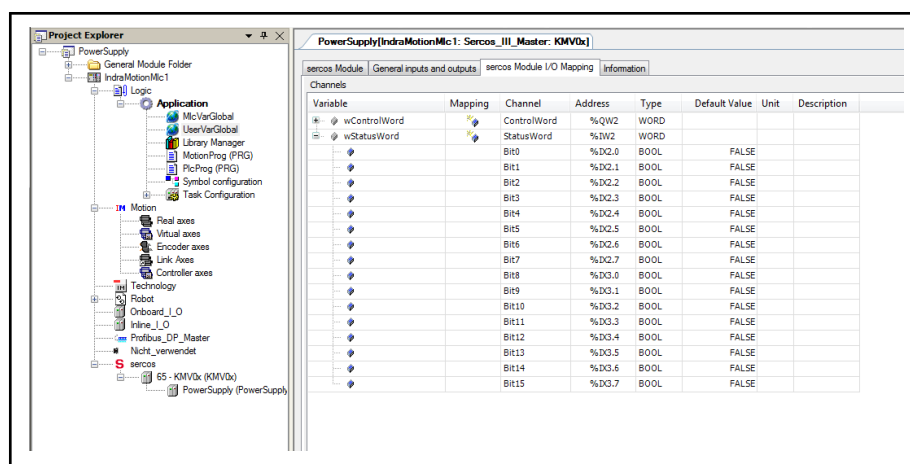


Fig. 6-94: Assigning PLC variables to the configured parameters

Configuring cyclic data

1. Double-click on the node of the power supply in the IndraWorks project. Select the "General Inputs and Outputs" tab.
2. Parameter S-0-1720.0.2 "Power supply status word" has to be configured.
Check for attribute in the help of S-0-1720.0.2: Length: 2 bytes, format: BIN.
Select "Add" for the inputs and fill out the fields as follows:
 - Field name: "wStatusWord"
 - Field FG: "1720.0.2"
 - Data type: Select "WORD"
3. Parameter S-0-1720.0.1 "Power supply control word" has to be configured.
Check for attribute in the help of S-0-1720.0.1: Length: 2 bytes, format: BIN.
Select "Add" for the inputs and fill out the fields as follows:
 - Field name: "wControlWord"
 - Field FG: "1720.0.1"
 - Data type: Select "WORD"
4. Enter more parameters until all parameters are configured as shown in [fig. 6-94 "Assigning PLC variables to the configured parameters" on page 320](#).
5. Go to the "Sercos Module I/O Mapping" tab and enter the PLC variables into the "Variable" column as shown in [fig. 6-93 "Example configuration of cyclic parameters in the drive diagram AT \(inputs\) and master data telegram MDT \(outputs\)" on page 319](#).
6. Load PLC program. Switching to "P4" should be possible via IndraWorks now.

6.7.4 Power supply (PowerSupply) as PLC device – PLC programming

The configured PLC variables of the cyclically configured parameters for control and status word can now be used in the PLC program.

For the bit assignment of the power supply parameters

- S-0-1720.0.1 "Power supply control word"

- S-0-1720.0.2 "Power supply status word", refer to the "Parameter Description" of the drive.

The power supply can be commanded in any cyclic task. (e.g. Plc_Task and PlcProg).

Program example

Switching on and off power can be seen in the following:

- **1st state: Activate main contactor**
Set control word, bit 12 (16#1000)
Query status word, bits 4, 5 and 7 (16#00B0); Continue switching state
- **2nd state: Activate switch connector (load DC voltage)**
Set control word, bits 12 and 14 (16#5000)
Query status word, bits 4, 5, 6 and 7 (16#00B0); Continue switching state
- **3rd state: Enable operation mode -> constant intermediate circuit voltage**
Set control word, bits 12, 14 and 15. Bits 8 and 9 are not set for primary mode selection (16#D000)
Status word, bits 4, 5, 6, 7, 14 and 15 (16#C0F0)
Connected axes can be commanded
- **4th state: Switch off power**
Control word deletes all bits (16#0000)

PLC programming

1. Open the "PlcProg" program in the IndraWorks project.
2. Copy the first program code "Declaration" below the declaration part.
3. Copy the second program code "Implementation" below the implementation part. Do not copy page break or delete it subsequently.
4. Compile, load and start PLC program.
5. Set power request "bSetPower".
6. Check whether power was added (status word, bits 4, 5, 6, 7, 14 and 15).
7. For the final program implementation, sections from this program can be copied and used.

Declaration

```

PROGRAM PlcProg
VAR
    iStatePowerSupply: INT;          (* state for power supply *)
    bSetPower: BOOL;                 (* set power *)
END_VAR

VAR CONSTANT
    STATE_ENABLE_MAINS_CONTACTOR : INT := 1;    (* state enable mains contactor *)
    STATE_ENABLE_POWER_SUPPLY     : INT := 2;    (* state enable power supply, load dc voltage *)
    STATE_ACTIVATE_POWER_SUPPLY   : INT := 3;    (* state activate power supply *)
    STATE_POWER_OFF               : INT := 4;    (* state power off *)
END_VAR

(* S-0-1720.0.1 Power supply control *)
(* Bit 15: power supply activation *)
(* Bit 14: power supply enable *)
(* Bit 13: reserved *)
(* Bit 12: mains contactor *)
(* Bit 11: reserved *)
(* Bit 10: DC discharging *)
(* Bit 09: selection of OpMode *)
(* Bit 08: selection of OpMode *)

(* S-0-01720.0.2 Power supply status *)
(* Bit 15: voltage boost ON/OFF *)
(* Bit 14: DC bus okay *)
(* Bit 13: error *)
(* Bit 12: warning *)
(* Bit 11: brake resistor *)
(* Bit 10: DC discharging *)
(* Bit 09: selection of OpMode *)
(* Bit 08: selection of OpMode *)

```


Axes

(* Bit 07: reserved *)	(* Bit 07: feedback signal of mains contactor *)
(* Bit 06: reserved *)	(* Bit 06: mains barrage *)
(* Bit 05: reserved *)	(* Bit 05: hardware enable *)
(* Bit 04: reserved *)	(* Bit 04: mains status *)
(* Bit 03: reserved *)	(* Bit 03: reserved *)
(* Bit 02: reserved *)	(* Bit 02: reserved *)
(* Bit 01: reserved *)	(* Bit 01: reserved *)

Implementation

```

(* set power request and check status *)
IF bSetPower AND wStatusWord = 16#00A0 OR wStatusWord = 16#0020 THEN
  iStatePowerSupply := STATE_ENABLE_MAINS_CONTACTOR;
ELSIF bSetPower AND wStatusWord = 16#00B0 THEN
  iStatePowerSupply := STATE_ENABLE_POWER_SUPPLY;
ELSIF bSetPower AND wStatusWord = 16#40F0 THEN
  iStatePowerSupply := STATE_ACTIVATE_POWER_SUPPLY;
ELSIF bSetPower AND wStatusWord = 16#C0F0 THEN
  iStatePowerSupply := STATE_POWER_OFF;
END_IF

(* state machine for power supply *)
CASE iStatePowerSupply OF

  STATE_ENABLE_MAINS_CONTACTOR: (* enable mains contactor *)

    wControlWord := 16#1000;    (* set bit 12 *)

    IF wStatusWord = 16#00B0 THEN    (* check bit 4, 5, 7 *)
      iStatePowerSupply := STATE_ENABLE_POWER_SUPPLY;
    ELSE
      iStatePowerSupply := STATE_ENABLE_MAINS_CONTACTOR;
    END_IF

  STATE_ENABLE_POWER_SUPPLY: (* enable power supply, load DC voltage *)

    wControlWord := 16#5000;    (* set bit 12, 14 *)

    IF wStatusWord = 16#40F0 THEN    (* check bit 4, 5, 6, 7 *)
      iStatePowerSupply := STATE_ACTIVATE_POWER_SUPPLY;
    ELSE
      iStatePowerSupply := STATE_ENABLE_POWER_SUPPLY;
    END_IF

  STATE_ACTIVATE_POWER_SUPPLY: (* enable poser supply, operation mode constant DC voltage *)

    wControlWord := 16#D000;    (* set bit 12, 14, 15 *)

    IF wStatusWord = 16#C0F0 THEN    (* check bit 4, 5, 6, 7, 14, 15 *)
      iStatePowerSupply := STATE_POWER_OFF;
    ELSE
      iStatePowerSupply := STATE_ACTIVATE_POWER_SUPPLY;
    END_IF

  STATE_POWER_OFF: (* power off *)

    IF NOT bSetPower THEN
      wControlWord := 16#0000;    (* reset bits *)
    END_IF

    IF NOT wStatusWord = 16#0040 THEN    (* check bit 6 *)
      iStatePowerSupply := STATE_POWER_OFF;
    END_IF

END_CASE

```

7 Motion functionality

7.1 General information

This chapter describes the various axis modes. First, the single-axis modes, such as the positioning and velocity control, are described. Subsequently, synchronous modes, such as velocity and synchronous axis motions and 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, refer to the manuals "Rexroth IndraLogic XLC IndraMotion MLC 13VRSPLCopen Libraries" and "Rexroth IndraLogic XLC IndraMotion MLC 13VRS Generic Application Template" (see [chapter 1.4 "Required and supplementing documents" on page 18](#))

7.2 Cycle Times

To support the task management (see also [chapter 5.4 "Task system " on page 177](#)), 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).

	Minimum	Current	Maximum	Unit
Current Motion cycle time		16 000		µs
Time for actual value recording	7	24	100	µs
Interruption caused by motion-synchronous PLC task	0	0	43	µs
Time for command value generation	0	1	43	µs
Total time	8	25	104	µs

Fig. 7-1: "Cycle times" dialog

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". The sensitivity describes the number of successive cycles a timeout is accepted. Additionally, the absolute number of Motion errors and the

Motion functionality

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.



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...40ms).

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.

The following cycle time are set by default and can be changed:

L25	L45	L65
8 ms	4 ms	2 ms

Tab. 7-1: Sercos cycle times set by default according to the control hardware
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 the lists C-0-0413 and C-0-0414

7.3 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 into 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 bus, the acyclic communication is established, but not yet the cyclic communication.

Operating mode (BB): This operating state runs the Motion functionality (drives process the motion commands of the control). These motion commands are normally controlled by the PLC and, in this case, the PLC has to be in 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.

7.4 Parameters

The control and the drives communicate - with very few exceptions - using parameters.

The parameters perform the following tasks:

- Determining the configuration
- Parameterizing the control loop
- Triggering and controlling drive functions and commands
- Applying 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 parameter and the communication phase. Certain parameter values (operating data) are checked for validity by the control.

The following tools are used to retrieve, modify and save parameters into a file or to load the file and write the parameters.

- [chapter "Parameter editor" on page 109](#)
- [chapter "Parameter group" on page 111](#)
- [chapter "Parameters - Export" on page 113](#)
- [chapter "Parameters - Import" on page 115](#)

Parameters can only be accessed in the following modes:

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

7.5 Parameterization level

Irrespective of the control and communication phase, devices can be configured via the parameterization level. Thus, axes can be moved further, the I/Os can continue to exchange cyclic data and the PLC program keeps running. Thus, these devices support two modes:

- **Parameterization mode**

In the parameterization mode, all device parameters - that are not write-protected - can be changed. This does not apply to communication parameters such as configuration, timing or user-defined data containers.

Motion functionality

If a device is in the parameterization mode, it remains irrespective of the communication phase

- **Operating mode**

Only parameters without write protection in this mode can be changed in the operating mode. The parameter crosscheck is performed when switching from parameterization to operating mode. The adjustment of A- to S-/P-parameter is also executed

Supporting the parameterization level

For information on whether a device supports the parameterization level, refer to the parameter A-0-0004 "Axis type".

Switching the modes

It is switched between the two modes for one device via the parameter A-0-0024 "Axis state". To switch multiple devices, use the parameter C-0-1070, Command:"Switch PM/OM". To switch an axis, the corresponding axis has to be in the "Standstill" state. Furthermore, the real axis has to be in the "Power off" state. An axis in the parameterization mode cannot be traversed.

7.6 Modes for project editing

7.6.1 Introduction

Offline mode

In offline mode, all data saved on your PC can be entered into the project. That is data not requiring any data exchange with a control or drive. Firmware is also only downloaded to the control in offline mode.

If a project is opened, IndraWorks is automatically in offline mode.

Online mode

Data requiring a communication between the programming device and the control or drive can be changed in online mode. This data is stored - at least partially - in the control or drive or is retrieved from them and is thus only available if a communication is established. No devices can be created or deleted in online mode. An axis can only be created or deleted in offline mode!

There are different options to go to the online mode:

1. **Switching online:** This operation establishes a communication to the control without changing the control configuration. Even at system run-time, specified control data changes can be made or the system state can be diagnosed. Switch online either via the respective button in the "Project" toolbar or via the context menu of the control.

If the configured configuration matches the configuration in the control, control data and all connected devices can be subsequently accessed. If the two configurations differ, IndraWorks indicates the difference with a warning in the message window and the text "[partially synchronized]" at the control node. In this case, only the following online functions are permitted:

- Start of the "Error/Diagnostic Memory" dialog
Remark: For the control version 13V02 and higher, this dialog can also be started without switching online before
- Phase switching: Sercos bus
- Start of the parameter editor of the control. The parameter editors of the individual devices are locked. However, their parameters can

be accessed via the parameter editor by specifying the logic number in the parameter name. The position limit value of the axis with the axis number 4 can for example be accessed by entering "A004:A-0-0030"

- Start of the control oscilloscope
 - Start the parameter import or export
 - Start the FTP Explorer
 - Open the communication settings of the control
 - Open the advanced settings of the control
 - Start the "Task Viewer" dialog
2. **Download Motion configuration:** This operation downloads the Motion configuration (e.g. configured axes and kinematics) to the control and connects subsequently to the control. The configuration download ensures that all online functions are available. If the configuration could not be completely loaded to the control (e.g. since the Sercos interface of the control is not connected and thus switching to P2 is not possible), the same restrictions as described under the point "Switching online" apply. The Motion configuration can be downloaded via the "Synchronize" context menu of the control as well as via the corresponding button in the "XLC/MLC Info" toolbar.
 3. **Upload Motion configuration:** This operation applies the Motion configuration from the control to the project. The previous configuration in the project is then deleted except for the control. A communication is then established to the control. The Motion configuration can be uploaded via the "Synchronize" context menu of the control as well as via the corresponding button in the "XLC/MLC Info" toolbar.

 WARNING

Uploading the Motion configuration changes data in the project. Thus, when switching online, no differences are detected between the configured configuration and the control configuration. Currently, not all data can be uploaded, e.g. all icon assignments and parameters are lost for I/O devices. When logging in to the PLC application, the respective data is also lost in the control.

Only upload a Motion project from the control if there is no matching project on the PC. After uploading a Motion project from the control, check the PLC configuration before logging in to the application.

Compare Motion configuration: IndraWorks checks in all three cases, whether the Motion configuration in the project matches the configuration in the control. If there are configuration settings that do not match, IndraWorks only enables a subset of the available online functions. These differences can be displayed to decide for example whether to download, upload or manually adjust the project. Select the corresponding entry in the "Synchronize" context menu of the control or press the corresponding button in the "XLC/MLC Info" toolbar. The dialog is described under [chapter 7.6.4 "Comparing control and project configuration" on page 338](#).

Access to control or drive data also depends on the device state (P0/P2 parameterization mode/BB operating mode) of the Motion component and (STOP/RUN) of the PLC.

Motion functionality

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 not 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.

At the end of the "offline parameterization" mode, switch back to offline mode.

The 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**. See also [chapter 7.6.3 "Loading Offline Parameters to Device"](#) on page 337.

7.6.2 Transitions between modes

Mode switching

It is possible to switch between the three modes as shown in the following figure:

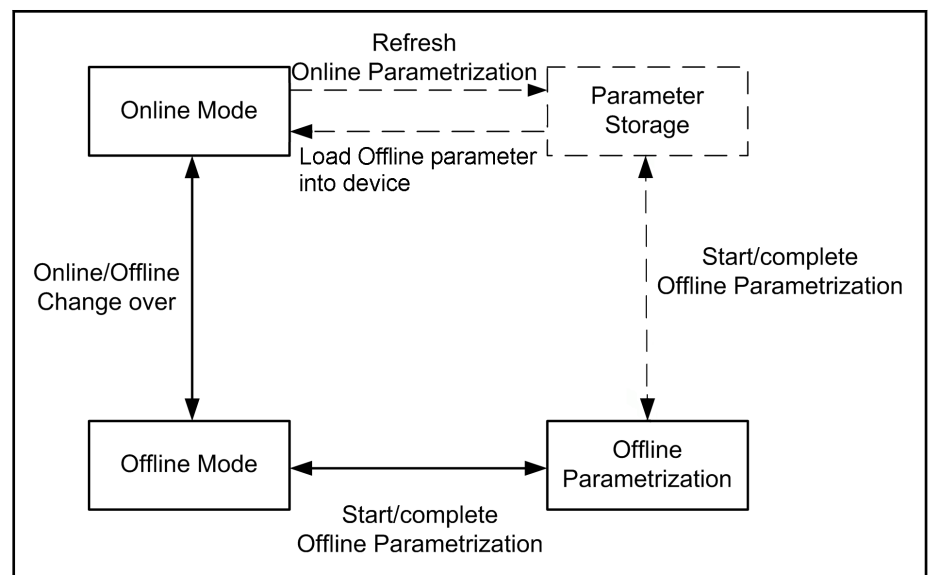


Fig. 7-2: Transitions between the modes

- To switch online/offline, use the toolbar or the context menu of the control, e.g. **Switch Online**, **Download Motion Configuration** or **Upload Motion Configuration** or **Switch Offline**
- Start the [chapter "Offline Parameterization" on page 328](#) 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 [chapter "Online mode" on page 326](#) via the context menu item **Refresh Online Parameterization**.

To switch to online mode, the following prerequisites have to be met:

- **Switching online:** The type of the configured control has to match the real control. Thus, it is not possible to switch a project online with a configured control of type IndraMotion MLC L45 to an actually connected control of the type IndraLogic XLC L45
- **Download Motion configuration, Upload Motion configuration:** In addition to the previous condition, the configured firmware version has to match the connected control. Thus, it is for example not possible to switch a project online with the configured firmware version MLC 12VRS to a connected control with the firmware MLC 13VRS.

If there are deviations, the configured control has to be adjusted to the actual conditions. See also ["Device type" on page 50](#).

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**. See also [chapter 7.6.3 "Loading Offline Parameters to Device" on page 337](#).

Adjustment of command/actual configuration

To adjust the command and actual configuration of the Sercos devices, use the dialog "Sercos Configuration". This dialog can be opened in online and offline mode via the context menu of the Sercos node. Additionally, the dialog is displayed when downloading the Motion configuration if the Sercos configuration check found a warning or an error (refer to the following figure).

Case 1: Download Motion configuration

Motion functionality

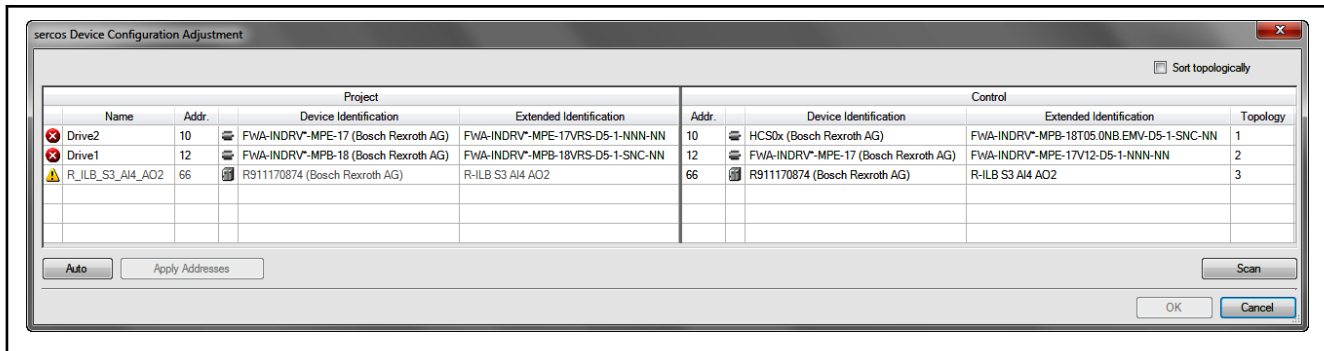


Fig. 7-3: Example of the Sercos configuration dialog when switching online

The left part of the table displays all configured devices and the right part all devices connected to the Sercos bus. The devices are assigned via the Sercos address.



To exactly identify a device in the field, let its Sercos LED flash. Therefore, go to the "Control" column and select the context menu item "Identify Device Visually (flashing)". As long as a device is visually identified, the device icon is highlighted in green. When closing the dialog, scanning the Sercos bus or switching the phase, the visual identification of the device is automatically switched off.

Note that not all Sercos devices support the command "Visual identification". In this case, the device does not respond to the command, even though the device is highlighted as "Visually identified" in the dialog.

To solve the conflicts in this example, proceed as follows:

1. If the Sercos bus cabling has been changed in the meantime, the found devices on the right side of the table can be updated via the **Scan** button, see [chapter "Scanning bus configuration" on page 333](#).
2. The first column displays error and warning icons. More detailed information is provided in the tooltip on the respective icon. This information is significant for a further analysis. The icons have the following significance.
 - **No icon:** Device and configuration match. There are neither warnings nor errors
 - **Yellow warning icon:** There are discrepancies in the configuration (e.g. an I/O device is disabled, but connected to the control), but the Motion configuration could still be downloaded. However, it can be possible that the Sercos bus cannot be switched to the operating mode later on
 - **Red error icon:** The difference between the configured and the connected Sercos device is so big that the Motion configuration cannot be downloaded, as the firmware version of a drive does not match for example. Adjust the configuration to download the Motion configuration

The columns "Device Identification" or "Extended Identification" contain information from the electronic type plate or the device firmware. To compare the configured and the connected device, only the content of the "Device Identification" column for PLC devices and only the content of the "Extended Identification" column for drives are compared.

Motion functionality

The device activation is shown by a grey font color. I/O devices are enabled and disabled via a context menu in the "Project" column, drive axes via a context menu in the "Control" column.

In this case, the I/O device is disabled in the project, but connected to the control. In addition, the firmware of the configured drives does not match the actually configured drives. It seems to be the case that the Sercos addresses of the drives were exchanged in the assignment.

3. Click on **Auto**. IndraWorks tries to adapt the activation of the configured devices to the actually found devices to match them the best possible (see [fig. 7-4 "Sercos Device Configuration" dialog after pressing the Auto button](#) on page 331).

I/O devices are differently activated than drive axes. The activation of an I/O device is saved in the project and set according to the existence of the actually connected I/O device. For drive axes, the activation is saved in the control. Saving in the control is only possible if the drive axis is already configured in the control. In this case, press the **Auto** button causes to adjust the activation in the control and thus, the project remains unchanged.

In this case, press **Auto** to adjust the activation of the I/O device in the project.

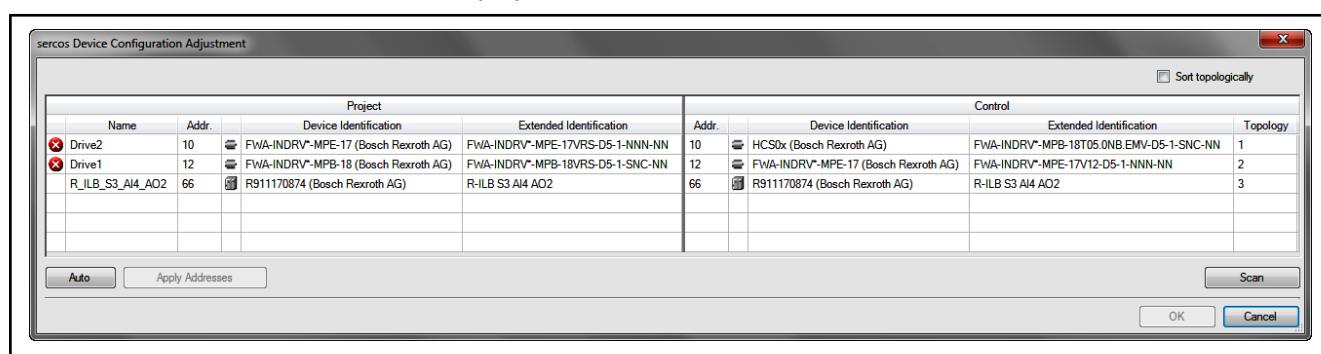


Fig. 7-4: "Sercos Device Configuration" dialog after pressing the **Auto** button

4. The problem of the non-matching firmware of both the drives was not solved by pressing the **Auto** button. Obviously, the Sercos addresses of both drives are exchanged. This cannot be solved by changing the activation. There are different approaches to solve this problem:
 - Adapt the Sercos addresses in the project to the connected drive axes by changing the Sercos addresses in the "Project" column. Then, press the **Apply Addresses** button. In the period between changing and pressing the button, the changed fields are highlighted in dark grey
 - The Sercos addresses of the connected drive axes can be changed directly at the display of the drives. Then, press the **Scan** button. This is generally not possible for I/O devices, as I/O devices are normally not provided with a display
 - Adapt the Sercos addresses of the connected drive axes to the axes in the project by changing the Sercos addresses in the "Control" column. Then, press the **Apply Addresses** button. In the period between changing and pressing the button, the changed fields are highlighted in dark grey. This method can also be used to assign a new Sercos address to I/O devices

Motion functionality

In this example, the problem is solved as described in the latest option. Thus, the Sercos addresses of the connected drives are adapted to the specifications in the project.

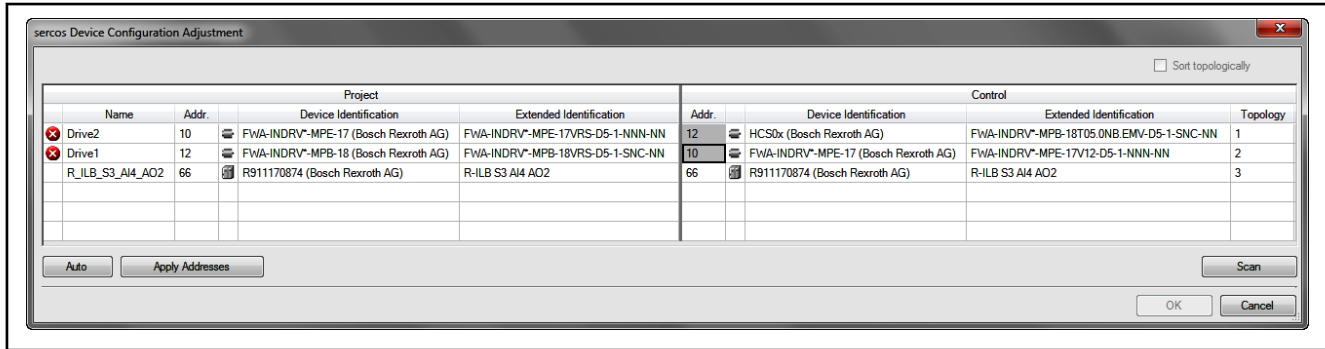


Fig. 7-5: "Sercos Device Configuration" dialog after shifting the devices (drives)

- Press the **Apply Addresses** button and the conflicts are recovered and the **OK** button is enabled. Press this button to close the dialog and to continue the Motion configuration download. The button is only available if the Sercos configuration is correct.

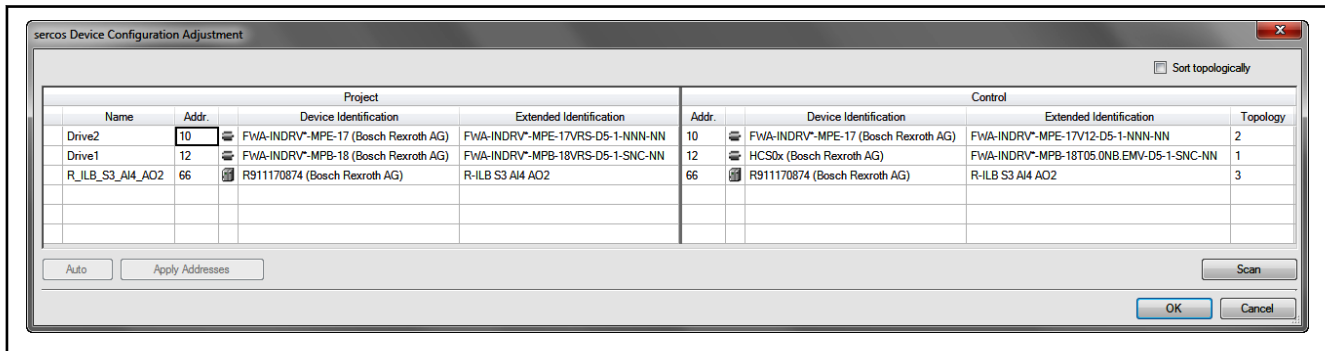


Fig. 7-6: Correct Sercos device configuration

Case 2: Online mode

In this case, the command configuration can no longer be changed. Addresses of connected devices and the activation of drive axes are still possible.

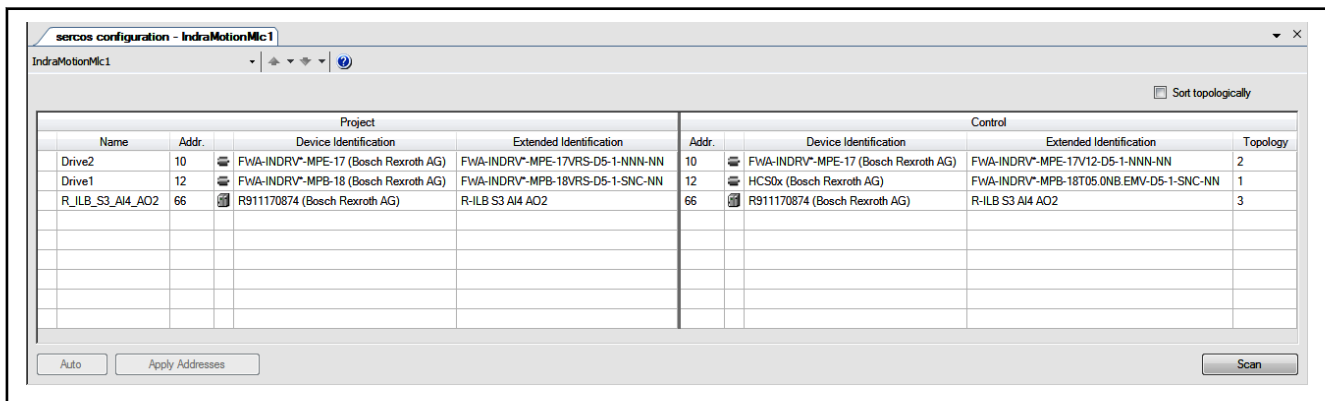


Fig. 7-7: Sercos device configuration dialog in online mode

Case 3: Offline mode

In this case, the dialog is as in case 1 "Download Motion configuration". The essential difference is that the buttons **OK** and **Cancel** are missing. If the dia-

Motion functionality

log is opened, the right side of the table is empty. To fill the right side of the table, press **Scan**.

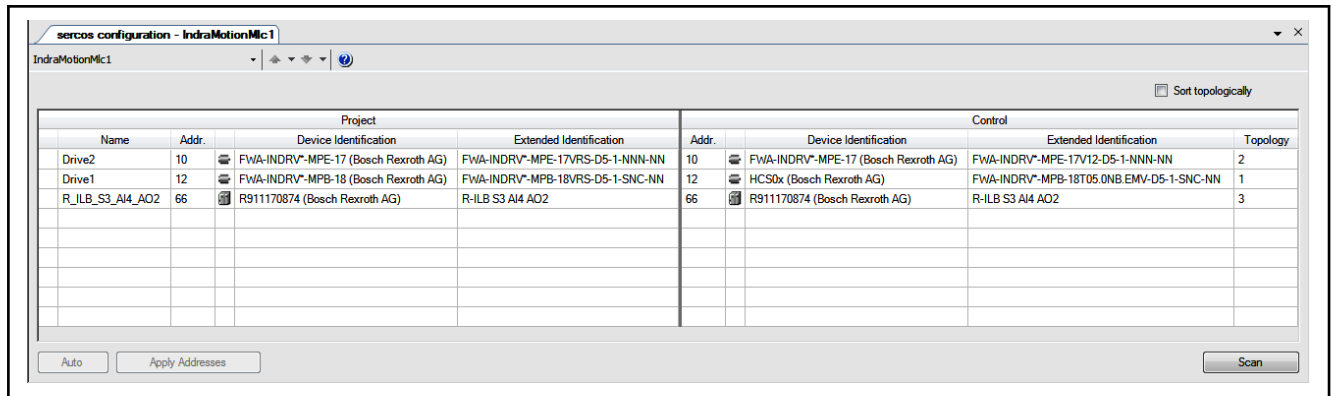


Fig. 7-8: "Sercos Device Configuration" dialog in offline mode

Scanning bus configuration

Use the "Scan Bus Configuration" command to scan and apply drives and I/O devices connected to the Sercos bus of the control to the project. The following dialog is displayed:

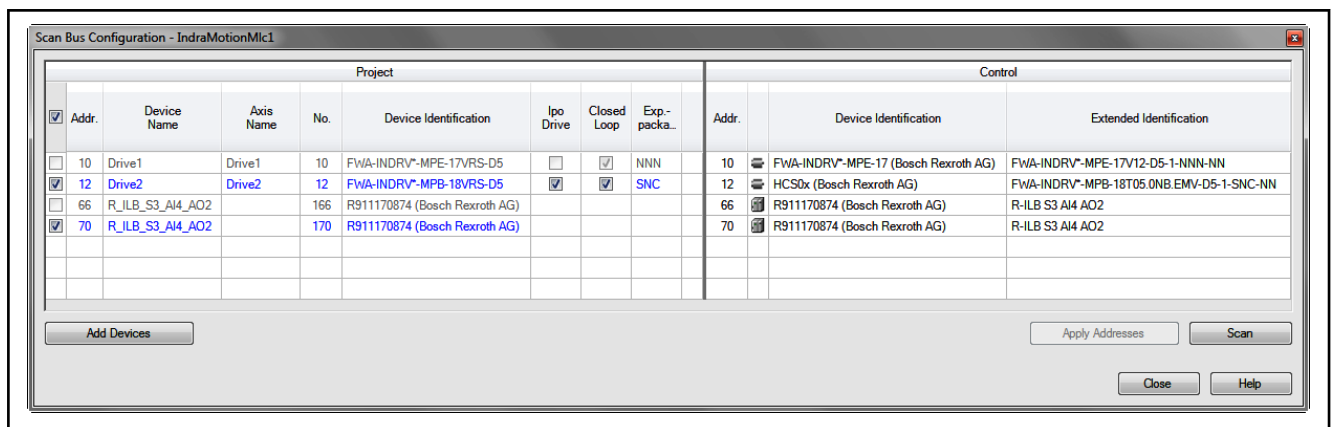


Fig. 7-9: "Scan Bus Configuration" dialog

The dialog displays the devices sorted according to their Sercos address. The data of the devices found at the bus are located on the right of the table. If several devices with the same address are connected to the Sercos bus, a corresponding error message is displayed.

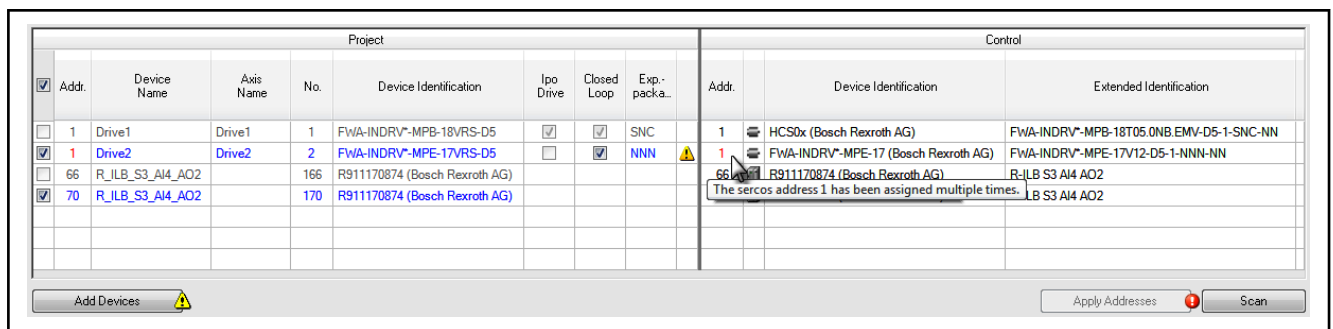


Fig. 7-10: Sercos address is assigned multiple times

In this case, assign unique addresses to these devices. The Sercos addresses of the corresponding entries (changed data is first displayed in gray) have

Motion functionality

to be changed. Apply these changes with the **Apply Addresses** button in the control (remote address allocation).

Project								Control			
Addr.	Device Name	Axis Name	No.	Device Identification	Ipo Drive	Closed Loop	Exp. packa...	Addr.	Device Identification	Extended Identification	
☐	1	Drive1	Drive1	1	FWA-INDRV-MPB-18VRS-D5	☒	☒	SNC	1	HCS0x (Bosch Rexroth AG)	FWA-INDRV-MPB-18T05.0NB.EMV-D5-1-SNC-NN
☒	1	Drive2	Drive2	2	FWA-INDRV-MPE-17VRS-D5	☐	☒	NNN	12	FWA-INDRV-MPE-17 (Bosch Rexroth AG)	FWA-INDRV-MPE-17V12-D5-1-NNN-NN
☐	66	R_ILB_S3_A14_AO2		166	R911170874 (Bosch Rexroth AG)				66	R911170874 (Bosch Rexroth AG)	R-ILB S3 A14 AO2
☒	70	R_ILB_S3_A14_AO2		170	R911170874 (Bosch Rexroth AG)				70	R911170874 (Bosch Rexroth AG)	R-ILB S3 A14 AO2

Buttons: Add Devices, Apply Addresses, Scan

Fig. 7-11: Remote address allocation

After the changes have been applied, the bus configuration is automatically scanned again.

To add the newly found devices to the project, some properties shown on the left have to be determined. Newly found devices are highlighted in blue. Already configured devices are displayed in gray and can here not be changed.

Highlight the devices to be added to the project in the first column.

The Sercos address ("Addr.") is applied by the scanned device and cannot be changed here.

For drives, device names or axis names can be changed. Since global variables have to be generated for the identifiers in the PLC program, they have to comply with the naming conventions of the IEC 61131. The axis name generally corresponds to the device name (except for multi-axis drives; the axis name consists of the device name and an appendix that can be changed here).

The logic number ("No.") can also only be changed for axes. The number is also fixedly specified for I/O devices.

In the column "Ipo Drive", it can be determined where the axis motion (interpolation) is to be calculated. Highlight the selection to calculate the command values in the drive. Otherwise, the command values are calculated in the control (also refer to [chapter 6.2.1 "General information" on page 208](#)).

Set in the "Closed Loop" column, whether to operate the drive with encoder (Closed Loop) or without encoder (Open Loop). Enable this selection only if the corresponding function package is enabled in the drive, as, otherwise, the Sercos devices cannot be switched to the operating mode (also refer to [chapter 6.2.1 "General information" on page 208](#)).

Use the **Add Devices** button to add the selected entries to the project located below the Sercos node of the control.

Incorrect entries such as invalid device or axis names or double axis numbers are labeled with a corresponding warning or tooltip. In most of the cases, the devices can still be created in the project. These properties can be subsequently adjusted. Before switching online, the configured data is checked again. Online switching is not possible in case of error.

If new devices were connected to the Sercos bus of the control in the meantime, click on the **Scan** button to load the bus configuration again.



For controls with a firmware $\leq 12VRS$, creating I/O devices is not supported. The I/O devices are scanned and displayed in the right section of the dialog. However, the left section to insert the project is empty in this case.

Axis synchronization dialog

Switching online

The axis synchronization dialog allows the user to adjust the axis configuration between IndraWorks and the control when going online.

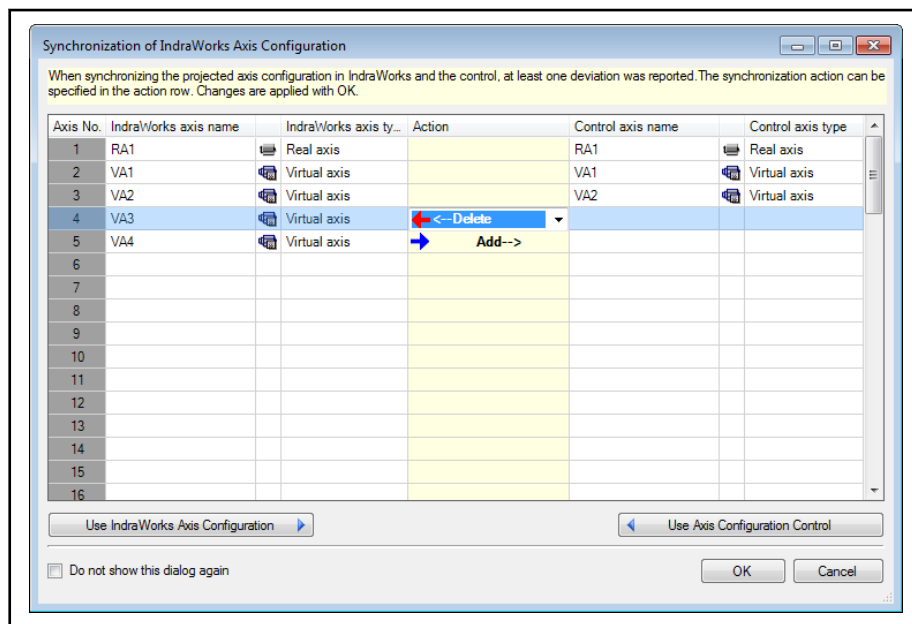


Fig. 7-12: Axis synchronization dialog

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 synchronization 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.

Use the **Use IndraWorks Axis Configuration** or **Use Axis Configuration Control** buttons to select all synchronization actions in a way that 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 (for a description, see "Rexroth IndraWorks 13VRS Engineering" manual, chapter "Versions control", see [chapter 1.4 "Required and supplementing documentations" on page 18](#)) and changes are applied to the IndraWorks project, all axes and drives are checked out after asking the user.



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!

Adjusting the behavior of the axis synchronization dialog

Adjust the behavior of the axis synchronization dialog via the IndraWorks menu item **Tools ► Options**.

Motion functionality

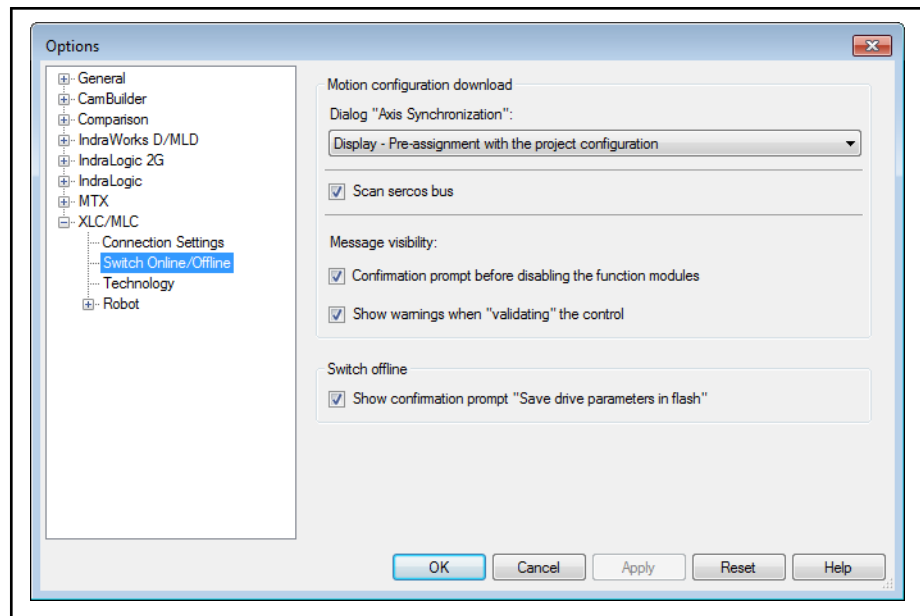


Fig. 7-13: IndraWorks menu item "Tools" -> "Options" -> "Switch Online/Offline"

The following options are available:

1. Do not display - Applying the project configuration: The dialog is not shown when switching online. The axis configuration saved in the project is written to the control.
2. Display - Preassigned with the project configuration: The dialog is shown and all axes are preset. The axis configuration saved in the project is written to the control with **OK**.
3. Display - Preassigned with the control configuration: The dialog is shown and all axes are preset. The axis configuration saved in the control is written to the IndraWorks project with **OK**.

Check the following prompt via the option "Confirmation prompt before disabling function modules".

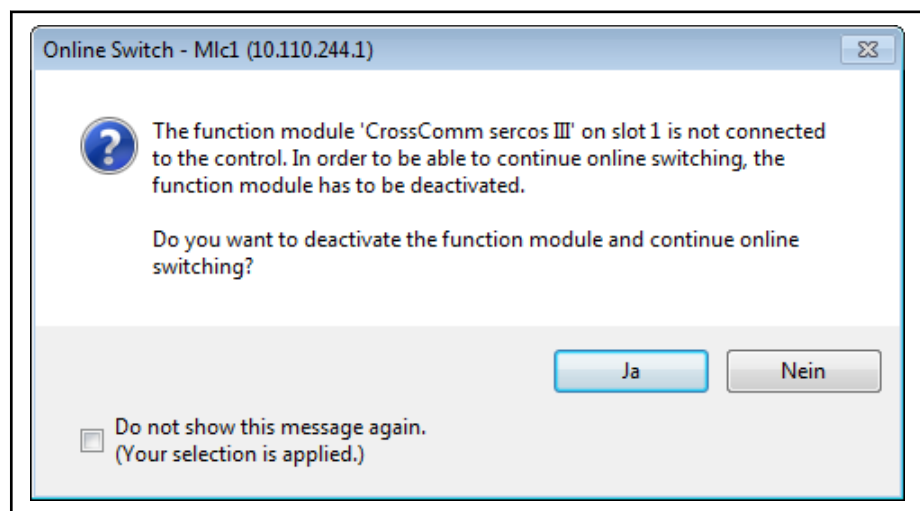


Fig. 7-14: Disabling dialog for function modules

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 in to the control.

7.6.3 Loading Offline Parameters to Device

The "Load offline parameters to device" menu item is used to transfer the modified parameters to the real environment (controls, axes and drives) during [chapter "Offline Parameterization" on page 328](#). For this function, go to the respective menu item in the context menu of the control:

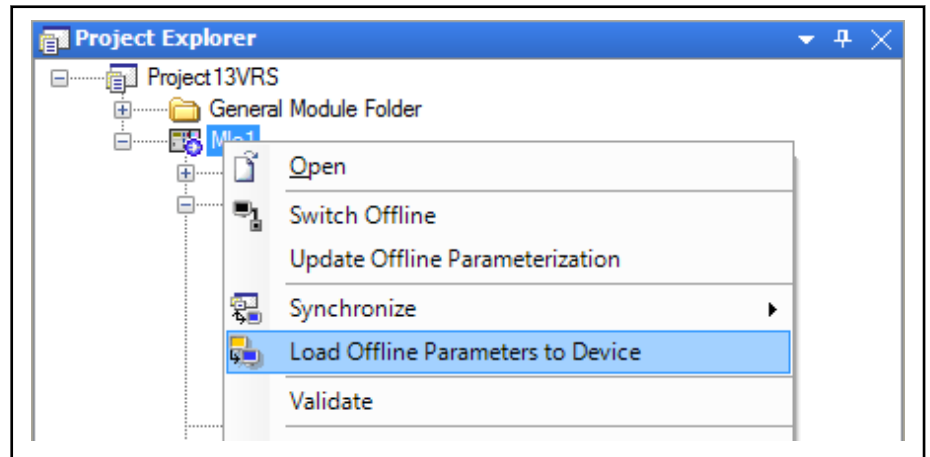


Fig. 7-15: Context menu of a control



This menu item is only available if offline parameters exist.

Subsequently, it is queried whether the synchronization (adjustment) is to be executed and whether the differences are to be displayed. It is also displayed when the offline parameters were created.

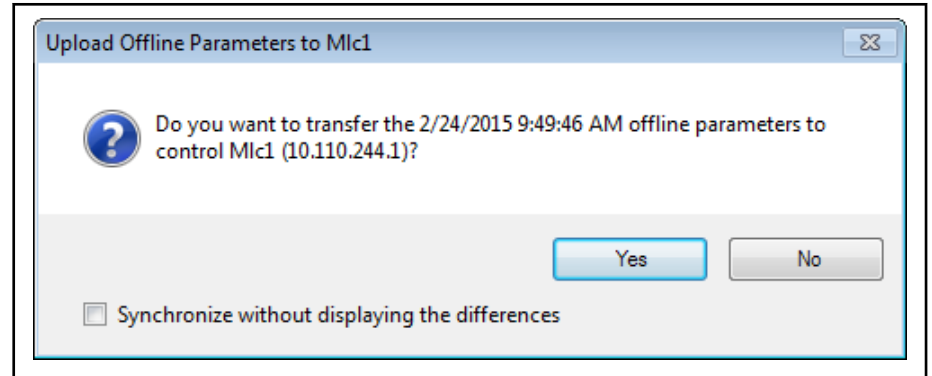


Fig. 7-16: Query for synchronization (adjustment)

The offline parameters are then compared with the ones of the control. Differences are displayed if desired:

Motion functionality

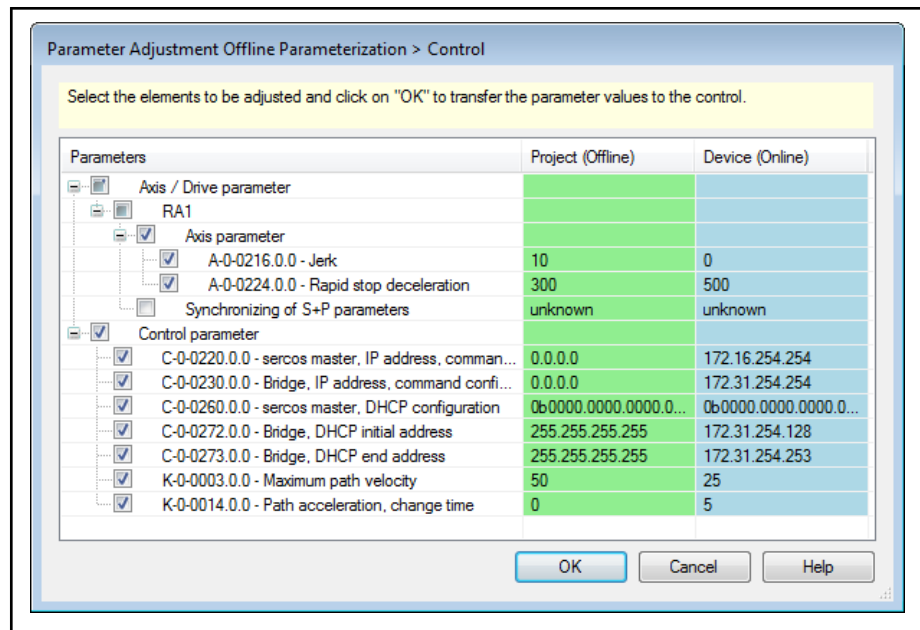


Fig. 7-17: Displaying control and drive parameter differences

Differences in the parameters are listed in the dialog. The parameters to be applied from the project to the devices 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.

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.

Press **OK** to apply the selected parameters.

7.6.4 Comparing control and project configuration

This dialog shows the differences in the Motion configuration between the connected control and the control in the IndraWorks project. Thus, it can be judged what will happen when downloading the Motion configuration to the control.

The configuration in the project and the configuration in the control are shown in two separate columns. All objects are sorted according to their logical address (that is, the axis number for axes, the Sercos addresses for Sercos devices, etc.) and the differences are highlighted.

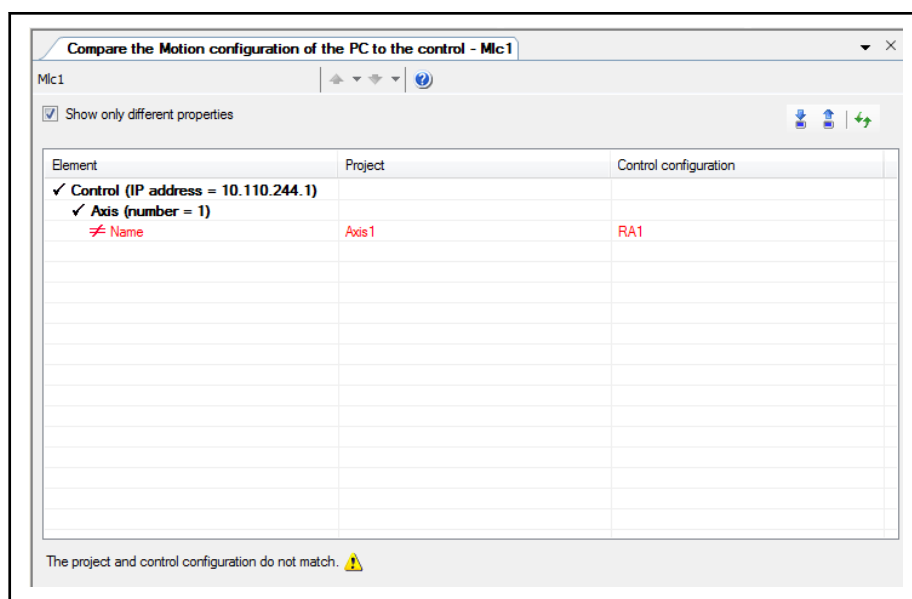


Fig. 7-18: Example: Display of a different axis name



The control data refers to the **control configuration**, not to the connected devices. The upper figure shows only that a Sercos drive called "RA1" is configured on the control. To ensure whether the device is really connected to the control, check the Sercos configuration, see [chapter "Adjustment of command/actual configuration"](#) on page 329.

Instead of showing the differences, all properties can also be displayed by selecting "Show only different properties". This representation is advantageous when copying the content of the table to the clipboard via the context menu for documentation purposes for example.

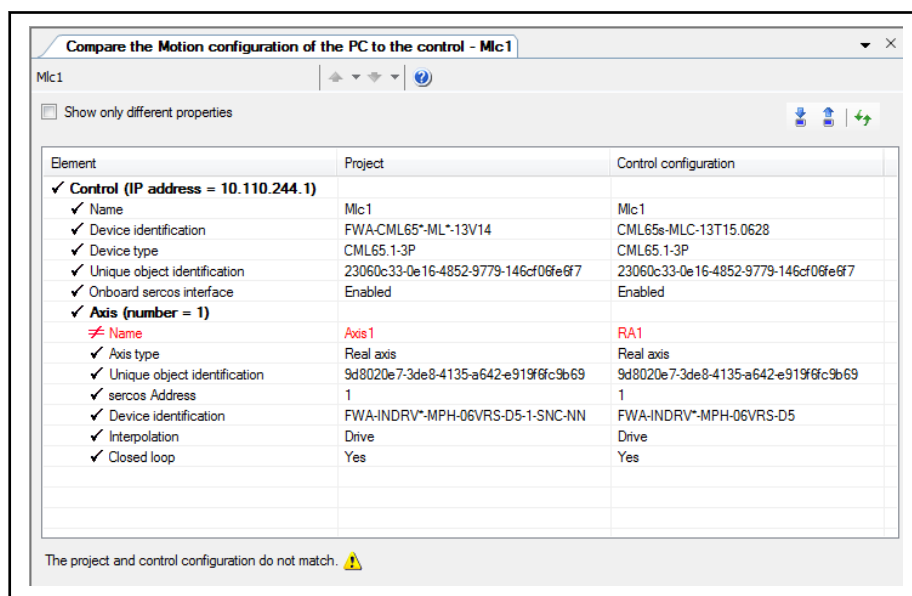


Fig. 7-19: Example: Display of the different axis name including all other properties

The property "Unique object identification" is displayed for many objects (e.g. all axes or the control). This internal property labels the identity of the object. This is comparable to a unique name. IndraWorks requires this identification

Motion functionality

to detect all changes in the configuration and to load them to the control. Even if the axis number or the Sercos address of an axis change, this identification remains unchanged. The actual value of the identification is of no significance. It is sufficient to know whether it matches between the project and the control. If the identification does not match, the object is different, even though all other properties match.



This dialog displays only data relevant for uploading and downloading the Motion configuration. Other parameters (e.g. velocity limit values of axes) or PLC data are **not** part of this dialog.

7.7 Axis modes

7.7.1 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 to the axis 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**
- While commissioning, only certain drives are to be moved. The other drives are configured and also programmed in the PLC program, but they do not yet exist physically or are currently designed.
 - 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 device upgrade is possible without software modifications
 - During the [chapter "Offline Parameterization" on page 328](#), 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

7.7.2 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 to the axis 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)

7.7.3 Notes on the use of 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.

The following conditions apply for all axes (active, parking or deactivated):

- Assigned axis number or Sercos address
- One global axis variable referencing the respective axis number has to exist for each axis in the PLC

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

Tab. 7-2: Properties

7.7.4 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 axis can assume the following states:

- 0 - Axis activated
- 1 - Axis parked
- 2 - Axis deactivated

The following states are added:

Motion functionality

- 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. Executing "MC_Power" 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".



Enabling the command value coupling is also possible if the MC_Power function block is active. Prerequisite is the PLCOpen state "Standstill" or "PowerOff".

An axis operated with an active command value decoupling has the following properties:

- Use as master axis
- Bit 10 "Actual position value is relative/referenced" is set in the parameter A-0-0021.
- Bit 14 "Ab - Control and power section ready for operation" is set in the parameter A-0-0021
- Bit 31 "Command value decoupling active" is set 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.

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 command value decoupling

The command value decoupling can be combined with the axis states "Activated", "Deactivated" and "Parked". The command value 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:

Motion functionality

Axis status	Without command value decoupling	Command value decoupling active
Active		
Parked		
Deactivated		

Tab. 7-3: 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. The command value decoupling 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:

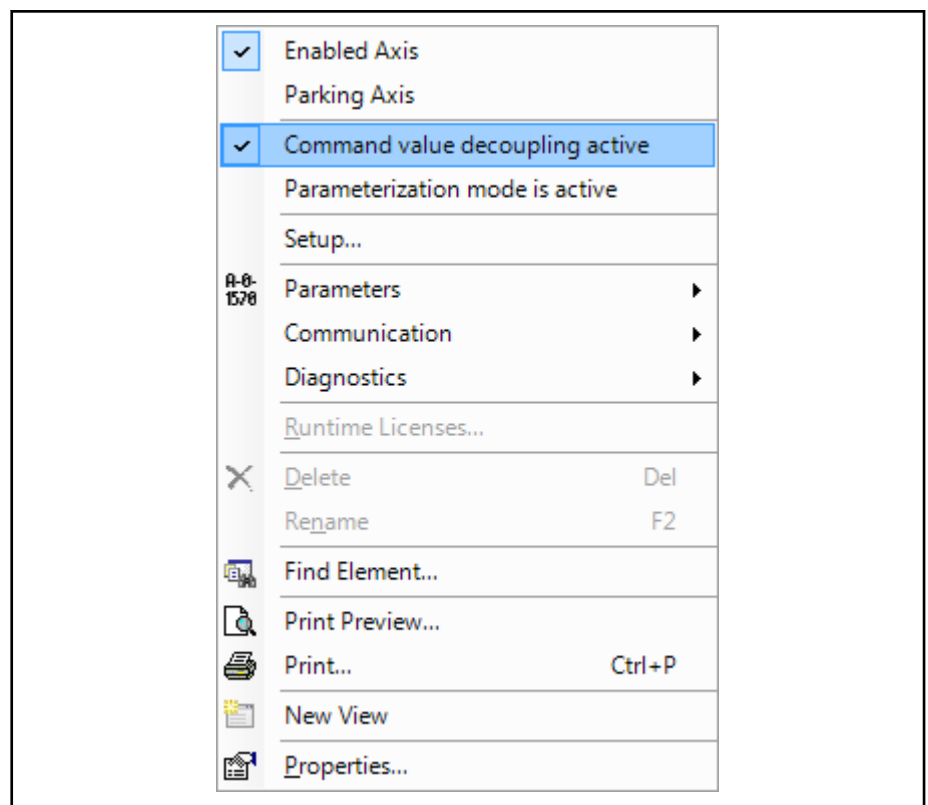


Fig. 7-20: Context menu of a Sercos drive

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):

Motion functionality

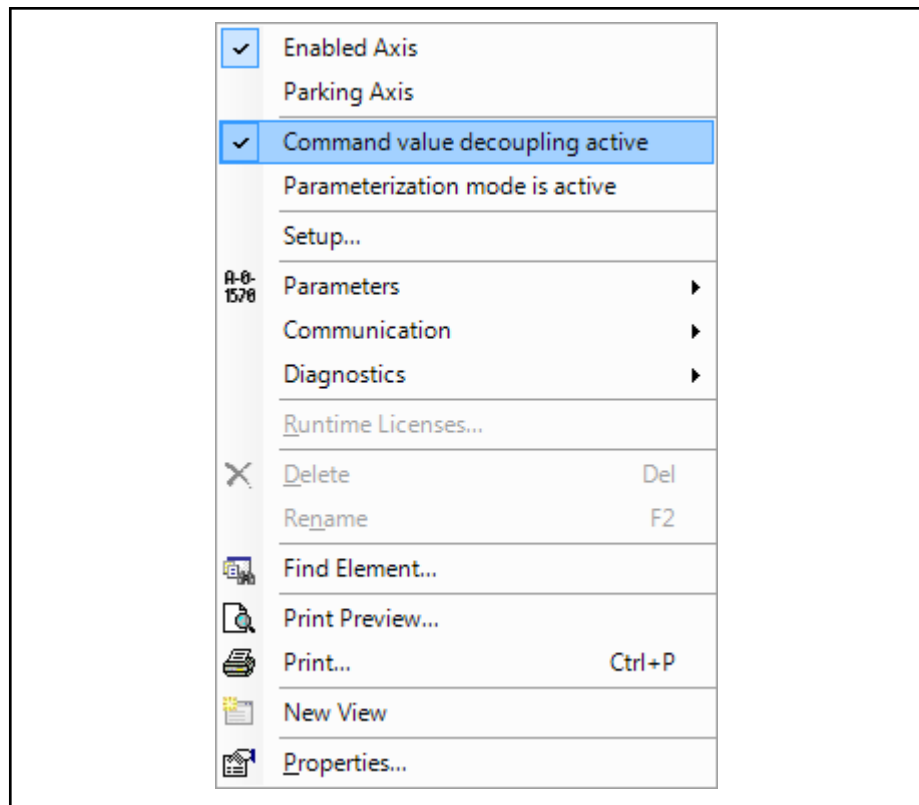


Fig. 7-21: Context menu of a IndraDrive drive



If the interpolation is executed in the drive, the menu item "Command Value Decoupling Active" is visible but cannot be selected.

7.7.5 Axis in parameterization mode

Properties Enabling the parameterization mode of an axis causes that the parameters of the axis can be changed without switching the Sercos bus to P2. This reparameterizes individual axes while the rest of the system continues running. For further information, refer to [chapter 7.5 "Parameterization level" on page 325](#).

The parameterization mode of an individual axis can be enabled or disabled via its context menu. In the Project Explorer, an enabled parameterization level is labeled by an attached "[PM]".

7.8 Single-axis modes

7.8.1 Overview

This chapter describes the various single-axis modes. A single-axis mode is always active if the axis does not follow another axis (if its motion only depends on instructions from the user or the PLC program).

The following operation modes are described in this chapter:

- [chapter 7.8.2 "Operation modes" on page 345](#)
 - Positioning: Absolute, additive or relative
The axis moves to a specific position. The position value can be specified absolutely or relatively to the command or actual position
 - Velocity control

- Continuous axis motion with defined velocity
 - Torque/force control
- Continuous axis motion with defined torque/force
 - [chapter 7.9 "Operation modes for synchronous motions with electronic gear function" on page 351](#)
 - 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

7.8.2 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 positionings 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 implemented the position 6,000 (velocity 0), its "Done" output enables "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. 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.

Motion functionality

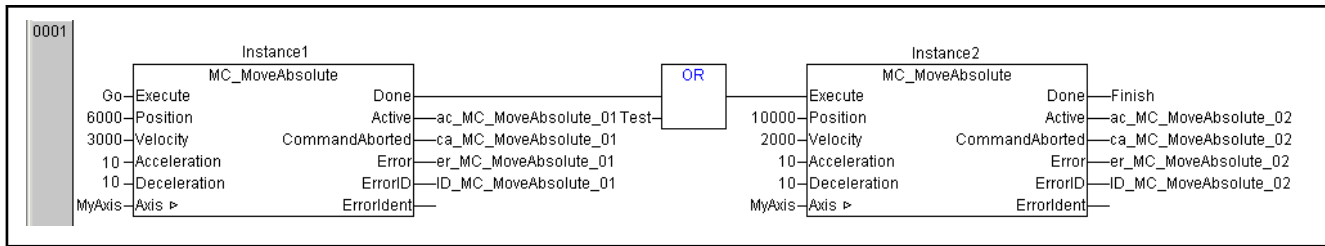


Fig. 7-22: Two instances of MC_MoveAbsolute

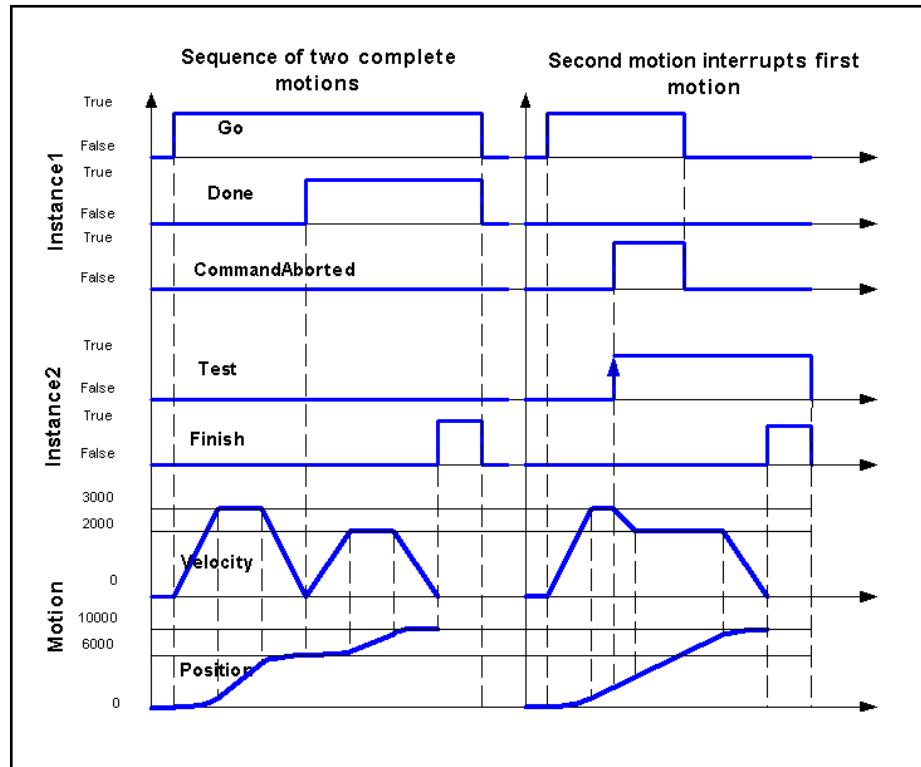


Fig. 7-23: Time diagram: Two successive motions of "absolute positioning" mode

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 "absolute and additive positioning":



Two different function block types 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 reaches the distance 6,000 (at a velocity of 0). Subsequently, the "Done"

output causes the second function block to travel the distance 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 by the "Test" signal at constant velocity. 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.

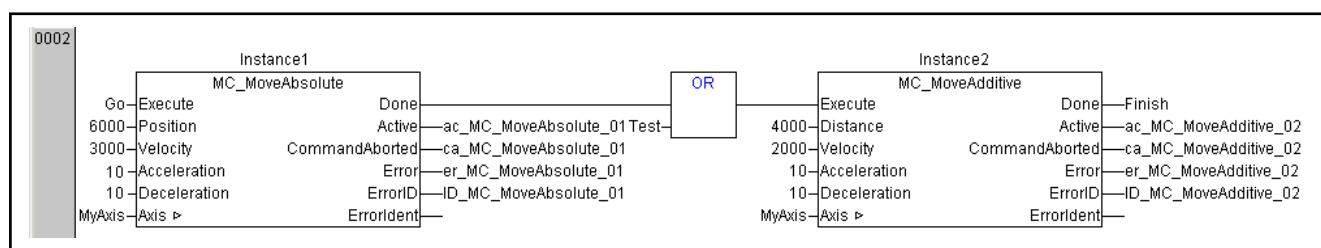


Fig. 7-24: Instances of MC_MoveAbsolute and MC_MoveAdditive

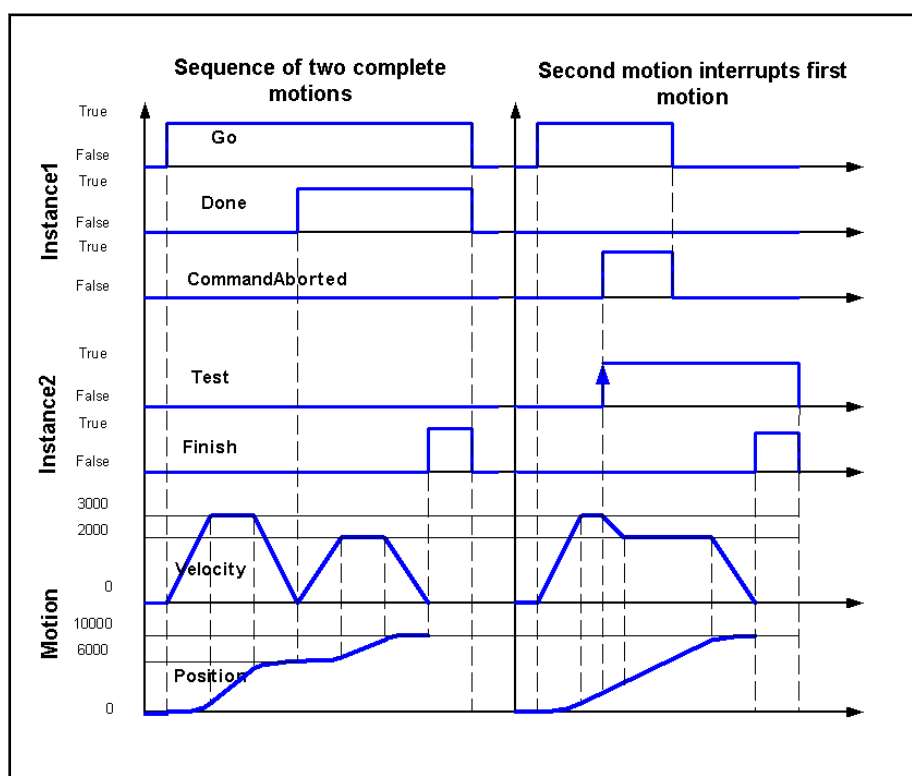


Fig. 7-25: Time diagram: Two successive motions of "absolute" and "additive positioning" modes

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

Motion functionality

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" was processed. After "Instance1" has reached the 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 side shows how "Instance2" is already activated by the "Test" signal while "Instance1" is still running. 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.

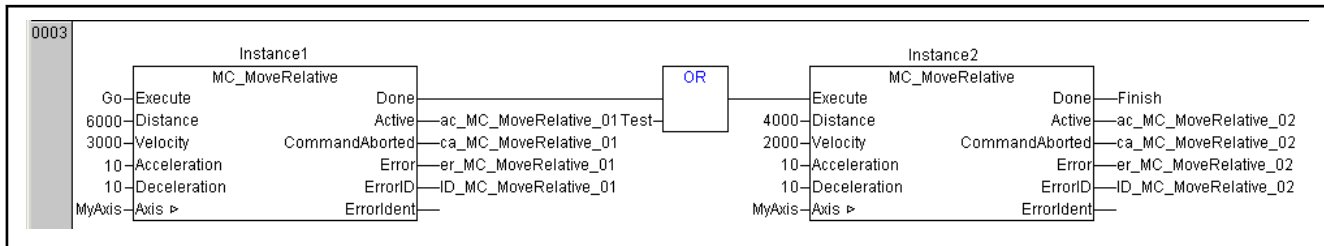


Fig. 7-26: Two instances of MC_MoveRelative

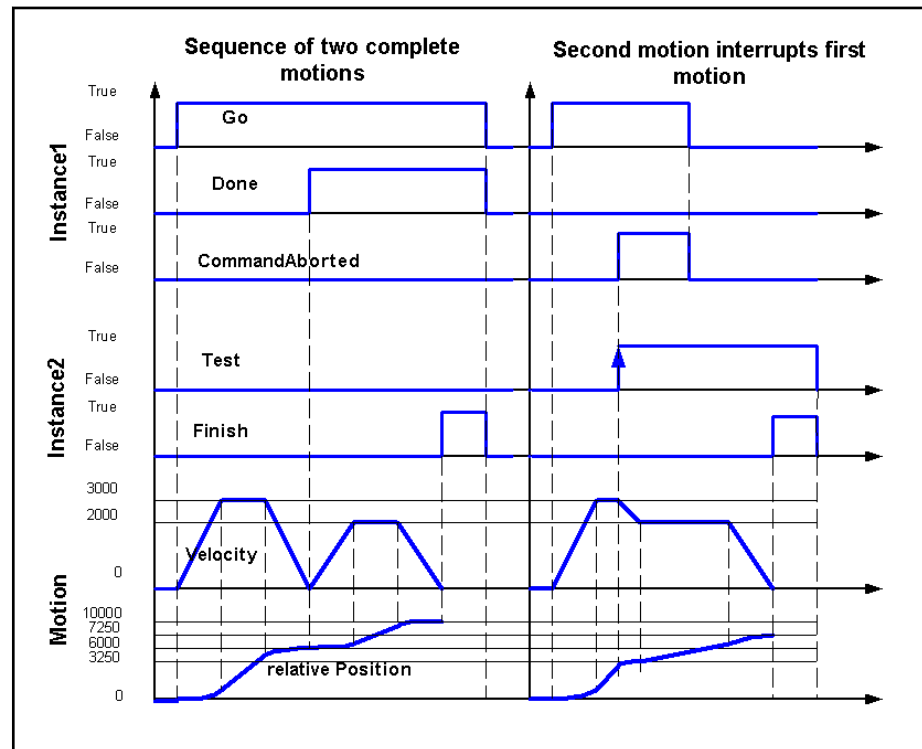


Fig. 7-27: Time diagram: Two successive motions of the "relative positioning" mode

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.

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 operation mode. 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:

Using 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 to 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 is complete. Once the first function 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 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. The second function block is now running and decelerates to a velocity of 2,000.

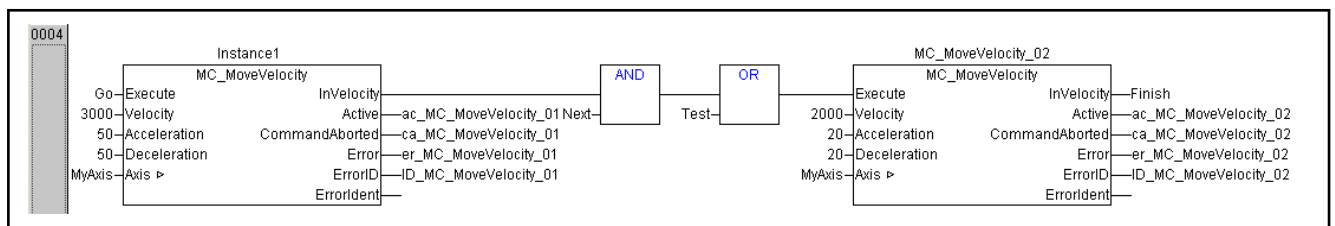


Fig. 7-28: Two instances of MC_MoveVelocity

Motion functionality

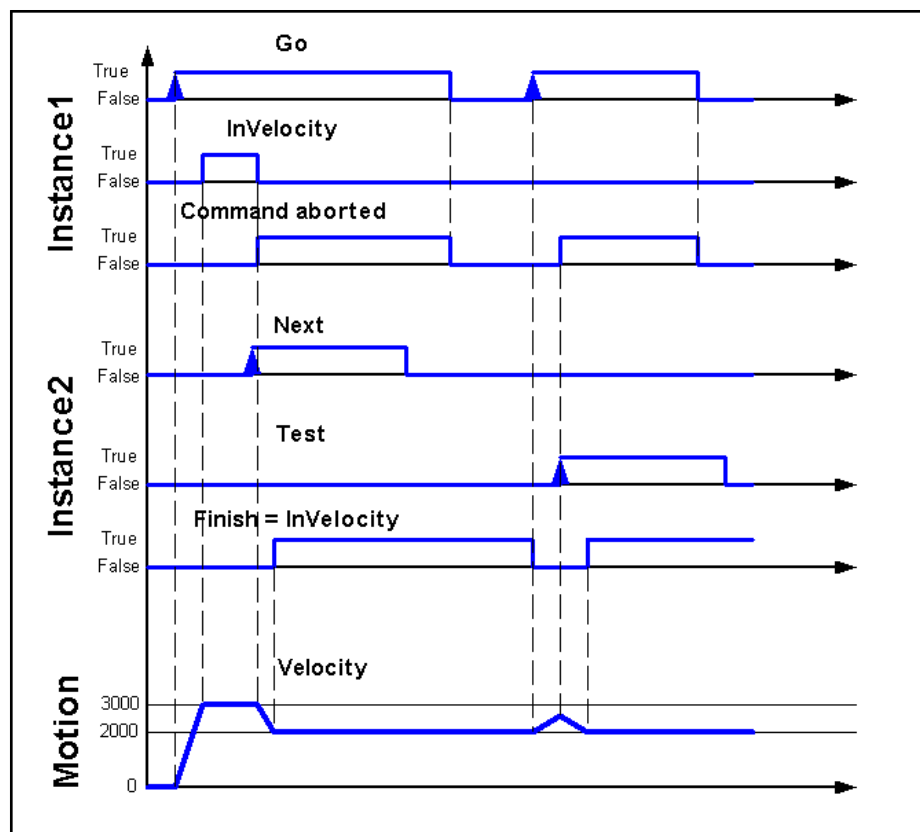


Fig. 7-29: Time diagram: Two successive motions of the "velocity control" mode

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 ("switched in parallel") 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 is complete. Once the first function block 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 torque 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 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.

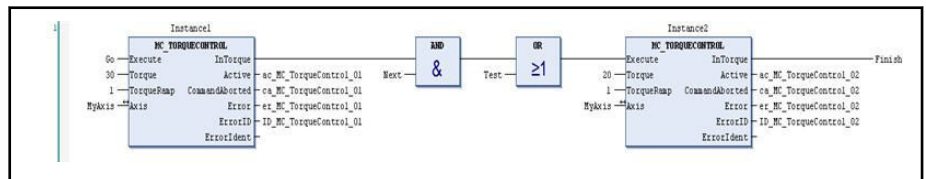


Fig. 7-30: Two instances of MC_TorqueControl

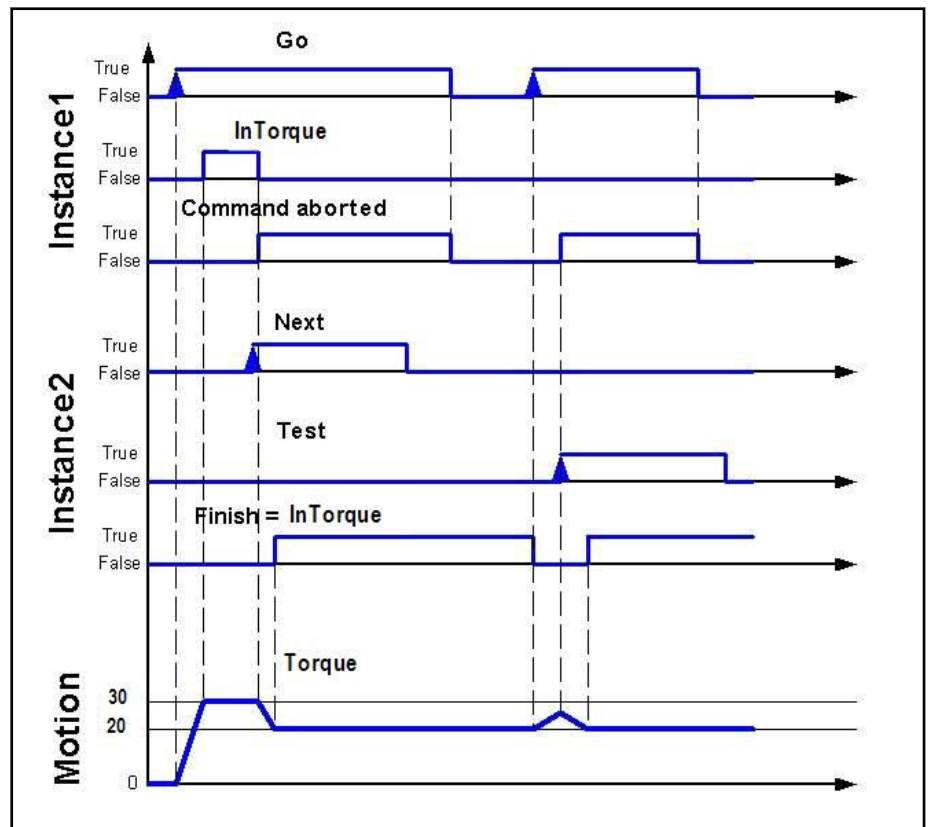


Fig. 7-31: Time diagram: Two successive motions of the "torque/force control" mode

7.9 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

Motion functionality

- Impression cylinders
- Paint application rollers
- Film/paper drawing ...

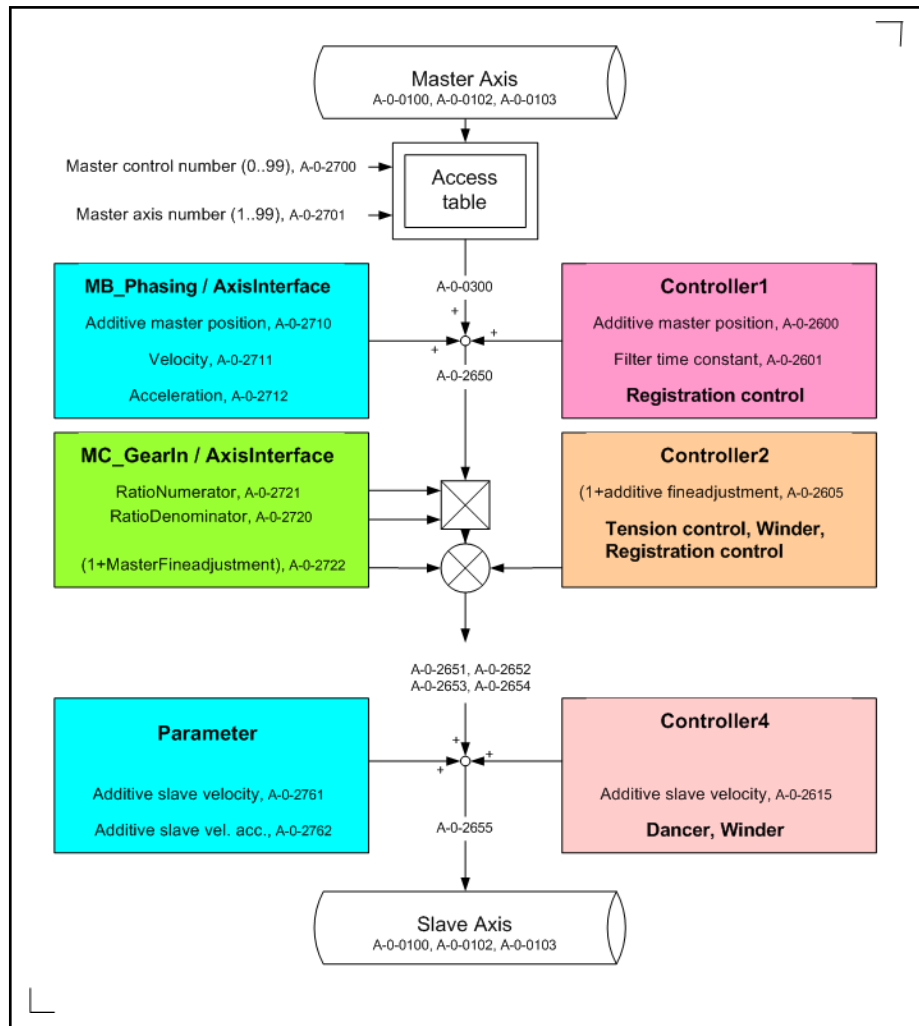


Fig. 7-32: MotionControl function chain: Velocity synchronization

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 the determination on the Sercos application profile PackProfile does not intend a velocity control/synchronization as secondary operation mode. In this case, the function block MC_GearIn or the "velocity synchronization" mode are represented in the drive via a "position control" by a control. The

Motion functionality

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:

Using 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

Tab. 7-4: 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 ...

Motion functionality

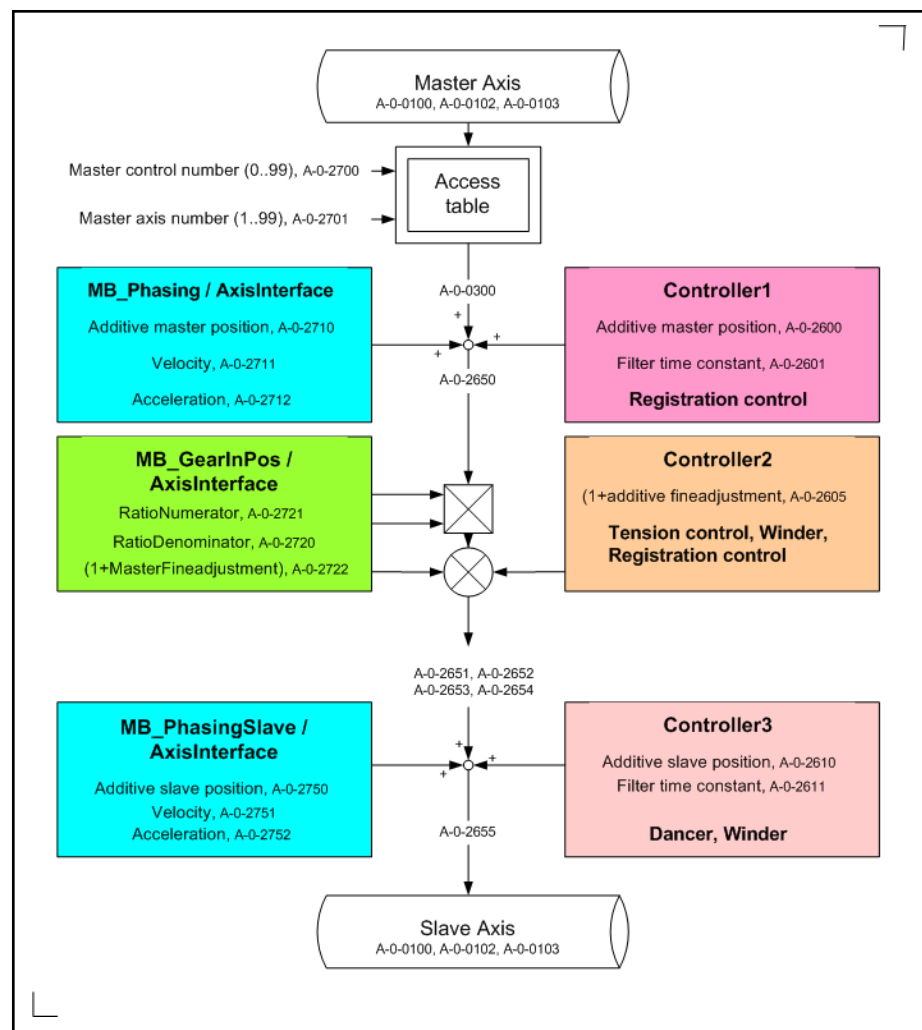


Fig. 7-33: MotionControl functional chain: Phase synchronization

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 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 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 in a cam table by points. 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.

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 IndraWorks to generate cam tables for specific applications such as cross cutting and cross sealing.



The "electronic cam" for real axes is only available for interpolation in the drive.

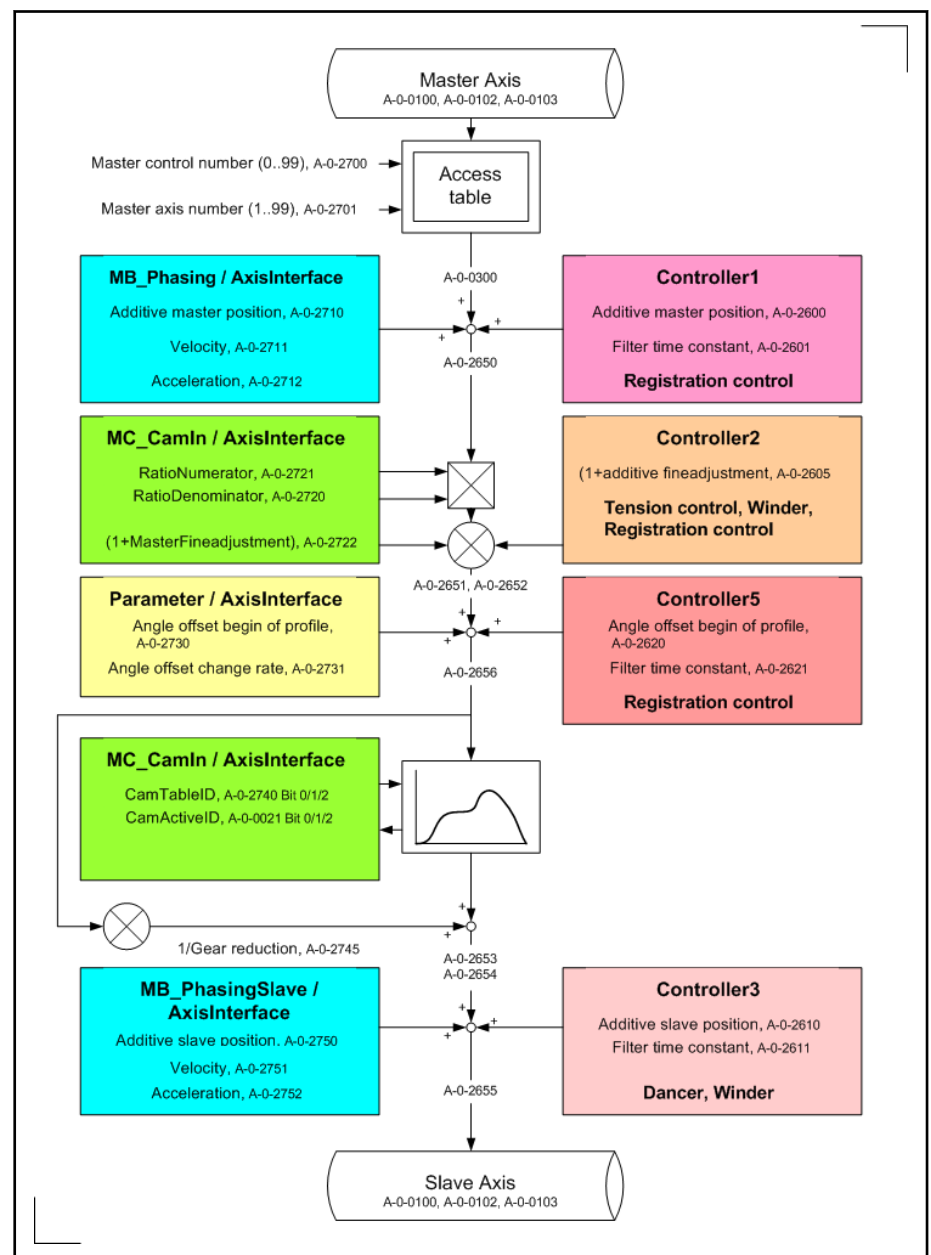


Fig. 7-34: MotionControl functional chain: Electronic cam

The operation mode provides the following access points:

Motion functionality

- 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 table 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 IndraLogic XLC are only provided for virtual axes and real axes with interpolation in the control.

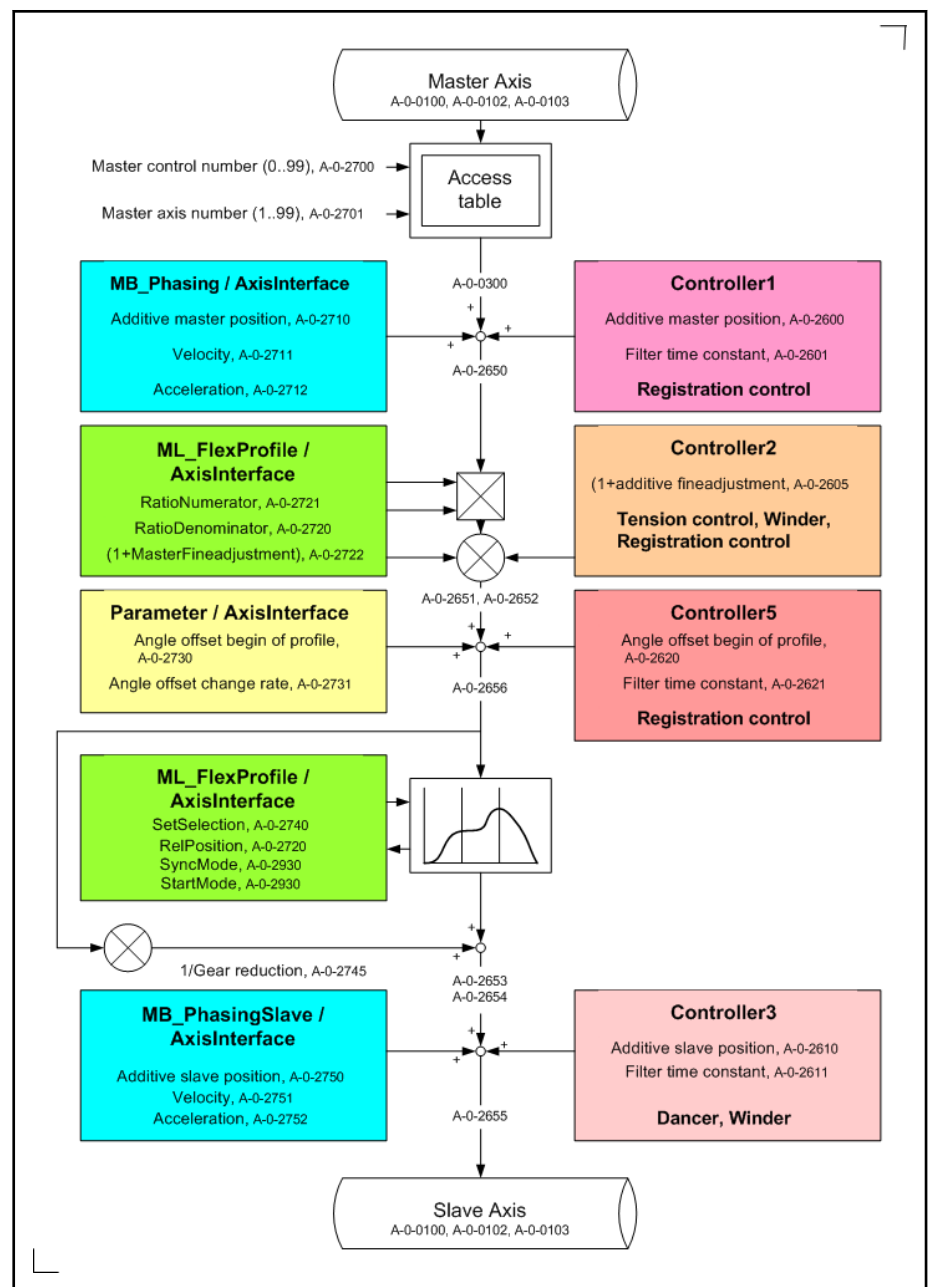


Fig. 7-35: MotionControl functional chain: Electronic Motion Profile FlexProfile

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 FlexProfiles to be saved in the control before: "A-0-2740, Cam table preselection".

These values are specified via ML_Flexprofile or the AxisInterface.

Motion functionality

- 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.

7.10 Boundary Conditions for an Axis Velocity Specification in the PLCopen State "Synchronized Motion"

The following condition applies to an axis calculated in the control (axis with interpolation in the control, virtual axis, encoder axis, controller axis, link axis) and generating a velocity command value for a coupled axis (in PLCopen state "Synchronized Motion"):

The velocity specification must not exceed the following maximum value $n_{\max}$ (Shannon sampling theorem):

$$n_{\max} = 0.5 * \text{modulo value} / T_{\text{cyc}}$$

with

T_{cyc} Motion cycle time (C-0-0410, Motion, cycle time (Tcyc), actual value, in msec),

and modulo value of the axis (A-0-0045, Modulo value).

The following applies to a module value of 360 degrees:

$$n_{\max} = (1000 \text{ rpm} * 0.5 * 60 \text{ msec}) / T_{\text{cyc}}$$

The following theoretical velocities result from the different cycle times:

Cycle time in ms	Velocity in rpm
1	30000
2	15000
4	7500
8	3750
16	1875

To ensure a safety distance to these theoretical values, a safety factor is taken into consideration.

The following values result when assuming a factor of 0.95:

- for 1 msec: 28500 rpm
- for 2 msec: 14250 rpm
- for 4 msec: 7125 rpm
- for 8 msec: 3563 rpm
- for 16 msec: 1781 rpm

The maximum velocity can also be achieved by specifying the command value to the master axis or by using gear and offset parameters between master axis/axes and slave axis/axes.

Example:

Two coupled virtual axes and one real coupled axis:

Motion functionality

For T_{cyc} 4 msec, a command value of 80 rpm to a virtual master axis VA₁ and an electronic gear calculated on the control (parameter A-0-2720, ELS, master axis gear, input revolutions, A-0-2721, ELS, master axis gear, output revolutions, for an axis calculated on the control) on the virtual axis VA2 of 100 (-> command value 8000 rpm for the coupled real axis), this condition is not complied with!

From MLC 13V12, the error F5113002 Velocity too high for cycletime is generated when the velocity is exceeded!

All axes are decelerated!

Possible remedy when reaching this boundary condition:

- Specify a lower command value or adjust the command value specification to the Motion cycle time
- Reducing the Motion cycle time
- If required, moving gears calculated in the control to the drive (in case of an axis with interpolation in the drive)

7.11 Axis commissioning

7.11.1 Overview

Use the "Axis Commissioning" dialog in the context menu of the "Motion" node to move axes without writing a PLC program. This is especially useful during commissioning if the PLC program is not complete or if an axis is to be moved differently as intended in the PLC program for test purposes.

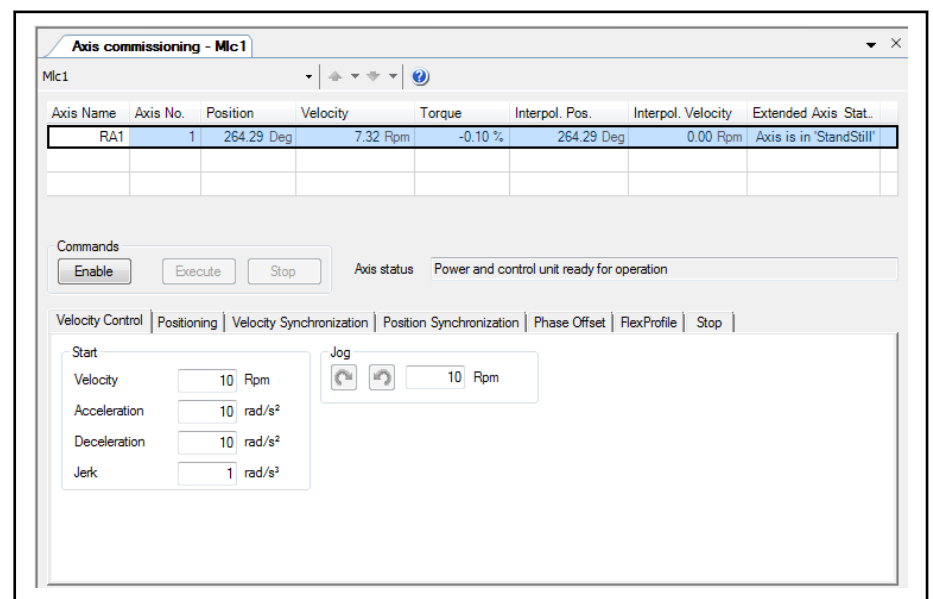


Fig. 7-36: "Axis Commissioning" dialog

The dialog is divided into an overview and a details section. The most important axis data can be displayed in the overview section. Axes to be moved can also be selected in this dialog. In the details section, motion commands can be sent to the currently selected axis. These motion commands orient at the state diagram of the PLCopen (refer to the manual "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Libraries", chapter "State Diagram of a Real Axis") and are the same as those commands created by the respective function blocks.

Motion functionality

All axis types can be displayed in the overview section of the dialog, but motion commands can only be sent to real and virtual axes. All other axis types are only displayed due to reasons of clarity.

Overview section

In this section, select the axes to be displayed and moved. Thus, click on a field in the "Axis Name" column and select the axis. To remove an axis from the overview section, select the empty entry or press <Del> after selecting a line.

The most important information on each axis is displayed in the table. More information is displayed in the tooltip when moving the mouse pointer over the corresponding axis.

Details section

All commands as well as the axis status in the details section refer to the axis selected in the overview section. Select the motion type and the corresponding command values via the tabs. All fields of a tab are only input fields and not automatically updated. If an invalid value is entered (entry is no digit for example), the input field and its tab are colored in red and an error message is displayed in the tooltip of the entry field. No motion command can be sent to the axis (corresponding buttons are disabled).

7.11.2 Safety functions

Before the actual commissioning dialog is displayed, confirm that the required measures for a safe axis operation were performed.



Fig. 7-37: Warning when opening the commissioning dialog

While the dialog is open, click on the "Stop" window to stop all axes.

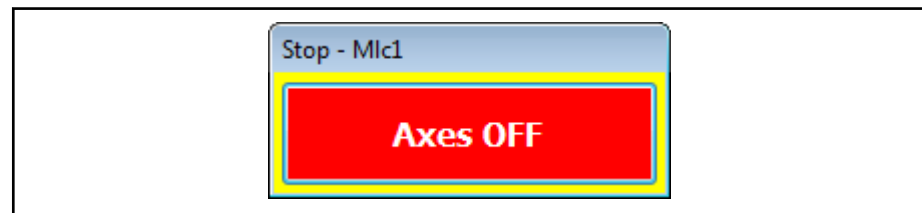


Fig. 7-38: Stopping axes



Real axes with interpolation in the drive cannot be stopped if the axes are already in the "Stopping" state. In this case, the axis is stopped with its originally set deceleration values ("stopping" the axis with changing deceleration values) is not possible.

Ensure that real axes with interpolation in the drive are not stopped with deceleration values that are too low.

Axes may only be moved as long as axes are displayed in the dialog. Thus, all axes are stopped when the dialog is closed. Furthermore, a running axis cannot be deleted or changed in the overview section.

To avoid conflicts with the PLC program, axes can only be moved if one of the following conditions apply:

- PLC is stopped
- The PLC runs and the variable "UserVarGlobal.AllowAxisCommissioning" contains the value TRUE. To move axes at a running PLC, create a Boolean variable named "AllowAxisCommissioning" in the node "UserVarGlobal". Apply this variable to the symbol configuration of the PLC and set its value to TRUE. Thus, the PLC program can control whether and when moving axes via the dialog is permitted

If axes are moved via the commissioning dialog and if the previously mentioned conditions are not fulfilled anymore, all axes are immediately stopped and further motion commands are inhibited.



When closing the dialog, a stop command is sent to all running axes and it is waited for their standstill. If an axis requires more than two seconds up to the standstill, the dialog is still closed and a warning is issued in the IndraWorks status bar. In this case, the axis is still stopped, but remains in the "Stopping" state. To finally stop an axis ("Standstill" state), open the dialog at any time and issue the "Stop" command again to the axis. Alternatively, the axis can also be stopped via the PLC.



Response when connection is interrupted between the IndraWorks interface and the control

If the connection between PC and control is interrupted (e.g. network cable defective or removed), the control outputs the error "F536000E MLPI: Watchdog error". This stops all commanding axes automatically.

7.11.3 Enabling an axis

To move a real or a controller axis, enable the axis before explicitly. The enabling functionality corresponds to the one of the "MC_Power" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCOpen Library"). An enabled axis is always highlighted in green in the overview and details section. If the button is pressed again, enabling is canceled.

To move a real axis, enable it explicitly before. The enabling functionality corresponds to the one of the "MC_Power" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCOpen Library"). An enabled axis is always highlighted in green in the overview and details section. If the button is pressed again, enabling is canceled.

To enable an axis, the following prerequisites have to be met:

Motion functionality

- Sercos bus is in P4 (operating mode)
- Axis is not in P2 (parameterization mode)
- Control and power section are ready for operation (bit 14 is set in the parameter A-0-0021)
- No error is present at the axis

If one of the mentioned prerequisites is not met, the **Enable** button is disabled and the missing prerequisites are shown as tooltip next to the **Enable** button.

Virtual axes are automatically enabled and thus do not have to be enabled explicitly by the user (they can receive a Motion command immediately).

7.11.4 Velocity control

An enabled axis can be moved at a constant velocity in the "Velocity Control" tab. Enter velocity, acceleration and jerk values into the corresponding fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the one of the "MC_MoveVelocity" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
Start/Velocity	Velocity
Start/Acceleration	Acceleration
Start/Deceleration	Deceleration
Start/Jerk	Corresponds to the content of the parameter A-0-0216 when starting the command

Tab. 7-5: Input fields in the dialog on the inputs of the MC_MoveVelocity function block

The "Jog" section can be used to move the axis only when the corresponding button is pressed. The velocity command value from the "Jog" section is used. The acceleration and jerk values originate from the "Start" section.

7.11.5 Positioning

An enabled axis can be moved to a position in the "Positioning" tab. Enter position, velocity, acceleration and jerk values into the corresponding fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the functionality of the "MC_MoveAbsolute" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
Start/Position	Position
Start/Velocity	Velocity
Start/Acceleration	Acceleration
Start/Deceleration	Deceleration
Start/Jerk	Corresponds to the content of the parameter A-0-0216 when starting the command

Tab. 7-6: Input fields in the dialog on the inputs of the MC_MoveAbsolute function block

Alternatively, use the "Jog" section to move the axis along a relative position. This function corresponds to the function of the function block "MC_MoveAbsolute" (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCOpen Library"). The incremental position command value from the "Jog" section is used. The acceleration, velocity and jerk values originate from the "Start" section.

The button **Position data reference** is used to determine the zero point of real axes. The Sercos bus has to be in phase 4, the axis may not be connected to power and no error may be present for the axis.

7.11.6 Velocity synchronization

In the "Velocity Synchronization" tab can be specified whether the velocity of an enabled axis is synchronized to the velocity of another axis. Enter master axis and gear settings into the input fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the functionality of the "MC_GearIn" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCOpen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
"Master Axis" section	Master
Gear/Input revolutions	RatioDenominator
Gear/Output revolutions	RatioNumerator
Gear/Fine adjustment	MasterFineadjust

Tab. 7-7: Input fields in the dialog on the inputs of the MC_GearIn function block

7.11.7 Position synchronization

In the "Position Synchronization" tab can be specified whether the position of an enabled axis is synchronized to the position of another axis. Enter master axis, gear and synchronization settings into the corresponding fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the functionality of the "MB_GearInPos" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCOpen Library"). The input fields are analogously significant to the input parameters of the function block:

Motion functionality

Input field in the dialog	Input parameters of the function block
"Master Axis" section	Master
Gear/Input revolutions	RatioDenominator
Gear/Output revolutions	RatioNumerator
Gear/Fine adjustment	MasterFineadjust
Synchronization/direction	SyncMode
Synchronization/starting mode	StartMode

Tab. 7-8: *Input fields in the dialog on the inputs of the MB_GearInPos function block*

7.11.8 Phase offset

An additional phase offset between two synchronized axes can be set in the "Phase Offset" tab. Enter phase, velocity and acceleration values into the corresponding fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the functionality of the function blocks "MB_Phasing" and ("MB_PhasingSlave" refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
Start/Phase offset	Position
Start/Velocity	Velocity
Start/Acceleration	Acceleration
Start/Deceleration	Deceleration
Effect on the position of master axis/slave axis	Function analog to the function blocks MB_Phasing/MB_PhasingSlave

Tab. 7-9: *Input fields in the dialog on the inputs of the function blocks MB_Phasing and MB_PhasingSlave*

7.11.9 FlexProfile

An enabled axis can be synchronized to another axis via a cam in the "Flex-Profile" tab. Enter the command values into the input fields and press **Execute**. If the button is pressed again, the command values are again transferred to the axis. The axis does not have to be stopped.

The functionality corresponds to the functionality of the "ML_FlexProfile" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
FlexProfile/Number	SetSelection
"Master Axis" section	Master
Starting mode and offset/ Master axis	ProfileEntry and MasterOffset

Motion functionality

Starting mode and offset/ Slave axis	ProfileEntry and SlaveOffset
Gear/Input revolutions	RatioDenominator
Gear/Output revolutions	RatioNumerator
Gear/Fine adjustment	MasterFineadjust
Starting options/Processing type	ExecutionMode
Starting options/Switching position	UseSwitchingPosition and SwitchingPosition
Synchronization	SyncType
Synchronization/Velocity	SyncVelocity
Synchronization/Acceleration	SyncAcceleration

Tab. 7-10: Input fields in the dialog on the inputs of the *ML_FlexProfile* function block

7.11.10 Stop

An enabled axis can be stopped in the "Stop" tab. Enter deceleration and jerk values into the input fields and press **Execute**.

The command values in this tab are also used when pressing the **Stop** button. Which tab is selected is of no importance.



A changed deceleration value for the stop is not applied to real axes with interpolation in the drive if the stop was already commanded with a different value.

A new value can only be commanded after reaching the standstill message of the axis.

If the axis has to be stopped immediately with a changed deceleration value at active stop, stop the axis using E-Stop.

The functionality corresponds to the functionality of the "MC_Stop" function block (refer to the documentation "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Library"). The input fields are analogously significant to the input parameters of the function block:

Input field in the dialog	Input parameters of the function block
Stop/Deceleration	Deceleration
Stop/Jerk	Corresponds to the content of the parameter A-0-0216 when starting the command

Tab. 7-11: Input fields in the dialog on the inputs of the "MC_Stop" function block

7.12 Validating Motion configuration

7.12.1 Overview

During the Motion configuration download, when switching online or switching to the offline parameterization mode, IndraWorks checks the entered Motion configuration automatically for projection errors. The validation result 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 com-

Motion functionality

plete action. Otherwise, it is switched to online or offline parameterization mode. Subsequently, it is validated online.

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 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. 7-13 "IndraWorks menu item "Tools" -> "Options" -> Switch Online/Offline" on page 336](#)).

For a better identification, a unique name is assigned to each error and warning message in the task list. Use this name to search for the message in the online help for example. The name 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.

7.12.2 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 may not be a digit. Additionally, two underscores may not directly follow each other and the last character 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 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 axis name. 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 identical to an IndraLogic keyword was specified (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

Motion functionality

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 configured axes (<Digit1>) is higher than the maximum number of axes allowed (<Digit2>)

Cause The number of configured axes exceeds the number the control is able to process. The maximum number of supported axes is written in <Digit2>.

- Solution**
- Delete the surplus axes
 - Convert the control into a control type of higher performance

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 into 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> characters)

Cause The specified axis name is too long. The maximum length is written in <Digit>.

- Solution**
- Enter a shorter name for the axis

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 respective axes. Delete the axes and import them again. When switching online the next time, 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 any encoder axes.

Cause The encoder axis <Name1> was assigned to the real axis <Name2>, but these axes do 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-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 the configured drives
 - Open the properties of the drive and change the control section
 - Delete the drive

MLs-E0A0024: The firmware <Name1> of the axis <Name2> is invalid

Cause A drive was configured. The drive firmware is not supported by the control, as an old project was converted or drives were imported from an old import file for example.

- Solution**
- Open the properties of the axis and change the firmware of the drive to a supported version
 - Delete the axis and add a drive of the "SercosDrive" type instead

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 to a 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 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 upon next Motion configuration download.

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

Motion functionality

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 the type "IndraMotion MLP VEP" or "IndraLogic XLC VEP". These controls are not supported by IndraWorks anymore for 12VRS or higher.
- Solution**
- Convert the control into a supported control type
 - Delete the control

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 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> characters)

- Cause** The specified control name is too long. The maximum length is written in <Digit>.
- Solution**
- Enter a shorter name for the control

Function module error messages

MLs-E0F0010: The number of configured function modules (<Digit1>) is greater than the maximum permissible number of function modules (<Digit2>)

Cause The number of configured function modules exceeds the number the control is able to process. The maximum number of supported function modules is written in <Digit2> .

Solution

- Delete the surplus function modules
- Convert the control into a control type of higher performance

MLs-E0F0011: More than the valid number (<Digit>) of function modules of type <Name> were configured

Cause More function modules of type "name" were configured 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-E0F0013: The function module <Name> on the slot <Digit> has to be plugged in as last function module

Cause More function modules were plugged in after plugging in the function module to be plugged in last. This is physically not possible, since no further function module can be plugged in at the left side of the function module. For more detailed information on the special features, refer to the description of the function module.

Solution

- Move the function module to the last slot
- Delete the following function modules

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 configured.

Solution

- Delete the function module
- Convert the control to a control type supporting the function module

Sercos device error messages

MLs-E0S0010: The number of configured Sercos devices (<Digit1>) is greater than the maximum permissible number of Sercos devices (<Digit2>)

Cause An invalid 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.

Motion functionality

- 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 participating 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 found in multiple Sercos I/O devices in the internal IndraWorks data storage. In practice, such a configuration is not possible. It can result if several Sercos devices are imported from another project.

- Solution**
- Open the "Sercos Configuration" dialog and automatically assign the topology addresses via "Correct Topology"

MLs-E0S0022: The I/O device <Name> does not match the configured Sercos protocol

- Cause** A Sercos III I/O device was configured, 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

Safety error messages

MLs-E0Y0010: The number of gateways "SLC-3-GS3S" (<Digit1>) is greater than the maximum permissible number (<Digit2>)

- Cause** The control cannot process the number of gateways of type "SLC-3-GS3S" that were configured. The maximum number of supported gateways is written in <Digit2> .

- Solution**
- Delete the surplus gateways of type "SLC-3-GS3S"

MLs-E0Y0011: The number of "CIP Safety on Sercos" targets (<Digit1>) is greater than the maximum permissible number of "CIP Safety on Sercos" targets (<Digit2>)

- Cause** More "CIP Safety on Sercos" targets - than the gateways are able to process - were configured for a gateway of type "SLC-3-GS3S". The maximum number of supported "CIP Safety on Sercos" targets is written in <Digit2>. This can be the case if axes from different project were imported into one project.

- Solution**
- Delete the surplus "CIP Safety on Sercos" targets via the context menu of the gateway for example

MLs-E0Y0012: The target number <Digit1> is outside the valid range (1..<Digit2>)

Cause "CIP Safety on Sercos" was configured for a real axis with a target number outside the valid value range. The maximum target number allowed is written in <Digit2>. This can only be the case if an axis - whose import files was manually modified before - is imported.

Solution • Change the target number via the context menu of the axis for example

MLs-E0Y0013: The target number <Digit> was specified multiple times

Cause The same target number was configured for the gateway of type "SLC-3-GS3S". This is invalid. The target number has to be unique in a gateway. This can be the case if axes from different project were imported into one project.

Solution • Assign a unique target number (e.g. via the context menu of the gateway) to all "CIP Safety on Sercos" targets
• Delete the "CIP Safety on Sercos" targets whose target number is not unique

MLs-E0Y0014: "CIP Safety on Sercos" was configured for the axis, even though it is not supported by the axis

Cause "CIP Safety on Sercos" was configured for this axis, even though it does not support safety engineering. This can be the case if an old project - in which the axis still supports the safety engineering - is applied.

Solution • Change the firmware of the axis supporting the safety engineering
• Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

MLs-E0Y0015: The axis is assigned to the gateway "SLC-3-GS3S" which does not exist anymore. The gateway name was <Name>, the Sercos address was <Digit>

Cause An axis was assigned to a gateway of type "SLC-3-GS3S" which does not exist anymore, since the gateway was deleted in the meantime for example. The old name and the Sercos address of the deleted gateway are specified in <Name> and <Digit> and are only used for a better understanding of the original configuration.

Solution • Assign an existing gateway of type "SLC-3-GS3S" to the axis
• Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

MLs-E0Y0100: "CIP Safety on Sercos with SafeLogic" was configured for the axis, but the required function module "Safety CPU" is missing

Cause "CIP Safety on Sercos with SafeLogic" was configured for an axis, but no "Safety CPU" function module is present below the control. However, the axis cannot use the selected safety function without this function module.

Solution • Add a "Safety CPU" function module to the control
• Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

MLs-E0Y0101: "CIP Safety on Sercos with SafeLogic" was configured for the axis, but no Safety profile was selected

Cause "CIP Safety on Sercos with SafeLogic" was configured for an axis, but no Safety profile is selected.

Motion functionality

- Solution**
- Select a Safety profile for the axis
 - Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

MLs-E0Y0102: The Safety profile <Name> selected for the axis is invalid

- Cause** "CIP Safety on Sercos with SafeLogic" was configured for an axis, but its Safety profile is invalid. This can be the case if a new device description file was installed for the corresponding drive and this device description does not contain the earlier selected Safety profile anymore.

- Solution**
- Select a Safety profile again for the axis
 - Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

7.12.3 Information messages

General information

Information messages provide notes on the validation. Information messages 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. Double-click to display details

- Cause** An internal error occurred during validation impeding some checks to be performed. This information indicates a potential problem with unknown effects. Report error details to the Bosch Rexroth Service.

- Solution**
- Report the error details to the Bosch Rexroth Service

7.12.4 Warning messages

General information

Warning messages indicate a potential problem in the Motion configuration. However, it can still be switched online or the Motion configuration can be loaded to the control.

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 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 configured encoder axes

MLs-W0A0021: The configured 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 configured 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 downloading the Motion configuration

Normally, the axis name is updated in the control when downloading the Motion configuration. 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-W0A0200: The configured axis is a axis different than the axis configured in the control (name: <Name>, axis number: <Digit>)

- Cause** It was logged in to the PLC, but the axis configured in the control does not match the axis configured in IndraWorks. Thus, the PLC program could move different axes as actually intended. This can be the case if it is logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to update the Motion configuration of the control

MLs-W0A0201: The configured axis is not available in the control

- Cause** It was logged in to the PLC and the axis configured in IndraWorks could not be found in the control. Thus, the PLC program could move different axes as actually intended. This can be the case if it is logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to update the Motion configuration of the control

MLs-W0A0202: An axis <Name> (axis number <Digit>) is in the control that is not in the project

- Cause** It was logged in to the PLC and the axis configured in the control is not configured in IndraWorks. Thus, the PLC program could move different axes as actually intended. This can be the case if it is logged in to the PLC without downloading the Motion configuration to the control before.

Motion functionality

- Solution**
- Load the Motion configuration to the control to update the Motion configuration of the control

Control warning messages

MLs-W0C0020: A higher device version (<Version>) is available for the control

- Cause** A project was opened 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

MLs-W0C0021: The connected control has a lower firmware release (<Firmware>) as configured (<Firmware>)

- Cause** Connected to a control obtaining a lower firmware release that configured in the properties of the control. Undesired effects can be caused during the operation, since it is possible that the control does not support all IndraWorks functions.

- Solution**
- Update the device in the properties of the control
 - Load a newer firmware to the control

MLs-W0C0022: Not all IndraWorks online functions are available, as the configuration of the IndraWorks project differs from the control configuration

- Cause** It is connected to a control whose Motion configuration differs from the configuration saved in the IndraWorks project. There are multiple possible reasons. An axis might have been added to the PC or the IndraWorks project is not suitable for the control. In this case, IndraWorks allows only the following operations:

- Phase switching via the toolbar and the context menu of the Sercos node
- Start of the parameter editor of the control
- Start of the control oscilloscope
- Start of the parameter import and export of the control
- View of the communication settings, advanced properties, invalid parameters and tasks of the control
- Setting the language of the control and the connected drives.

- Solution**
- Load the Motion configuration from the PC to the control
 - Load the Motion configuration from the control to the PC
 - Adapt the Motion configuration in the IndraWorks project to the configuration of the control. A detailed list of differences can be seen under "Synchronize" context menu of the control
 - Perform the online operations required and switch offline again

MLs-W0C0023: The hardware version <Value> of the connected control is syntactically incorrect

- Cause** The connected control has a hardware version, i.e. the value of the parameter C-0-0010, with an invalid value. Thus, the configuration of the control cannot be correctly compared to the configuration in IndraWorks and it is only possible to switch partially synchronized to online mode.

- Solution**
- Contact the Bosch Rexroth Service for a corrected hardware version of the control.

MLs-W0C0200: The Motion project in the control (<Name>) does not match the Motion project in IndraWorks

- Cause** It is logged in to the PLC with a project, but the Motion project was not loaded to the control. A Motion project is required for example to operate I/O devices on the Sercos bus or to see the control parameters.
- Solution**
- Load the Motion configuration to the control This function can be started either via the toolbar or the "Synchronize" menu item in the context menu of the control

Sercos device warning messages

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.
- Attention: This warning refers to the actually connected devices and **not** to the devices configured in IndraWorks.
- Solution**
- Set unique Sercos addresses for the devices. 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 is 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**
- Configure the device in IndraWorks and download the Motion configuration to the control
 - Remove the device from the control

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 task is set, but no multiple of the Sercos cycle time. Thus, the Sercos bus cannot be switched to BB.
- Solution**
- Correct the values 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 "Cycle time" dialog, which can be opened via the context menu of the "Motion" node

MLs-W0S0024: The device with the Sercos address <Digit> is configured in the control, but not connected

- Cause** A Sercos device is configured as active in the control, but not connected to the Sercos bus. Thus, the Sercos bus cannot be switched to BB.

Motion functionality

- Solution**
- Disable the device in IndraWorks and load the Motion configuration to the control
 - Connect the device to the control

MLs-W0S0200: For the I/O device (Sercos address <Digit>), the device identification in the project (<Name1>) does not match the device identification (<Name2>) configured in the control

- Cause** It is logged in to the PLC, but the device identification of the I/O device (parameter C-0-0478) configured in the control does not match the device configuration of the I/O device configured in IndraWorks. Thus, the PLC program cannot access this I/O device. This can be the case if I/O devices are changed below the "Sercos" node and logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to apply the changed I/O configuration to the control

MLs-W0S0201: The configured I/O device (Sercos address <Digit>) is not configured in the control

- Cause** It is logged in to the PLC, but the specified I/O device is not configured in the control (parameter C-0-0478). Thus, the PLC program cannot access this I/O device. This can be the case if I/O devices are added below the "Sercos" node and logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to apply the changed I/O configuration to the control

MLs-W0S0202: In the control, an I/O device (device identification <Name>) is configured to the Sercos address <Digit>, but it is not in the project

- Cause** It is logged in to the PLC, but the I/O device configured in the control (parameter C-0-0478) is not configured in IndraWorks. Thus, the PLC program cannot access this I/O device. This can be the case if I/O devices are deleted below the "Sercos" node and logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to apply the changed I/O configuration to the control

MLs-W0S0203: For the I/O device (Sercos address <Digit>), the activation in the project does not match the activation in the control

- Cause** It is logged in to the PLC, but the activation of the I/O device (parameter C-0-0478) configured in the control does not match the activation of the I/O device configured in IndraWorks. Thus, under certain circumstances, the PLC program cannot access this I/O device. This can be the case if the activation of an I/O device is changed below the "Sercos" node and logged in to the PLC without downloading the Motion configuration to the control before.

- Solution**
- Load the Motion configuration to the control to apply the changed I/O configuration to the control

Non-object-related warning messages

MLs-W0X0001: A function that is not released was enabled. This can end the warranty

- Cause** After contacting the customer support, you enabled IndraWorks functions that were not yet completely developed and tested. These function are useful in specific cases (e.g. testing). However, unexpected effects might result. Thus,

use these functions only after contacting the customer support and disable these functions again, once they are not used anymore. To disable the functions, double-click on this message in the message window.

- Solution**
- Disable the non-released functions as soon as they are not used anymore. Close and start IndraWorks again.

Safety warning messages

W0Y0010: "CIP Safety on Sercos" was configured for the axis, but the actually connected drive does not support this functionality

Cause "CIP Safety on Sercos" was configured for an axis, but the control section of the assigned drive does not contain a corresponding component. Thus, the safety functions cannot be used.

- Solution**
- Connect the drive whose control section contains the corresponding safety component
 - Delete the selection "CIP Safety on Sercos" for the axis (e.g. via its context menu)

W0Y0011: No "CIP Safety on Sercos" target was assigned to the gateway "SLC-3-GS3S"

Cause A "SLC-3-GS3S" gateway was configured, but no "CIP Safety on Sercos" target was assigned to this gateway. Thus, the safety functions of the gateway cannot be used.

- Solution**
- Assign at least one "CIP Safety on Sercos" target to the gateway (e.g. via the context menu of the gateway)
 - Delete the gateway if it is not required

7.13 Electronic cams

7.13.1 General information

Cams are used wherever complex, nonlinear motion takes place or where rotary and translatory 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. The reference signal is usually an angle of rotation " ϕ ". Using a cam, this angle of rotation is compiled into a new slave axis motion with the angle of rotation " ψ ". The figure below shows a simplified representation.

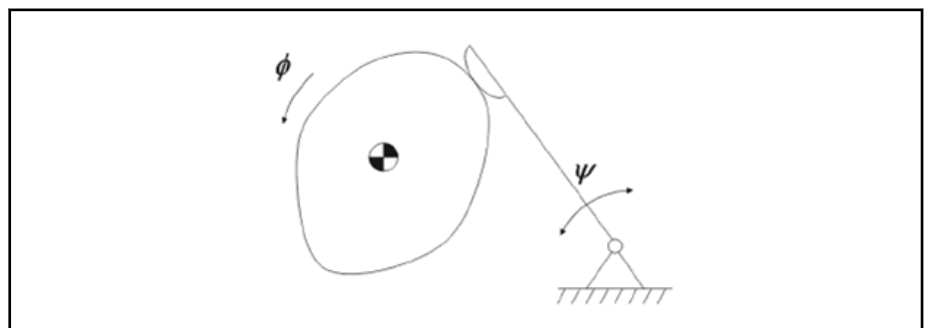


Fig. 7-39: Block diagram of a general cam

In general, two cam solutions can be defined. These are the mechanical and the electronic cam solution. The purely mechanical cams are not discussed in detail. An electronic cam can be constructed either as cam table or analytically as polynomial profiles (FlexProfile, MotionProfile).

Motion functionality

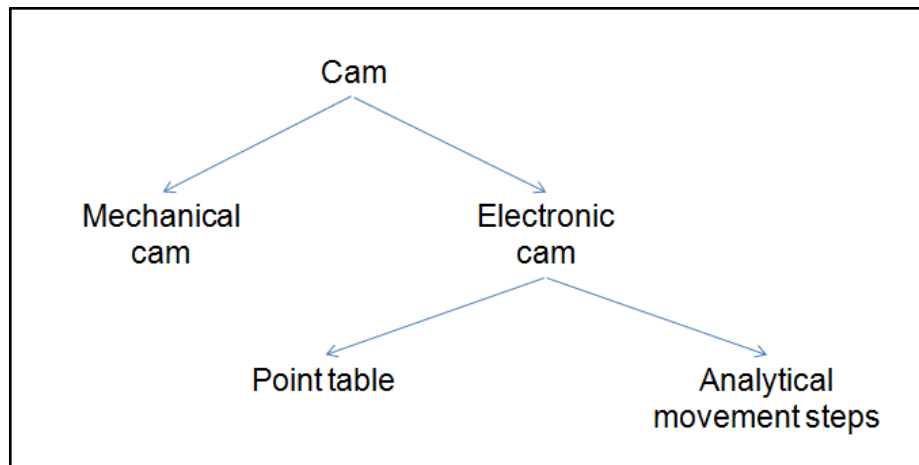


Fig. 7-40: Basic cam solutions

The FlexProfile parameterization is executed via the dialog of the respective axis (see e.g. [chapter 6.5.22 "FlexProfile" on page 275](#)).

An exemplary use case for the FlexProfile is described in "Rexroth IndraLogic XLC 13VRS First Steps" (refer to [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

7.13.2 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 an interpolation between the corresponding data points of the current interval. The slave axis position is defined by the master axis position. Various interpolation procedures can be used depending on the application.

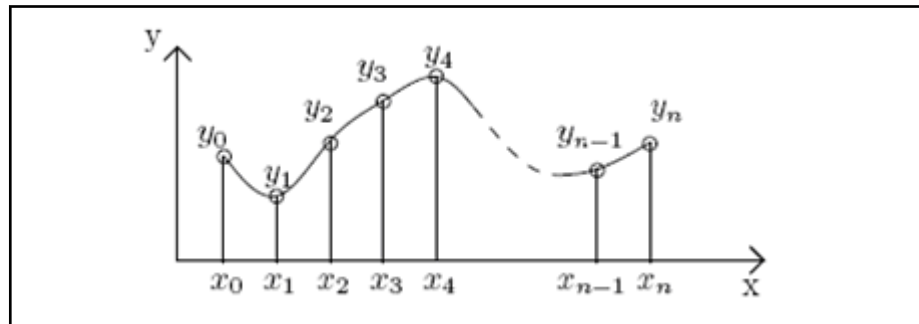


Fig. 7-41: Cam table with cubic compensation spline

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. To dynamically generate better curves, Hermite spline functions can be used while the cubic interpolation splines mentioned above are based on polynomials of 5th order. These splines execute a curve approximation resulting in a smoother curve. Irrespective of the interpolation procedure, cams are very often described like this with regard to the 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 motions 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.

When using or creating cam tables used in the control, note the following:

The control firmware scans the cam table in each motion cycle. The motion distance from scanning to scanning point is computed using the shortest distance.

If the cam table is provided with a motion of $> \text{Modulo}/2$ from scanning to scanning point (corresponds to $> 50\%$ in the cam table), the motion is $< \text{Modulo}/2$ (with regard to the shortest distance).

the command values derived from the cam table do not perform motions higher than $\text{Modulo}/2$ even if the table points between the data points contain a longer motion distance.

The axis performs a motion different than expected according to the cam table.

NOTICE**Limiting the maximum master axis velocity**

The user has to ensure that such a case with a corresponding safety area is excluded (the maximum master axis velocity may not cause a velocity of the "cam axis" higher than $\text{Modulo}/2$ per analysis cycle of the cam).

The analysis cycle corresponds to the motion cycle time, control parameter C-0-0400, Motion, cycle time, command value.

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

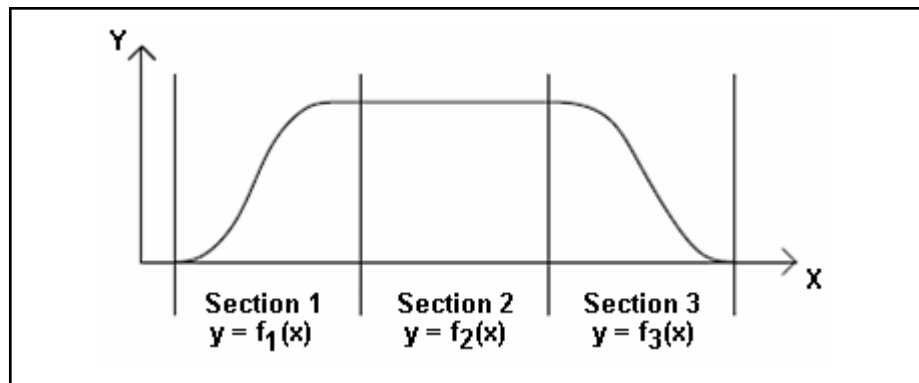


Fig. 7-42: Cam profile consisting of three motion steps

Mainly polynomial functions of the 5th order or higher are used since these polynomial functions 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 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, increased calculations at runtime are caused. 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 the PLC function blocks **MB_ChangeProfileSet** (for MotionProfile), **MB_ChangeFlexProfileSet** (for FlexProfile) or the IndraWorks interface.

7.13.3 Electronic cam

In the control, the typical coupling between the master 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. Each axis whose current actual position can be added to the cam table can be used as cam profile. It can also be virtual master axes. 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.

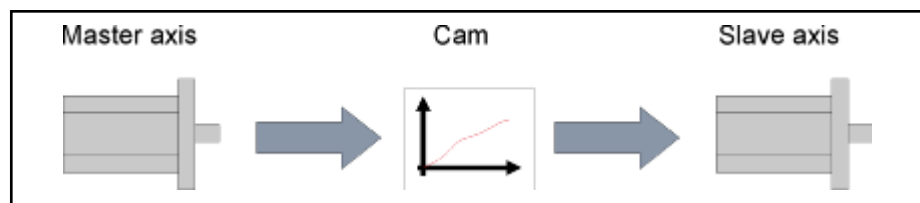


Fig. 7-43: Signal flow and cam localization

7.13.4 Use cases

Typical use cases 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

7.13.5 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 IndraLogic XLC 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.

7.13.6 FlexProfile – Basic functionalities

General information

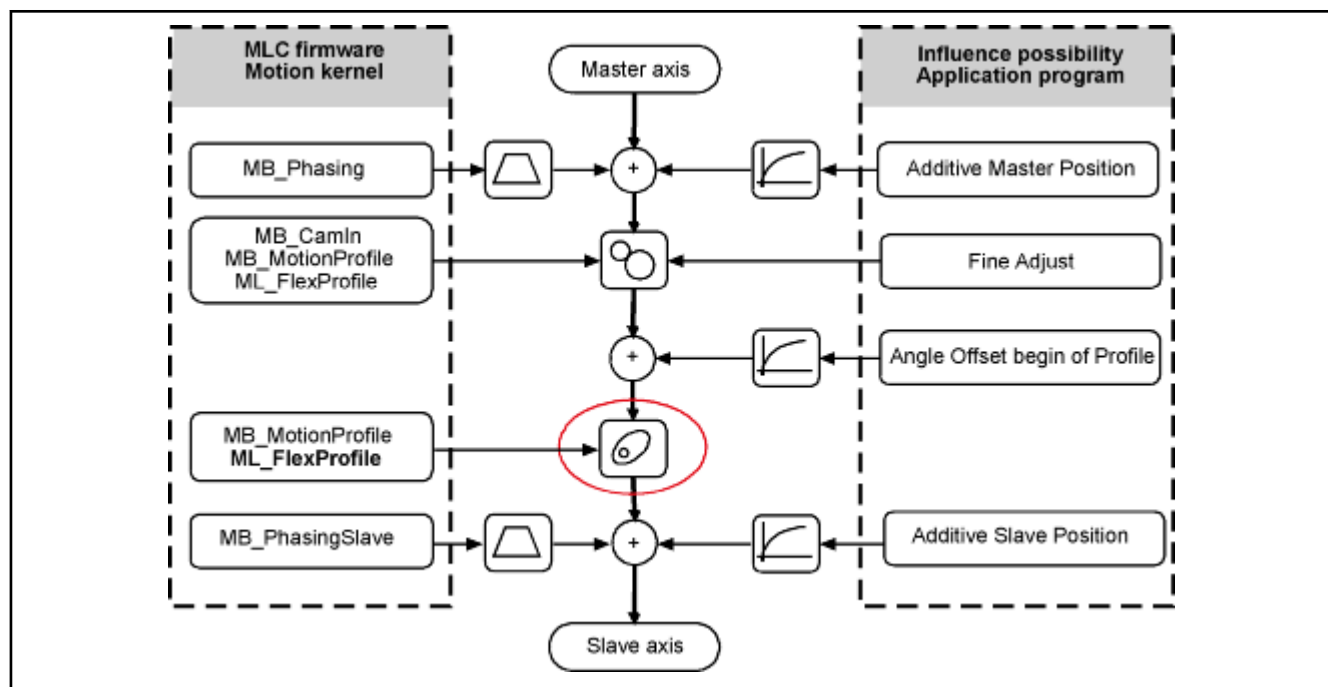
FlexProfile belongs to the group of "electronic motion laws". The FlexProfile is described by mathematical motion laws as analytical functions. In addition to the functions described in [chapter 7.13.4 "Use cases" on page 382](#), the following properties can be selected in the FlexProfile:

- 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. 7-44: Synchronous operating states **ML_FlexProfile**

The FlexProfile can be started using the PLC function block **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 absolute reference can be established again using a FlexStep.

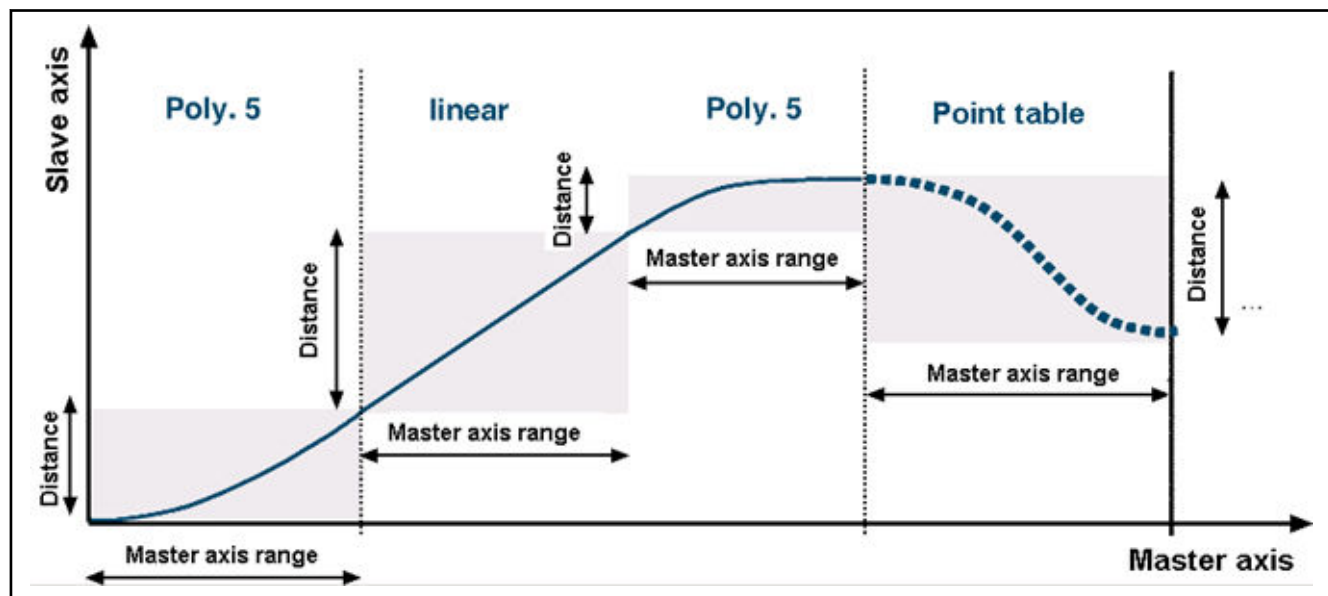


Fig. 7-45: Configuring and creating the FlexProfile

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 FlexStep (see [chapter "Flex step" on page 387](#)) must 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	$v < > 0$	$a < > 0$

Tab. 7-12: 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 special motion laws improve the adaptation of the MotionProfile to the motion problem with different degrees of freedom.

Generally, the following boundary conditions are specified and result in the master axis range (refer to the following figure):

- Distance
- Velocity
- Acceleration
- Jerk

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.

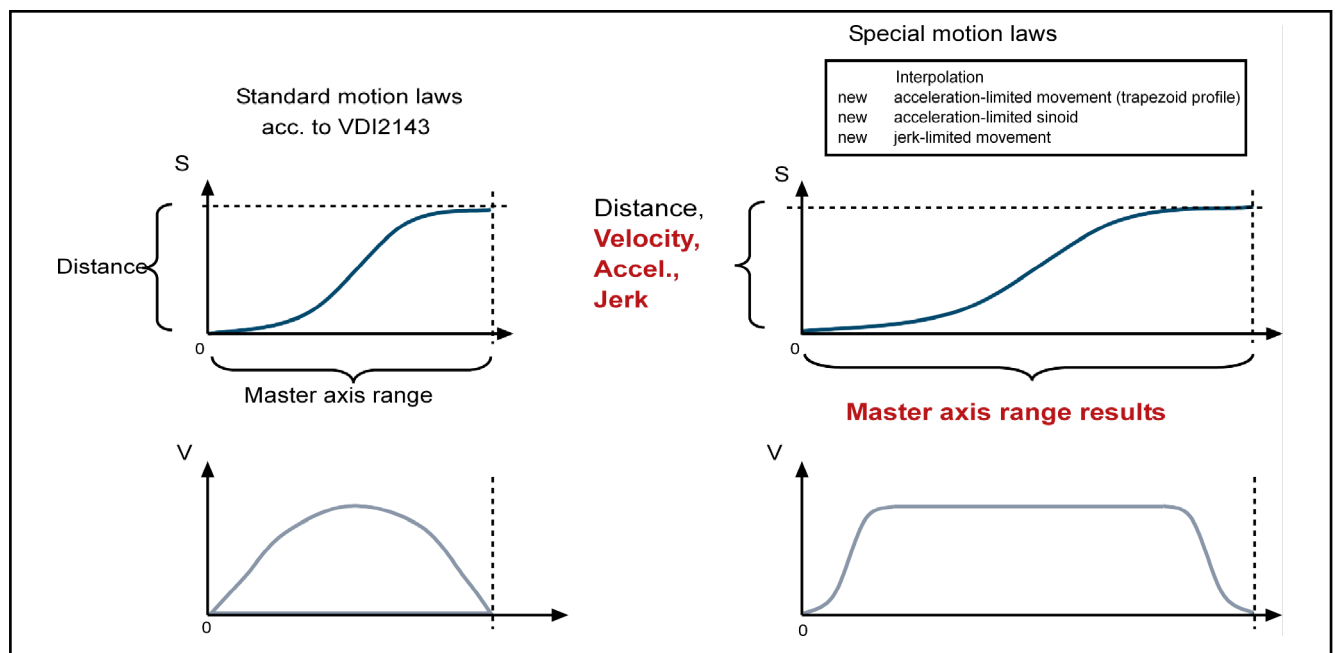


Fig. 7-46: Properties of the special motion laws

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).

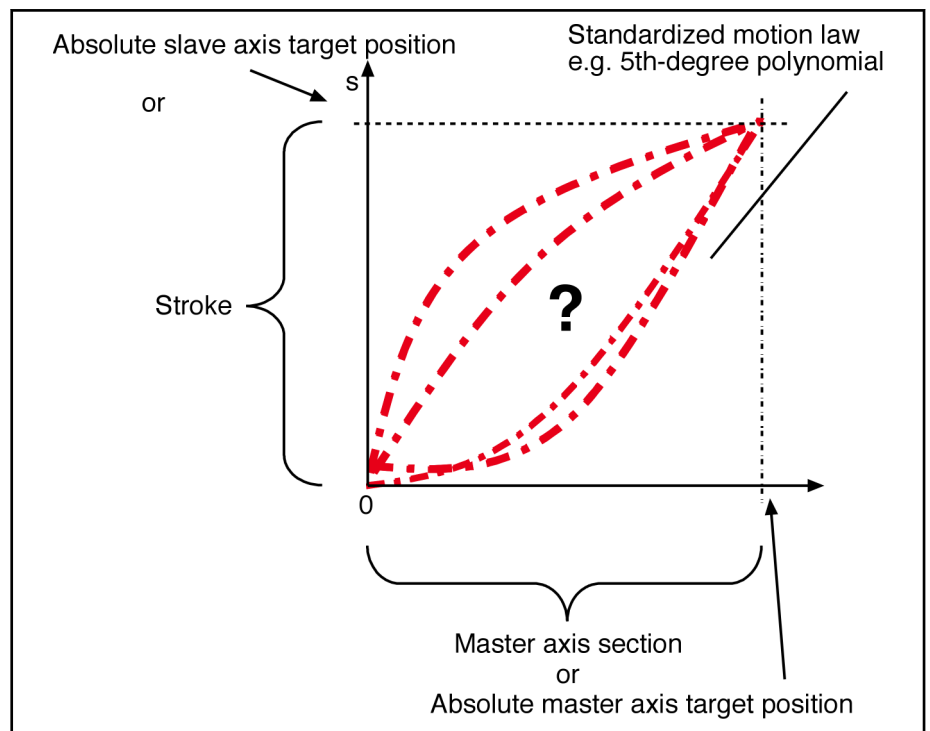


Fig. 7-47: Configuration possibilities of the FlexStep

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 relative distance and relative master axis range
- FlexStep with absolute distance and relative master axis range
- FlexStep with relative distance and absolute master axis range
- FlexStep with absolute distance and absolute master axis range

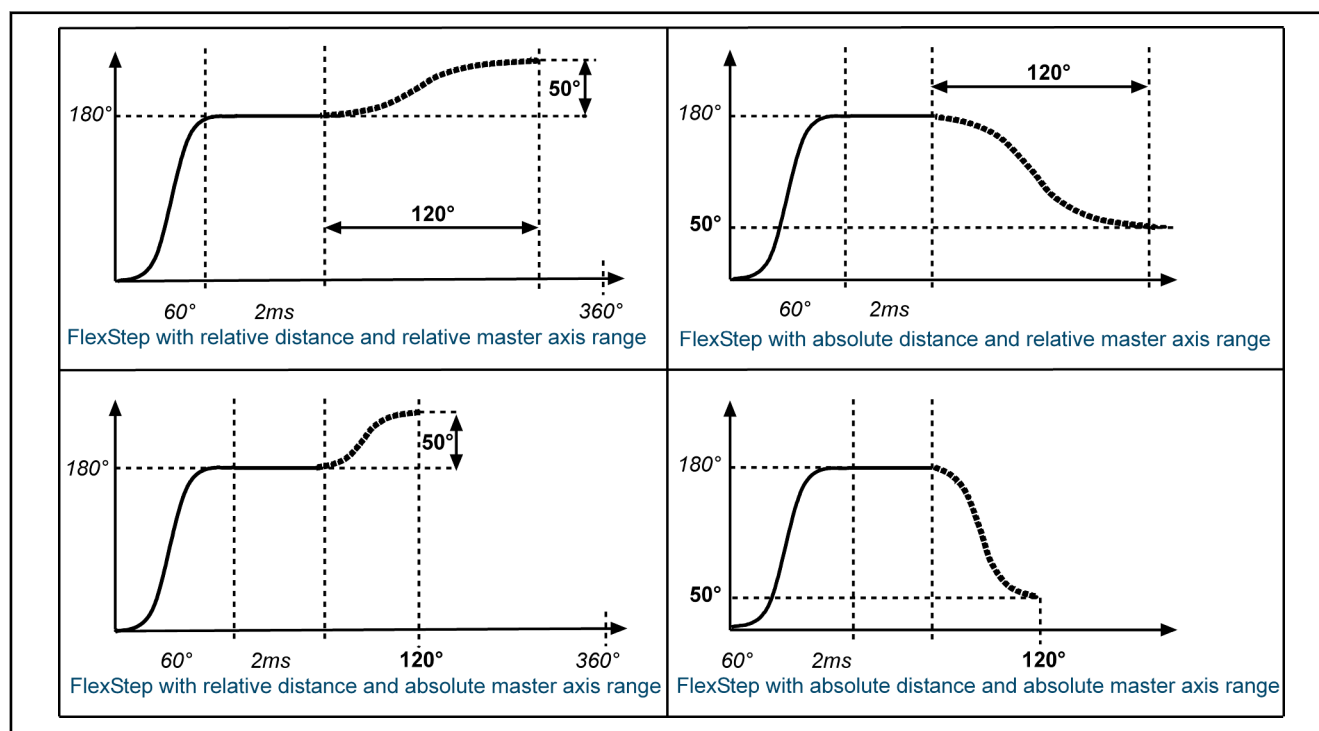


Fig. 7-48: Configuration of the FlexStep



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)		Tra-versing range		Boun-dary val-ues left			Boun-dary val-ue right			Max (not consid-ered ev-ery-where)			TPS pos-sible	Flex-Step possi-ble	Opera-tion 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 line	REST_IN_REST_INCLINEDSINE	0	*	*	0	0	=	0	0	=	=	=	=	x		CT, MP, FP
Acceleration-optimum inclined sinusoid	REST_IN_REST_SINEACC	3072	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Torque-optimum inclined sinusoid	REST_IN_REST_SINE-TORQUE	3328	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Gutman sinusoid	REST_IN_REST_GUTMANSINE	2816	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Modified sinusoid	REST_IN_REST_MODIFIED_SINE	3840	*	*	0	0	=	0	0	=	=	=	=	x		CT, FP
Modified acceleration trapezoid	REST_IN_REST_MODIFIED_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
Square 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)		Tra-versing range		Boun-dary val-ues left			Boun-dary val-ue right			Max (not consid-ered ev-ery-where)			TPS pos-sible	Flex-Step possi-ble	Opera-tion 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 sinusoid	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 (starting acceleration zero)	X_VEL_STARTACC	33280	=	*	*	0	=	*	=	=	=	=	=			CT
Velocity 2nd Order (end acceleration 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																

Motion functionality

	Definition of the motion law at the input "MotionLaw" (ProfileStepAdr)		Tra-versing range		Boun-dary val-ues left			Boun-dary val-ue right			Max (not consid-ered ev-ery-where)			TPS pos-sible	Flex-Step possi-ble	Opera-tion modes
			Distance	Range	StVel	StAcc	StJerk	EndVel	EndAcc	EndJerk	MaxVel	MaxAcc	MaxJerk			
Acceleration start; motion (trapezoid profile)	X_FIT_VEL_TRAPEZE_ALIM	7000	*	=	*	=	=	*	=	=	*	*	=			CT, FP
Acceleration start; sine curve	X_FIT_SINE_TRAPEZE_ALIM	28928	*	=	*	=	=	*	=	=	*	*	=			CT, FP
Jerk-lim. motion (trapezoid profile)	X_FIT_ACC_TRAPEZE_JLIM	29184	*	=	*	=	=	*	=	=	*	*	*			CT, FP
- Other																
Acceleration start, 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 start (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

Tab. 7-13: Boundary conditions of the FlexProfile motion laws

Switching and synchronizing 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

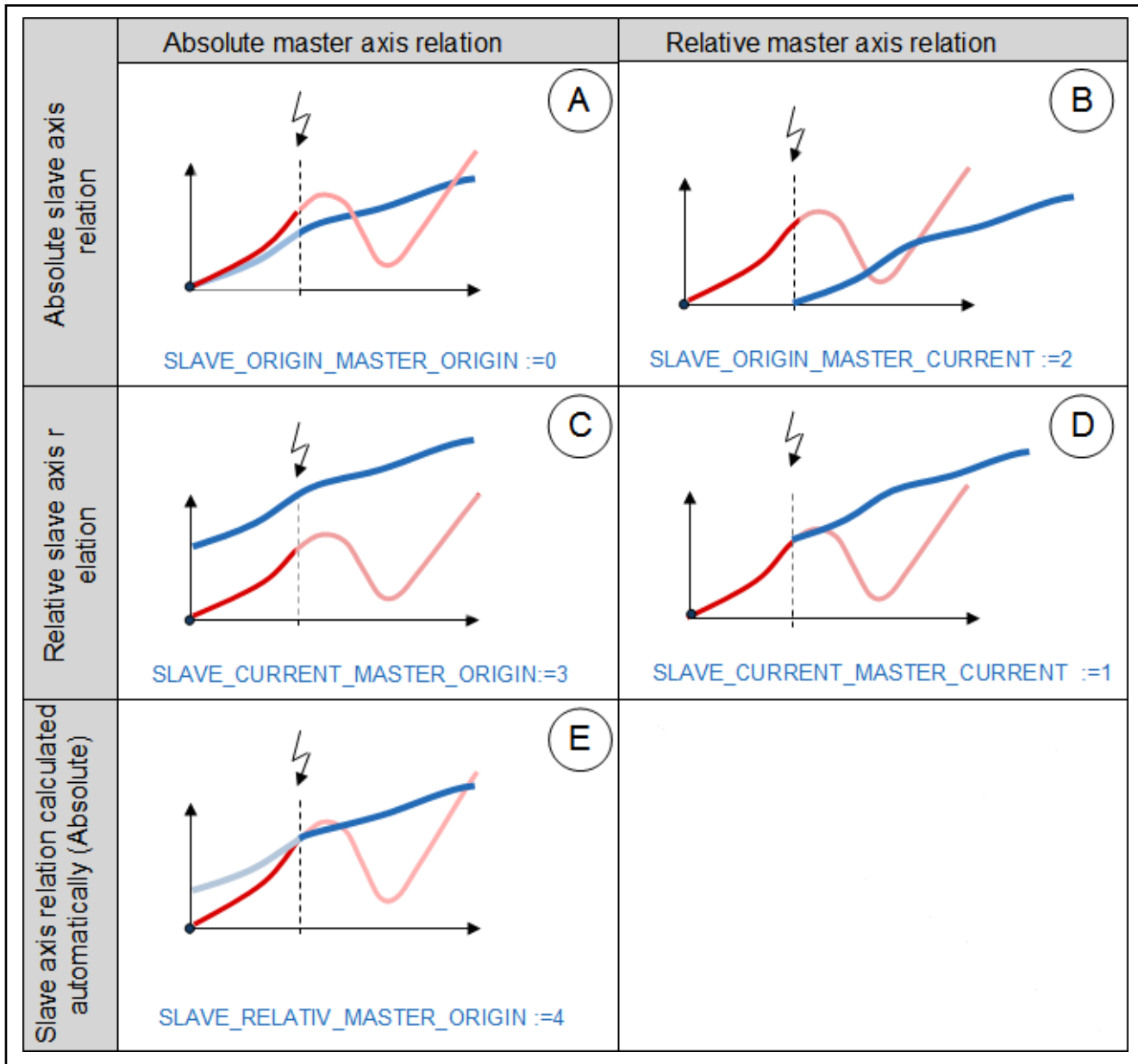


Fig. 7-49: Selecting the coordinate system

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

zero point of the coordinate system (absolute master axis position and absolute slave axis position - case A). The MotionProfiles are not moved.

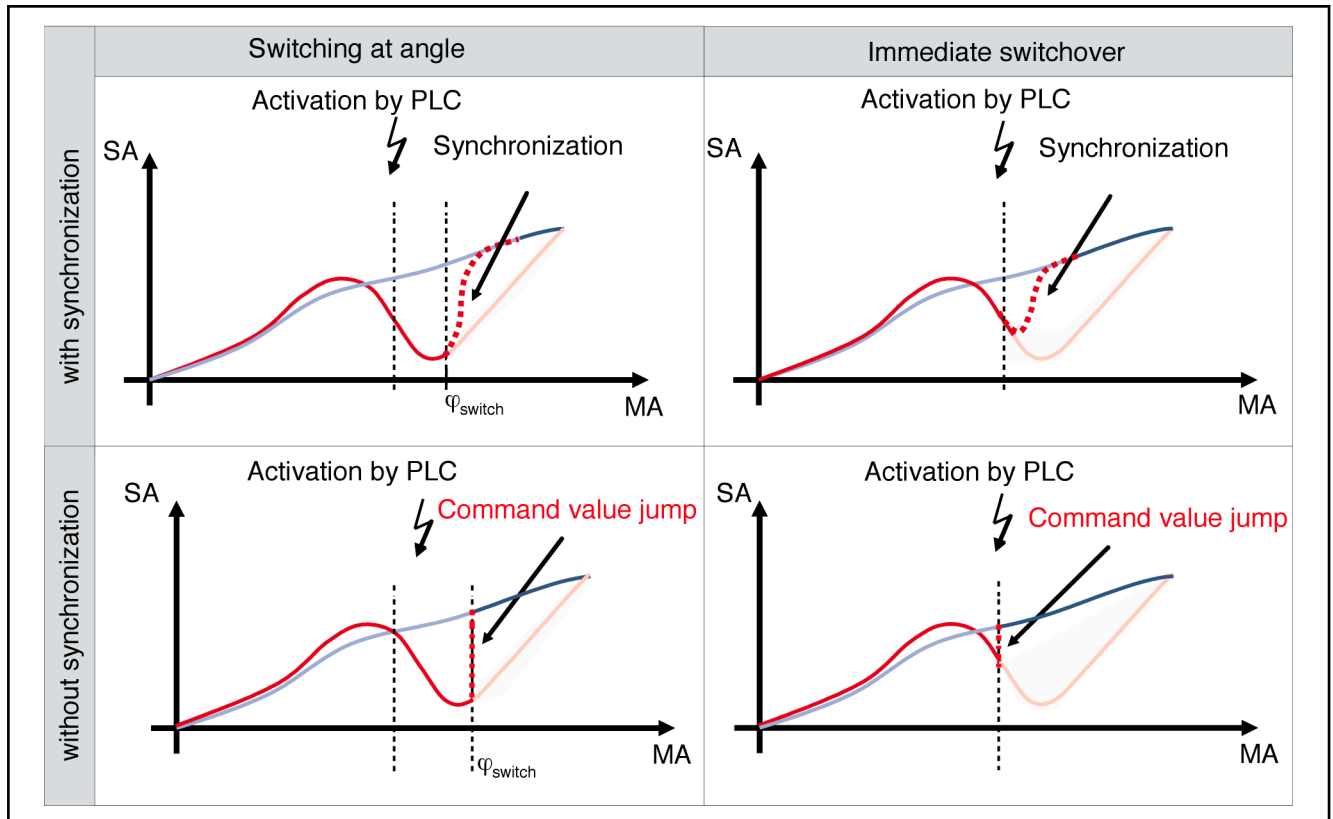


Fig. 7-50: Case A – Switching and synchronizing with absolute master axis position and absolute slave axis position

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

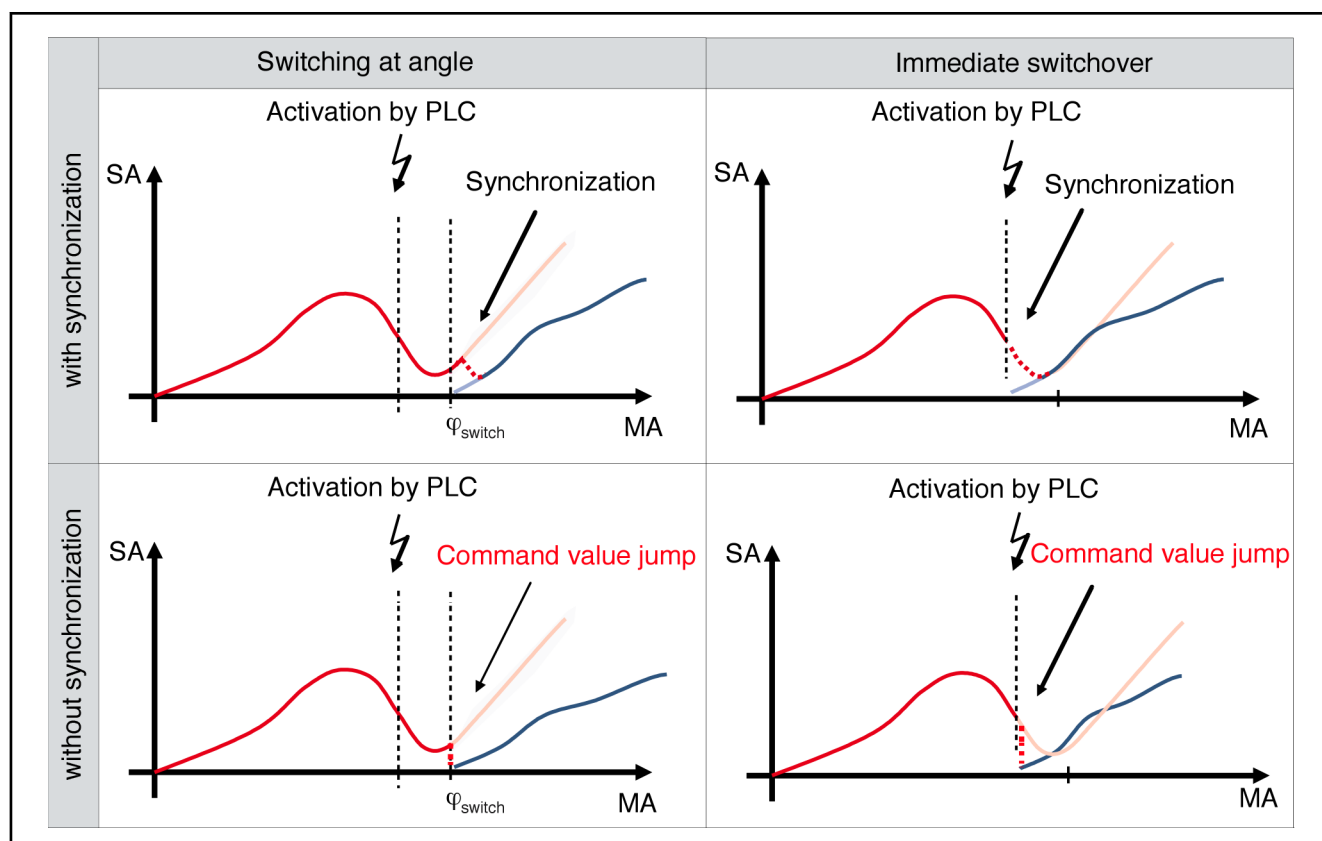


Fig. 7-51: Case B – Switching and synchronizing with relative master axis position and absolute slave axis position

The following figure shows the switching to a different MotionProfile. The origin of the second MotionProfile is moved 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.

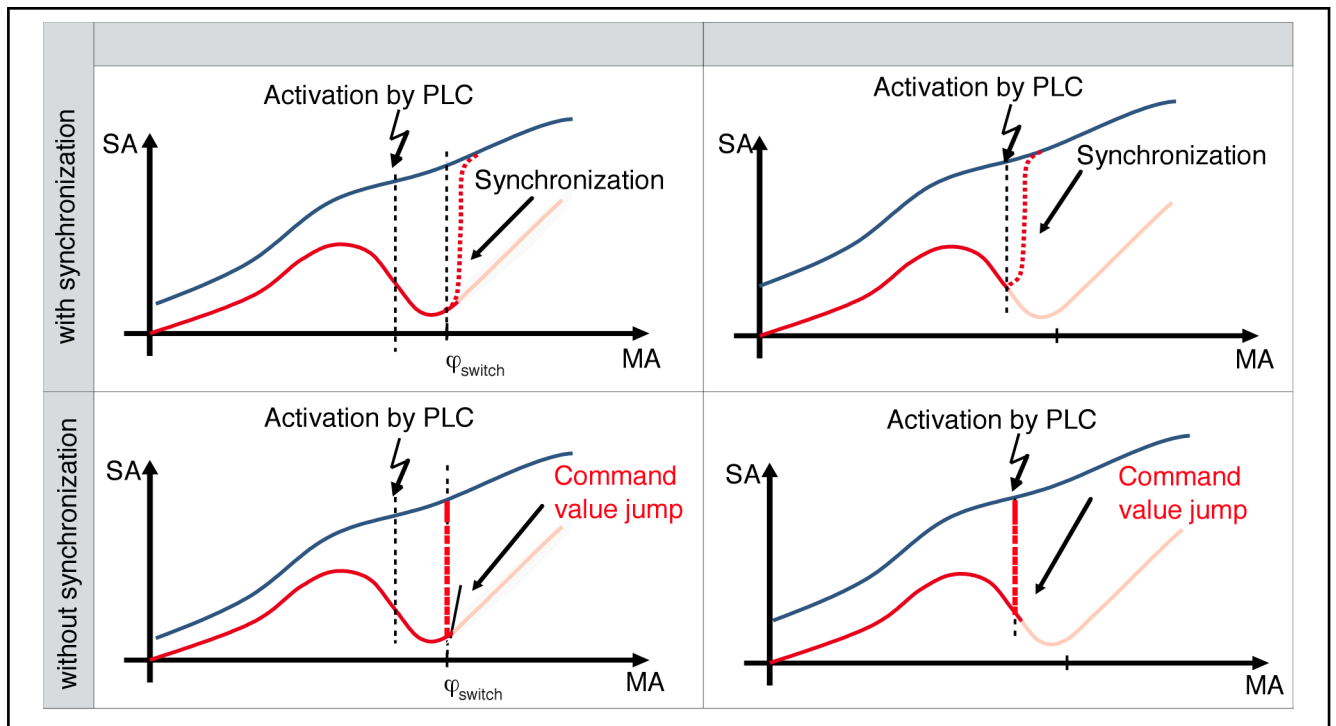


Fig. 7-52: Case C – Switching and synchronizing with absolute master axis position and relative slave axis position

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

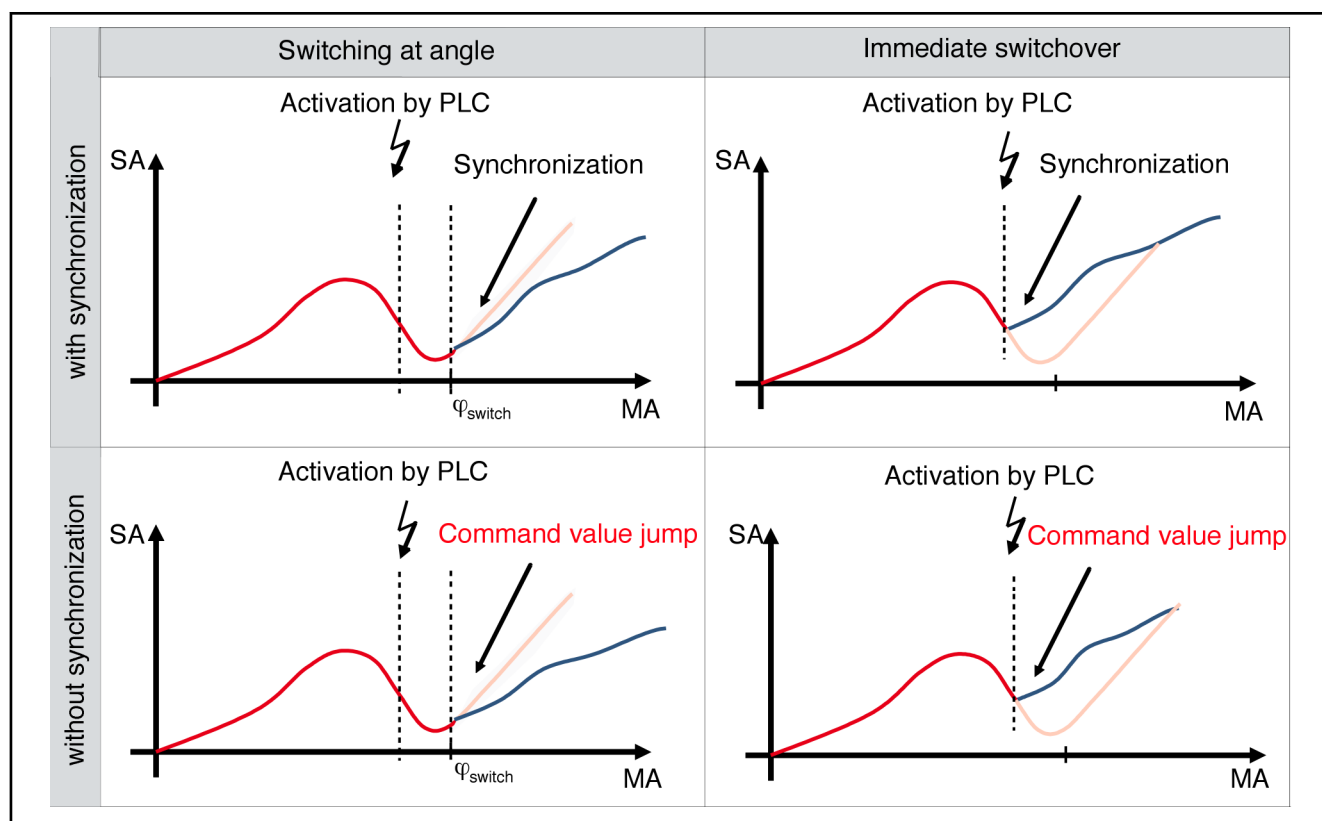


Fig. 7-53: Case D – Switching and synchronizing with relative master axis position and relative slave axis position

The following figure shows the switching to a different MotionProfile. The second MotionProfile is moved 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.

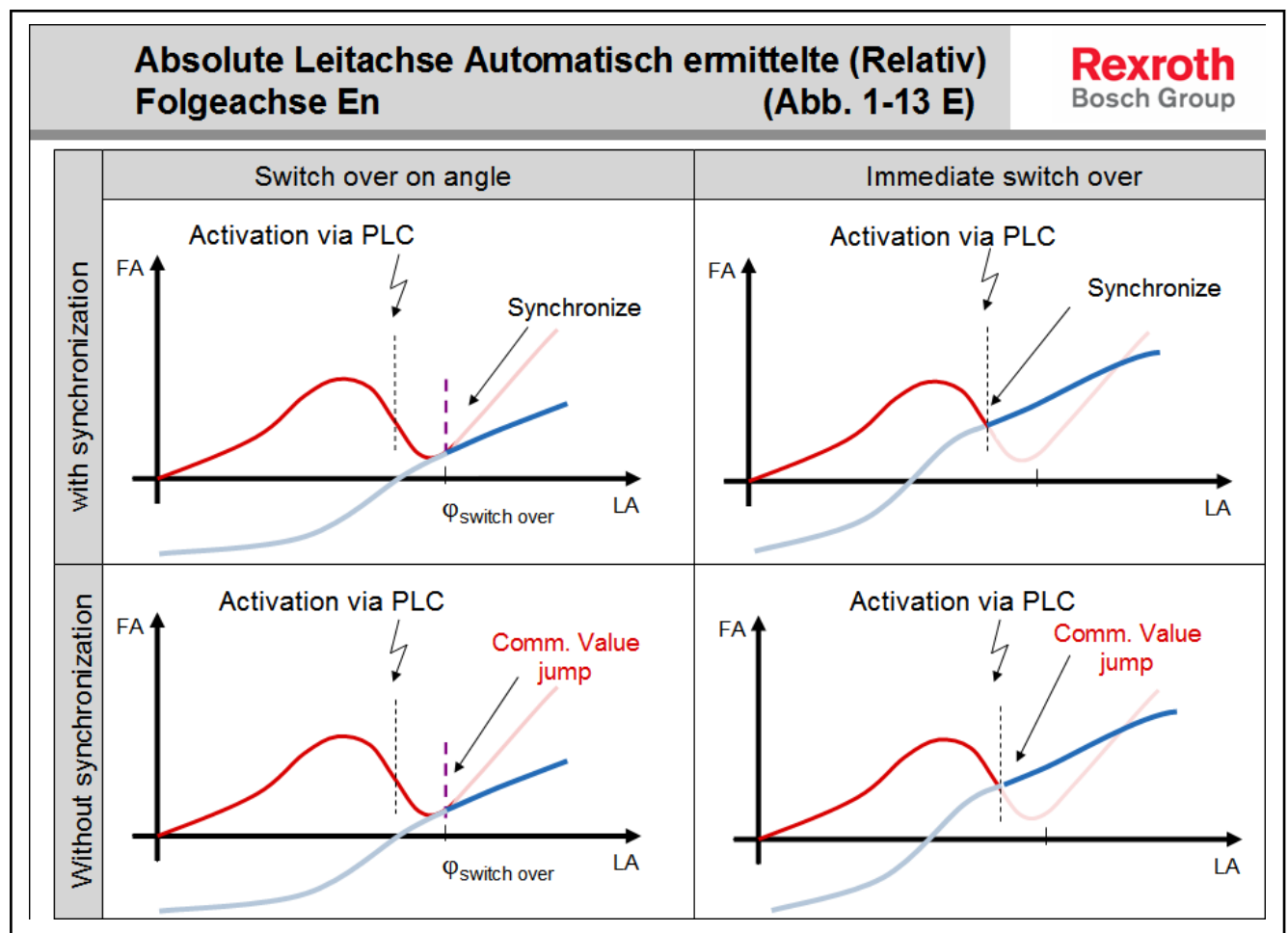


Fig. 7-54: Case E – Switching and synchronizing with absolute master axis position and automatically determined slave axis position (relative)

Processing type

There are two processing types (refer to the following figure) are distinguished:

- 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" function block.

Motion functionality

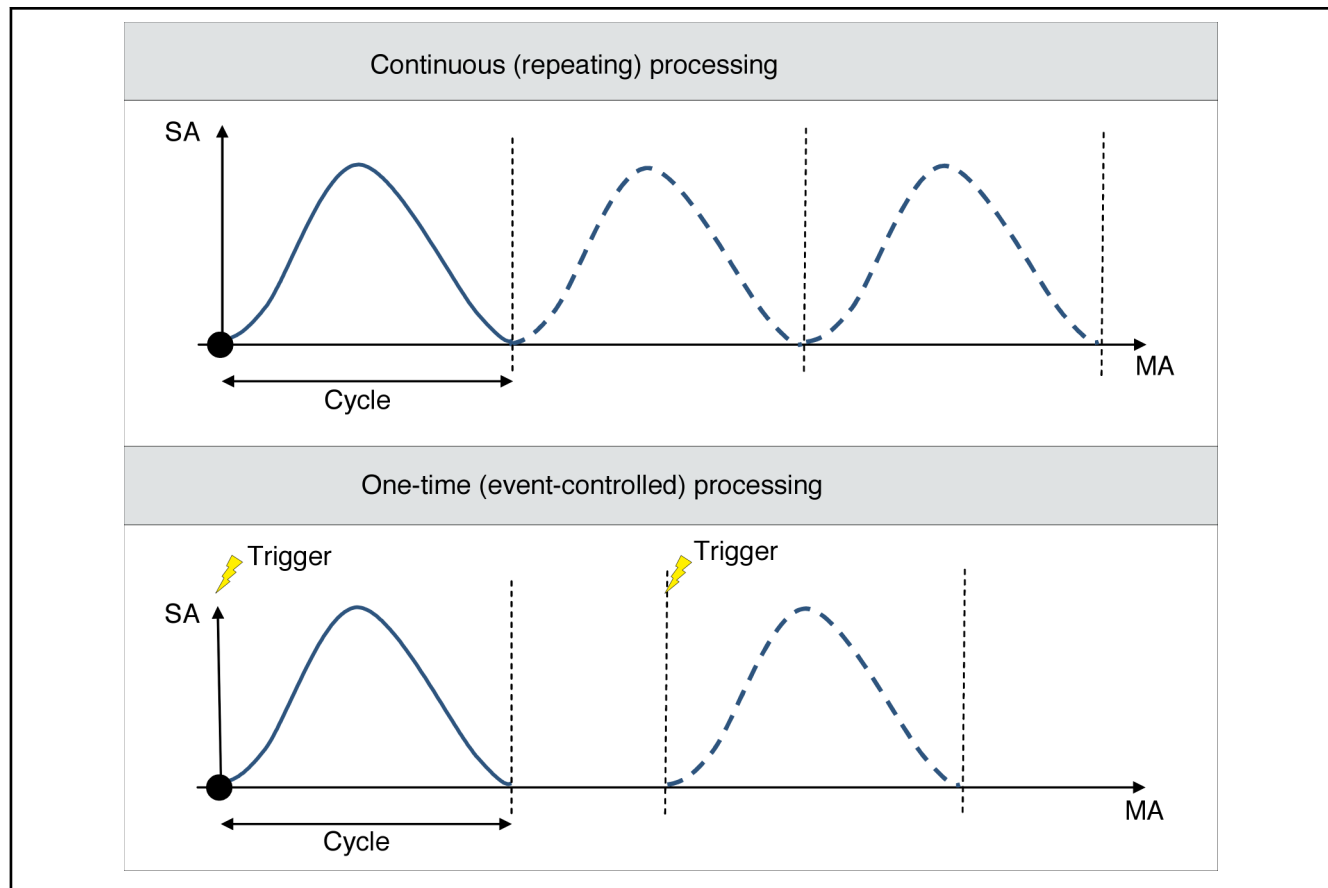


Fig. 7-55: Continuous processing and one-time processing

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.

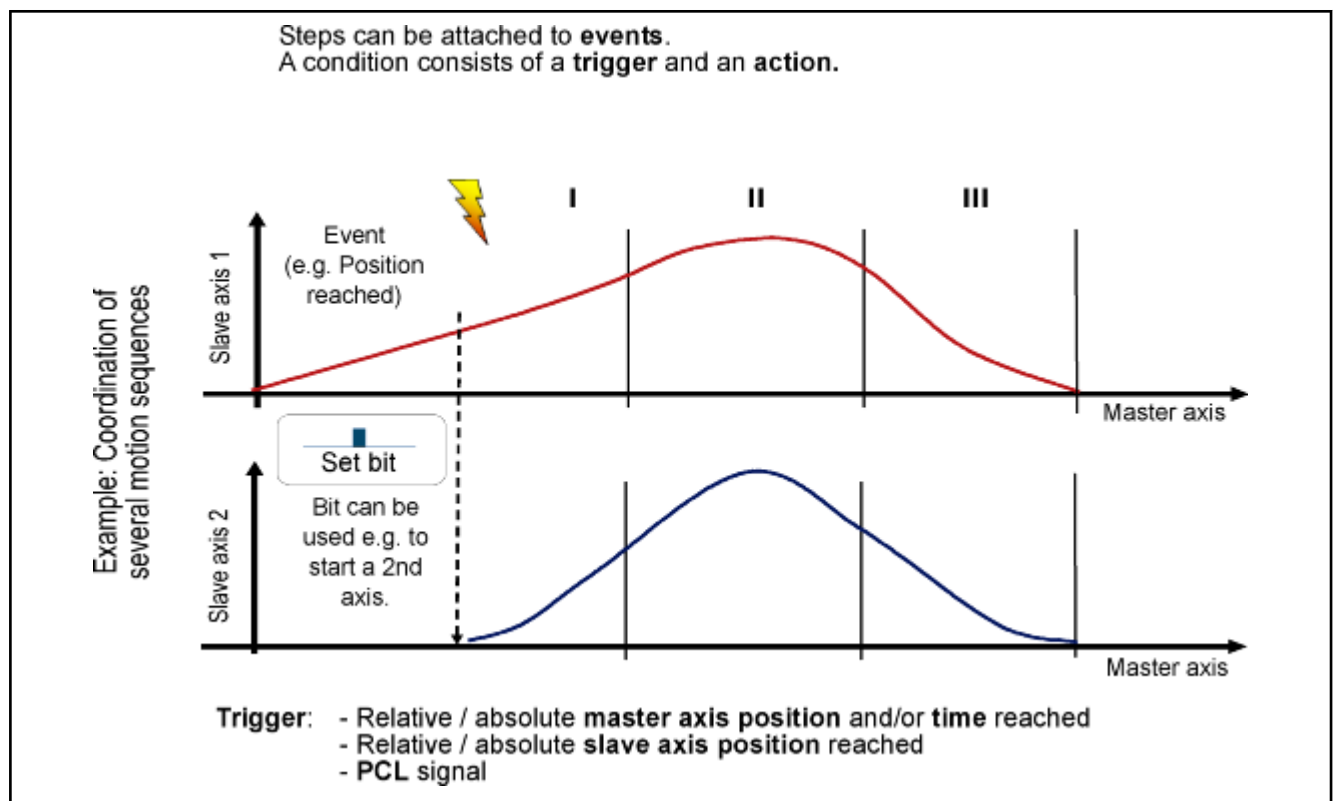


Fig. 7-56: Example of an event control

Events are very suitable to mark these points in the MotionProfile. It is very easy to respond on these points 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 states 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 the master axis 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.

The following actions are possible:

- 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 motion 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

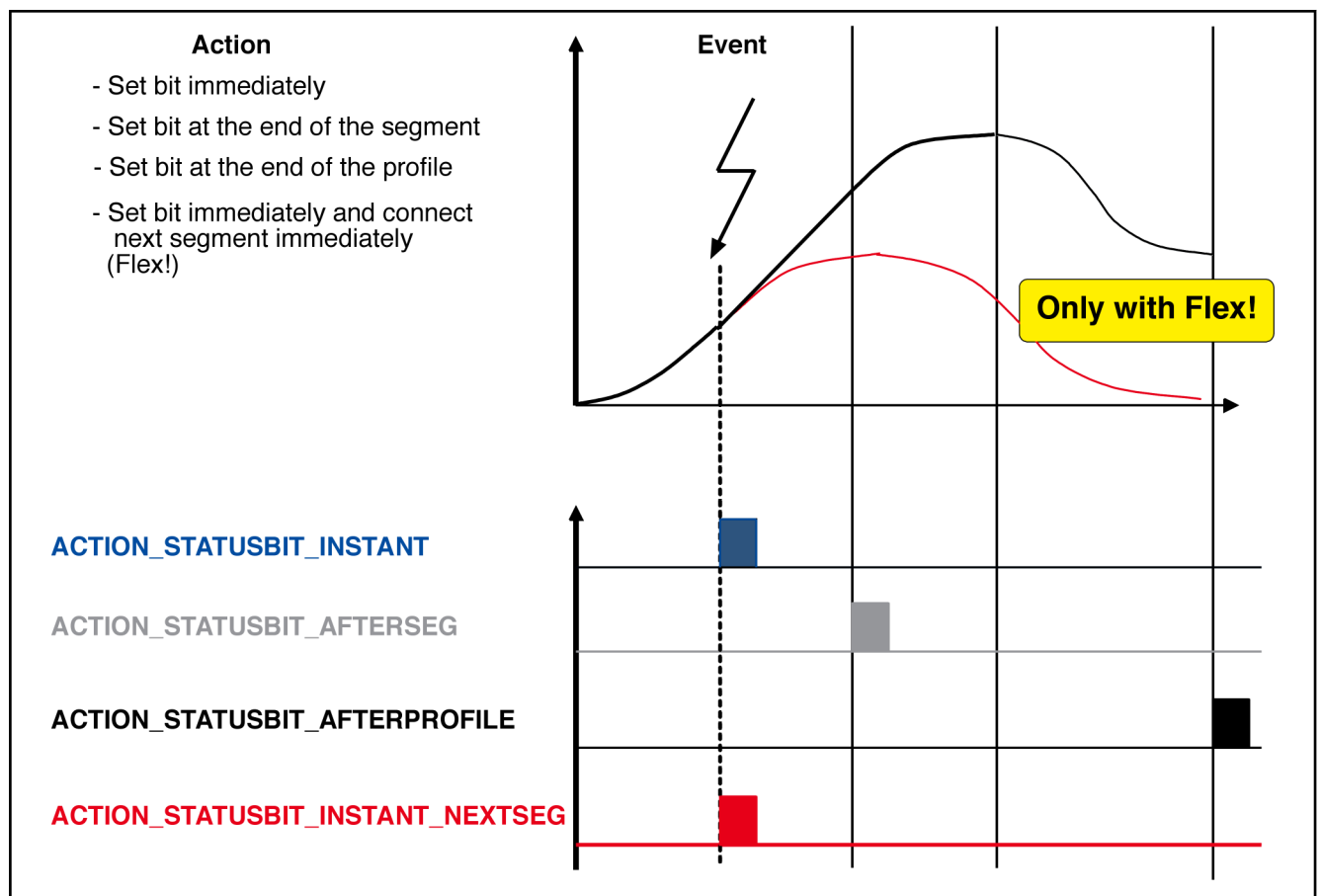


Fig. 7-57: Possible actions

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

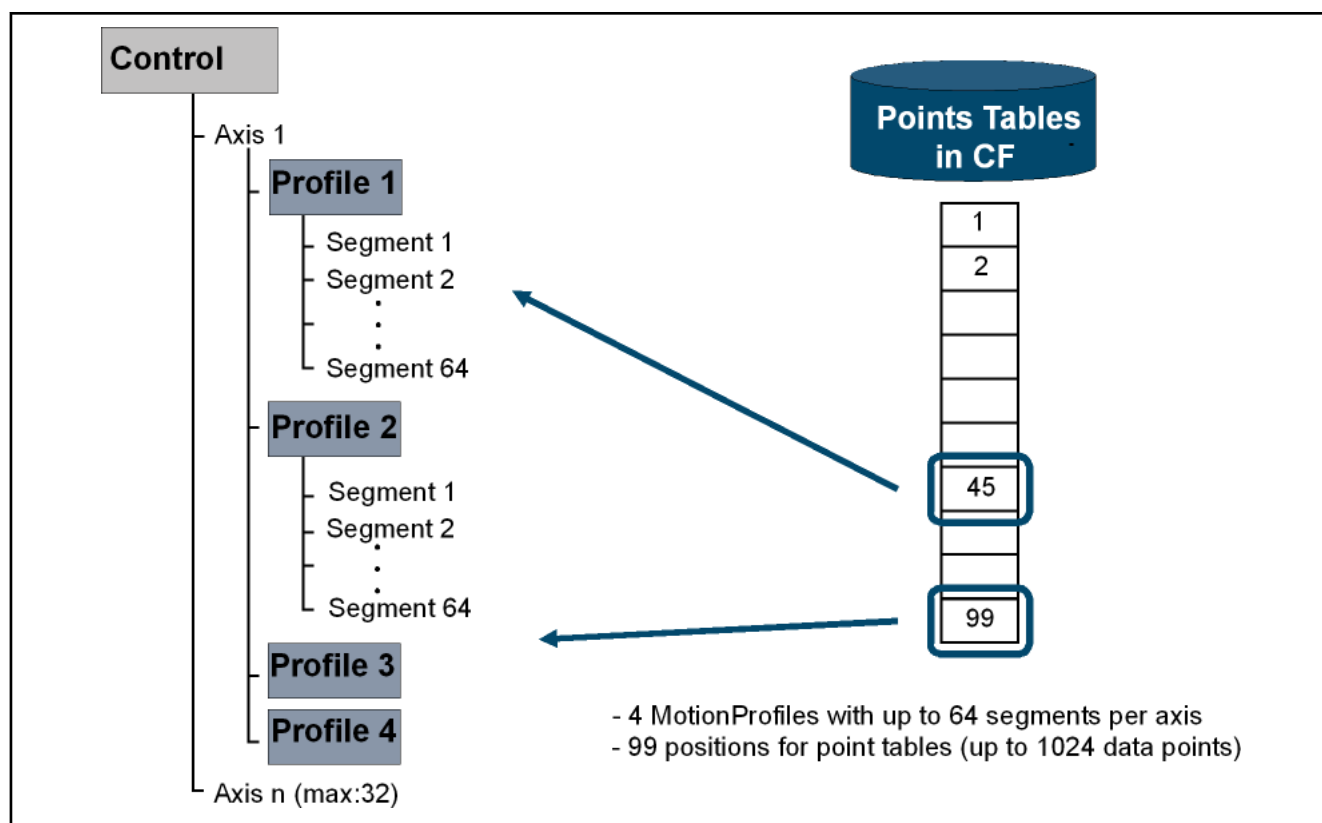


Fig. 7-58: Assigning MotionProfile and axis

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. 7-59: 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.

Configuring via PLC function blocks

The PLC provides four function blocks allowing the FlexProfiles use. In general, these function blocks 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 have to be defined during the motion step configuration, as these are provided with 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.

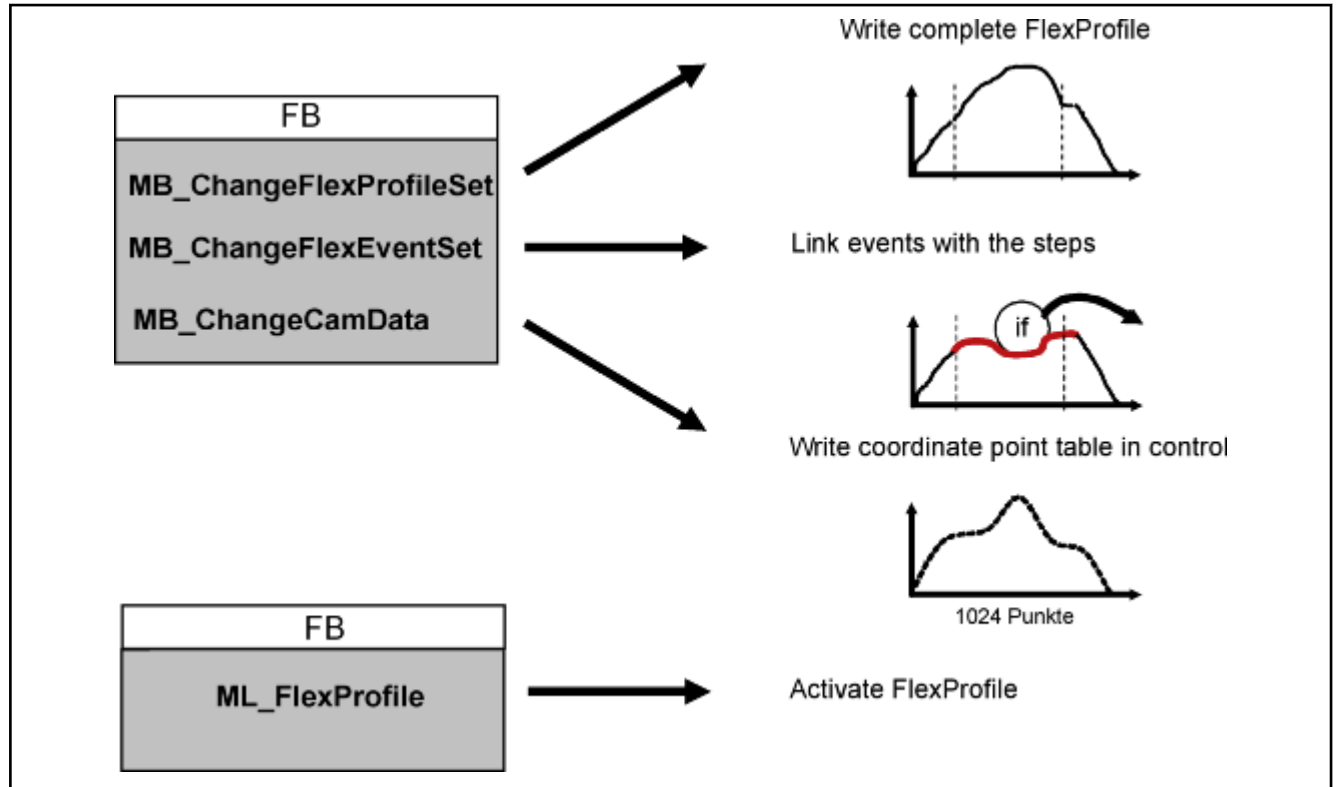


Fig. 7-60: Overview of the PLC function blocks

Finally, the corresponding motion steps can be linked to cam tables. These cam tables 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".

Configuring via IndraWorks

Click on the "FlexProfile" entry in the axis tree of the desired project and axis and a dialog opens to set the general profile configuration.

Motion functionality

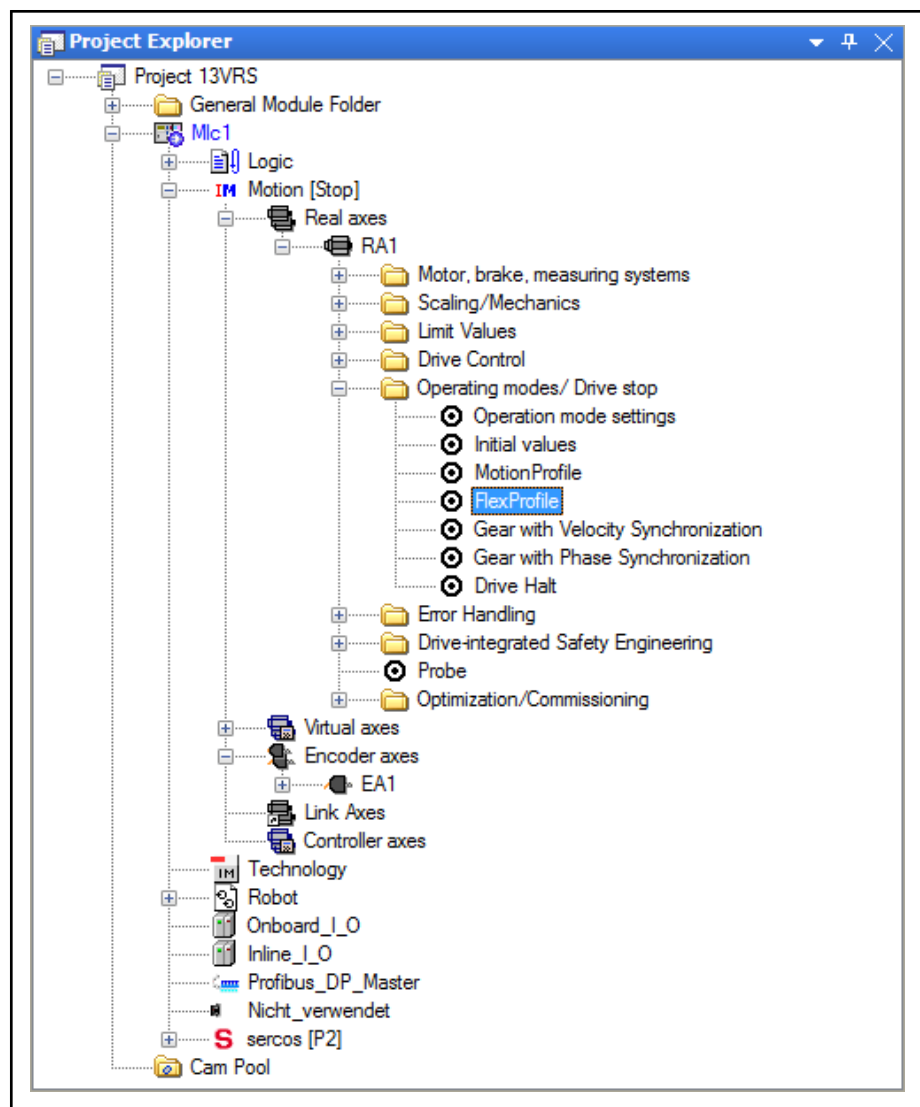


Fig. 7-61: Locating the FlexProfile in the axis structure

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.

Motion functionality

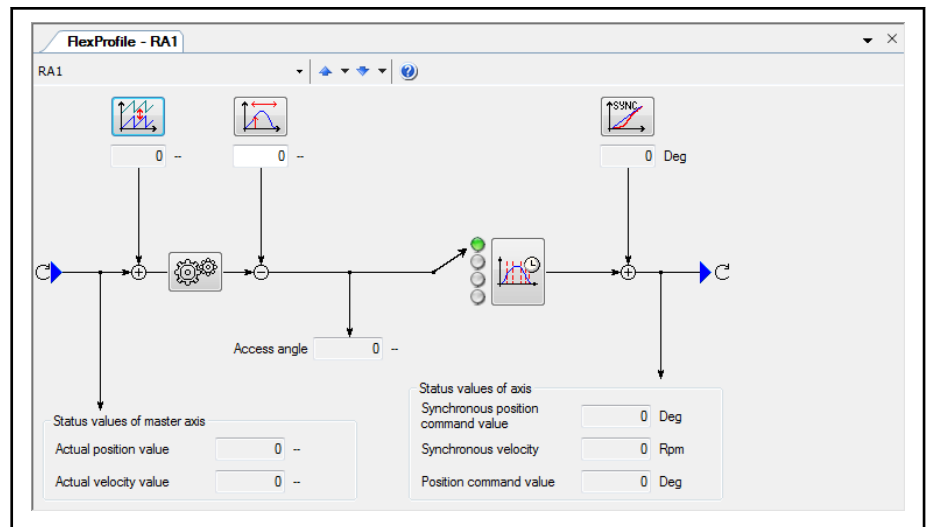


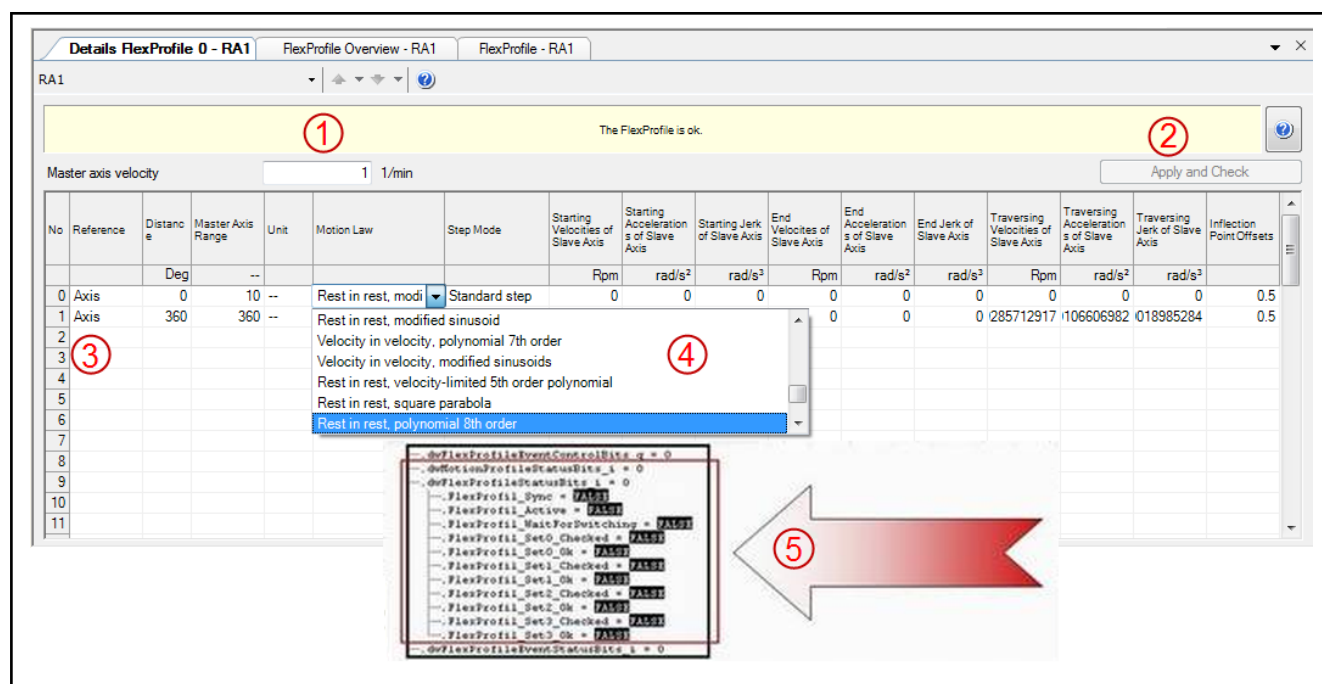
Fig. 7-62: FlexProfiles: Action chain between master axis and slave axis

Click on the "FlexProfile" icon in the middle of the dialog, to go to the MotionProfiles of the FlexProfile (see [fig. 7-62 "FlexProfiles: Action chain between master axis and slave axis" on page 405](#)). 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 (refer to ③ in the following figure).



- 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

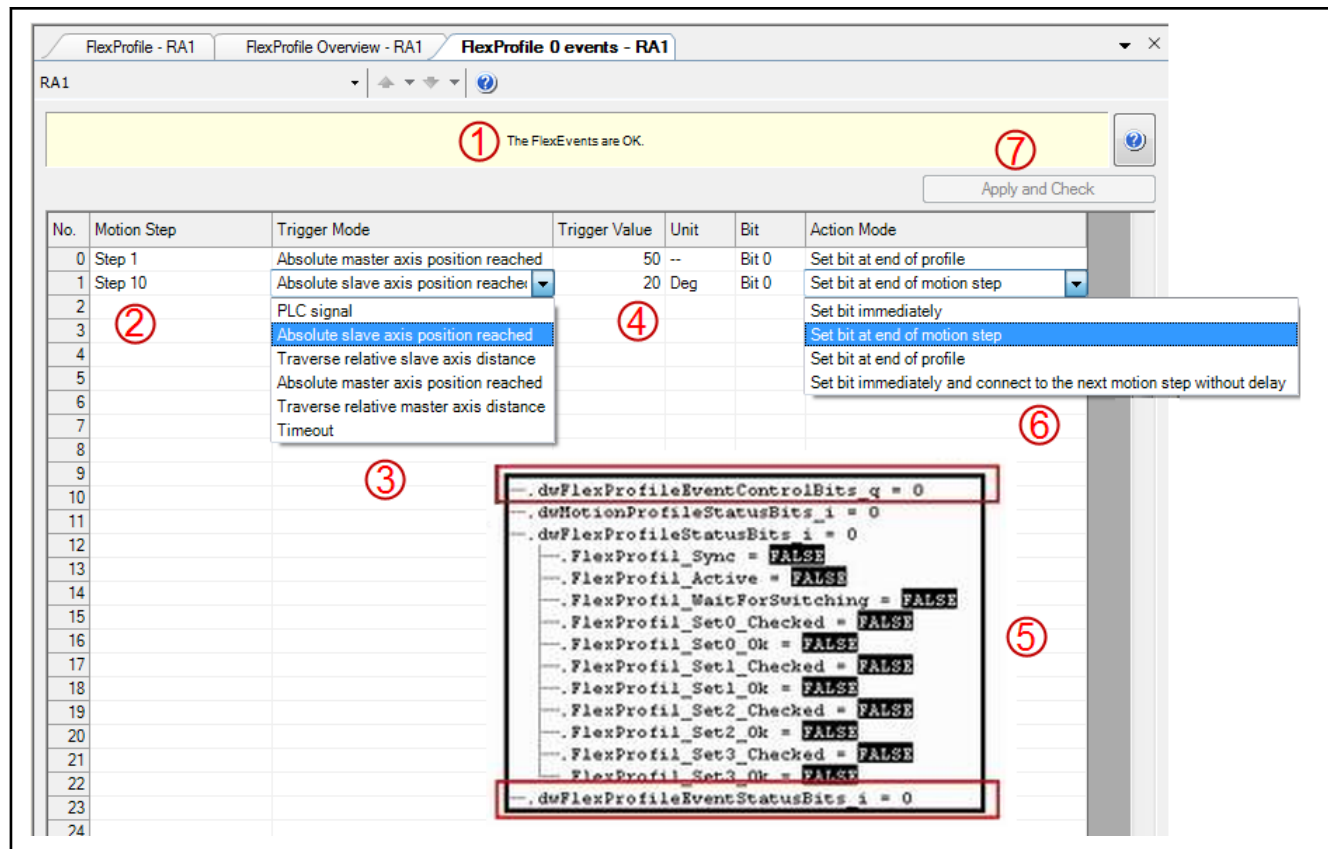
Fig. 7-64: FlexProfile configuration



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 in fig. 7-63 "FlexProfile overview window" on page 406). 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 infig. 7-64 "FlexProfile configuration" on page 407).

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. 7-65: Event configuration window

Configuring via CamBuilder

For the description of the CamBuilder, refer to "Rexroth IndraWorks 13VRS CamBuilder" (see [chapter 1.4 "Required and supplementing documents"](#) on page 18).

7.14 Touch probe

7.14.1 Introduction



The touch probe function is only available with IndraLogic XLC Lxx variants with a Sercos interface.

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.

7.14.2 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 position values of all configured axes
- Free assignment of touch probe input to evaluation unit and axis
- Measuring 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)

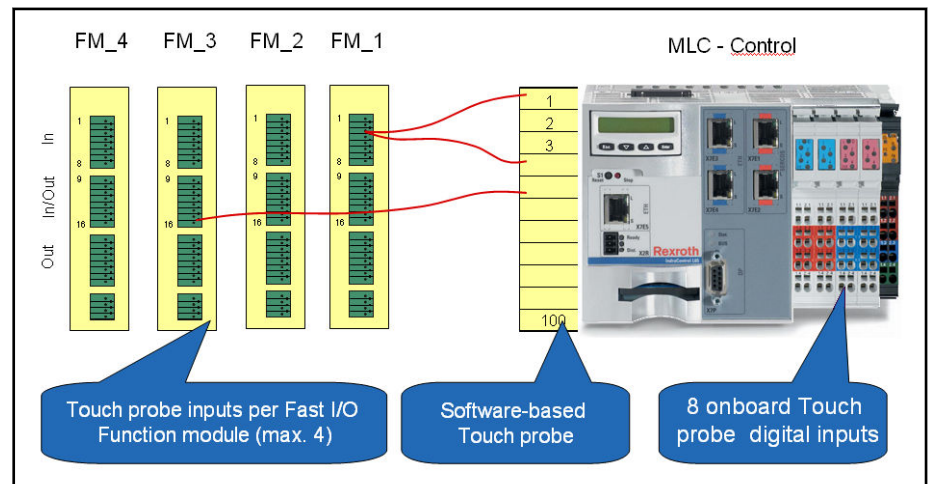


Fig. 7-66: Touch probe hardware architecture

7.14.3 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 "Rexroth IndraControl Lxx 13VRS" Function Modules (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

Motion functionality

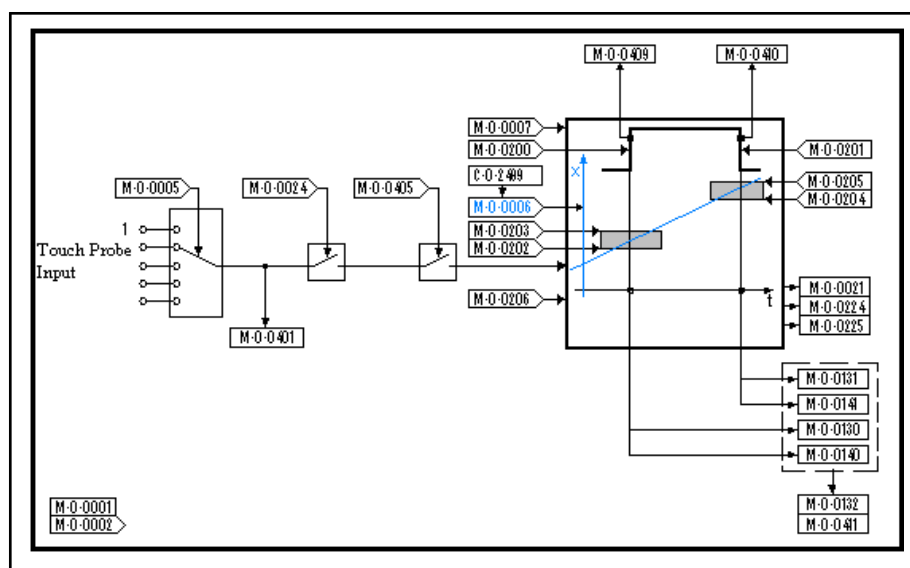


Fig. 7-67: Overview and mode of operation of the touch probe-related parameters

The following touch probe-related parameters are listed:

Common C-parameters for all touch probes

- C-0-2499, Touch probe, IDN list signal selection



The parameter attribute "Touch probe" indicates whether a parameter is intended for the signal selection.

M parameter dataset for each touch probe

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-0-0009, Osci: list of all M-parameters that can be oscilloscoped
M-0-0010, List of all M-parameters
M-0-0016, Project identification number
M-0-0021, Status word
M-0-0024, Command: Activation
M-0-0130, Measured value, positive edge
M-0-0131, Measured value, negative edge
M-0-0132, Difference value
M-0-0133, Time difference
M-0-0140, Time stamp, positive edge
M-0-0141, Time stamp, 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, Maximum 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, Enabling
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
M-0-1000, Command: Load basic touch probe parameters
M-0-2100, List of all backup operation M-parameters

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, the function module has to be activated (status LED flashes in green).

7.14.4 Accuracy

The position values are received in the grid of the set Sercos cycle time by the control or generated by the drive. The position value belonging to the point of time of the signal edge is determined by the configurable interpolation between the current and the following position value (refer to the parameter M-0-0007).

7.14.5 Data acquisition of the touch probe

Activating 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 "M-0-0024 Command: Activation"
- The respective touch probe input for the measured value acquisition was enabled in "M-0-0405, Enabling"

Motion functionality

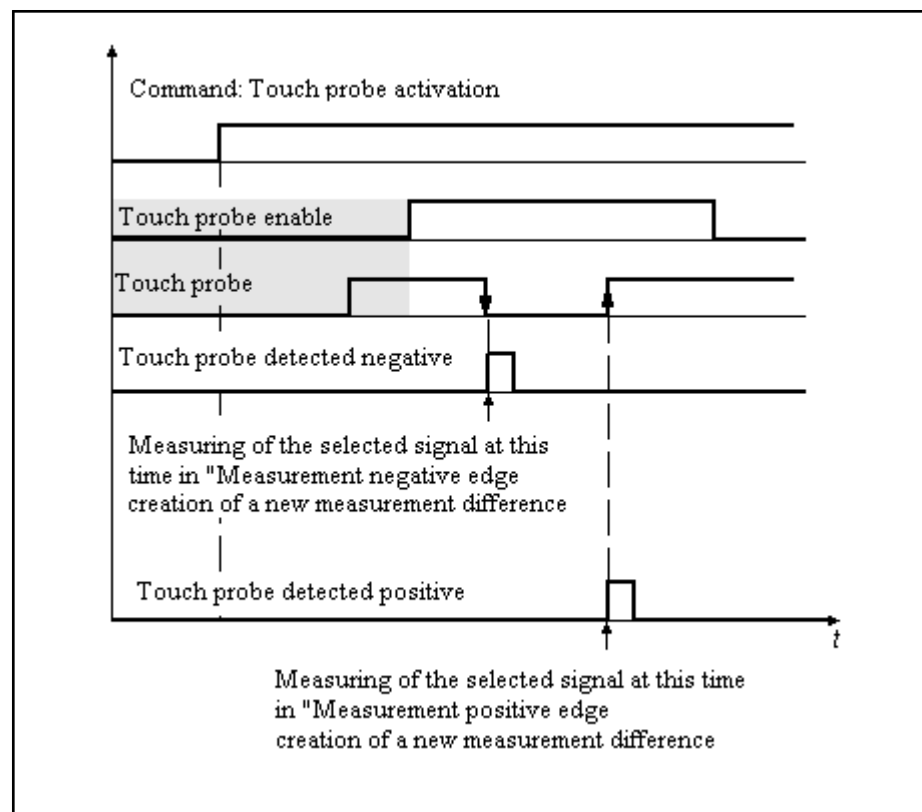


Fig. 7-68: Evaluation of touch probe edges (in case of switched on evaluation of positive and negative edges in M-0-0007)

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.

Selecting measuring signal

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 position values (expectation window) in which measurements can take place

- No trigger failure monitoring registering the passing of this range 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.

The following settings can be made:

- 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

The following parameters can be used to compensate the signal runtime of the actual touch probe event up to the detection in the control.

- M-0-0200, Dead time compensation, positive edge
- M-0-0201, Dead time compensation negative edge

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:

$$P_{\text{Det}} = P_{\text{Act}} + v_{\text{Act}} \cdot T_{\text{Det}}$$

Fig. 7-69: 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_{Det} : 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):

$$T_{\text{Det}} = \frac{P_{\text{Det2}} - P_{\text{Det1}}}{v_{\text{Act2}} - v_{\text{Act1}}}$$

Fig. 7-70: Dead time calculation

P_{Det1} , P_{Det2} : Determined touch probe position for measurement 1 and measurement 2

Motion functionality

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:

$$T_{TotPos} = \frac{101^\circ - 100^\circ}{3000 \left[\frac{U}{\min} \right] - 10 \left[\frac{U}{\min} \right]} = \frac{1^\circ}{2990 \cdot \frac{360^\circ}{60s}} = 55.74ns$$

Fig. 7-71: 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. 7-72: Dead time calculation for negative edge

Restarting or disabling measured value acquisition

A single shot measurement or a continuous measurement restart is triggered when resetting the parameter for the touch probe enabling (1 ⇒ 0) and setting it again (0 ⇒ 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

7.14.6 Touch probe expectation window

Using the expectation window

The value range of the signal to be measured, 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 the set value range. 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 value of the signal to be measured 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 value range 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.

Setting the expectation window

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".

Motion functionality

- 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"

7.14.7 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, Logic touch 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.

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.

Motion functionality

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
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

Bit	Values/function
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:

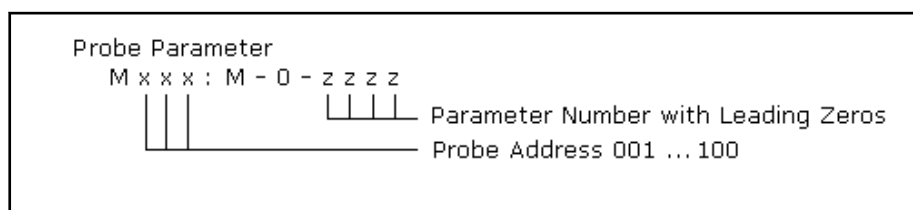


Fig. 7-73: Identity number format

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.

Motion functionality

The status parameter includes information on the

- availability of newly measured values
- 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.
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". The status 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:

- Latest measured value at a positive edge - Latest measured value at a negative edge
- Latest measured value at a positive edge - Penultimate measured value at a positive edge
- Latest measured value at a negative edge - Latest measured value at a positive edge

Motion functionality

- Latest measured value at a negative edge - Penultimate measured value at 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**
- Enabling the touch probe function through parameter "M-0-0024, Command: Activation",
 - Enabling at least one negative edge by setting bit 1 to bit 0 in the parameter "M1-0007, Control word",
 - Enable signal by setting "M-0-0405, Enabling".

M-0-0133, Time difference

Function The time difference between two probe edges is saved in this parameter in the unit micro seconds.

In case of regular differential measurement, it is the time difference between the last positive and the last negative edge.

If an extended difference measurement is configured, various combinations are possible:

- Time difference between the last positive and the last negative edge
- Time difference between the last positive and the next to last positive edge
- Time difference between the last negative and the last positive edge
- Time difference between the last negative and the next to last negative edge

Since a time difference value is not formed with each new measured value, there is a separate counter "M-0-0411, Counter of difference values probes".

The time difference value is deleted with each release.

- Use**
- Activating the probe function through parameter "M-0-0024, Probe activation"
 - Activating at least one edge by setting bit 1 to bit 0 in parameter "M1-0007, Control parameter probe"
 - Apply enable signal by setting "M-0-0405, Enabling probe"

M-0-0140, Time stamp, positive edge

Function The time stamp of the last positive edge of the probe is saved in this parameter. The time stamp refers to the time interval between the edge and the start of the sercos cycle in which the edge was found. The time interval is represented in micro seconds.

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 of the last negative edge of the probe is saved in this parameter. The time stamp refers to the time interval between the edge and the start of the sercos cycle in which the edge was found. The time interval is represented in micro seconds.

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 μ s).

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 μ s).

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".

Motion functionality



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.



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.

Motion functionality

Structure	Value	Release of the touch probe	Comment
	0	No	
	1	Yes	

Tab. 7-14: 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".

M-0-1000, Command: Load basic touch probe parameters

Function This parameter is used as a command to load the basic parameters of the touch probe instance.

Use *Execution in the parameter editor:*

- Enter the binary value "0000000000000011" (=0x0003) in this parameter. This triggers the loading of the basic parameters.
- After the loading process is complete, the control automatically resets the parameter to "0000000000000000".

When using the parameter, e.g. from the PLC:

- Enter the binary value "0000000000000011" (=0x0003) in this parameter. This triggers the loading of the basic parameters.
- When the loading is complete, the control automatically resets the parameter to "0000000000000000".

M-0-2100, List of all backup operation M-parameters

Function This list includes all M-parameters to be considered when backing up parameters. It is entered in the ID number format.

The entries are sorted according to identity number in ascending order.

Structure The ID number format is as follows:

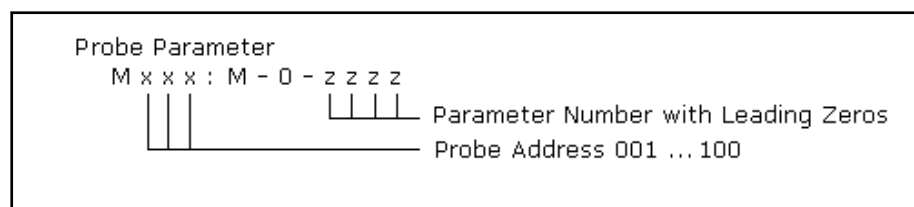


Fig. 7-74: ID number format

Use M0001: M-0-0002 - Probe 1; Parameter M-0-0002.

7.14.8 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

Motion functionality

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

Tab. 7-15: Function blocks for the touch probe functionality of the control hardware

For the description of these function blocks, refer to "Rexroth IndraLogic XLC IndraMotion MLC 13VRS PLCopen Libraries" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

7.15 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.

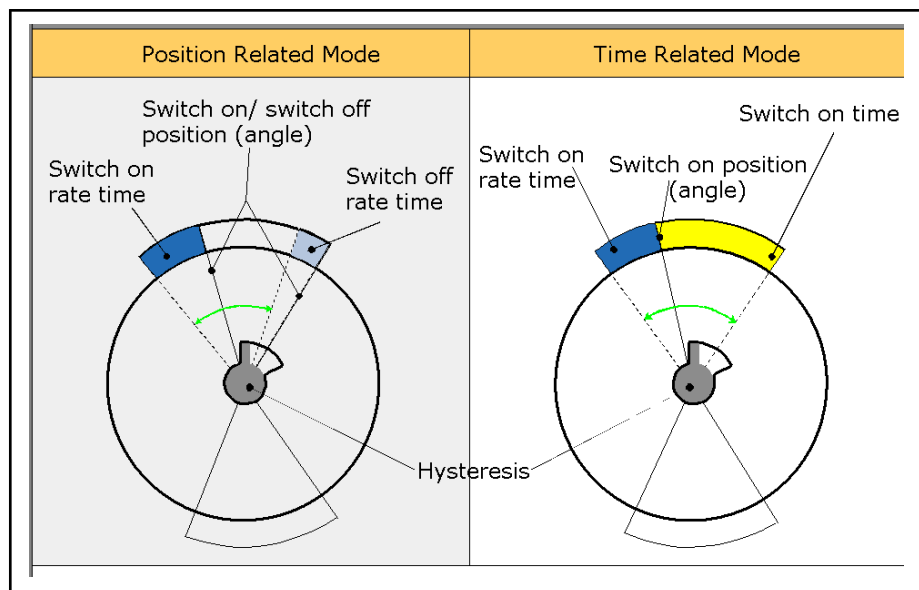


Fig. 7-75: Operating states

The programmable limit switch is available both as hardware and as software.

Motion functionality

**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 the "Rexroth IndraMotion MLC 13VRS Technology Basic Libraries" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

**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.



For IndraLogic XLC controls, function modules are only available with IndraControl variants with a Sercos interface.

For more information on the function module, refer to "Rexroth IndraControl Lxx 13VRS Function Modules" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

8 CamBuilder

8.1 CamBuilder functionality

The CamBuilder application is integrated in IndraWorks. The CamBuilder is described in detail in the "Rexroth IndraWorks 13VRS CamBuilder" documentation (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

9 Oscilloscope

9.1 Oscilloscope functionality

The oscilloscope functionality is integrated in IndraWorks. This function allows the signal recording of the control or axes (drives).

The use of the oscilloscope is described in detail in the documentation "IndraWorks 13VRS Engineering" in the chapter "Operating the oscilloscope function" (see [chapter 1.4 "Required and supplementing documentations" on page 18](#)).

10 Error cases

10.1 Introduction and overview

This chapter describes different error categories as well as troubleshooting.

General questions in case of error

The following information has to be provided:

1. When does the error occur?
 - Upon control start
 - Upon login with IndraLogic
 - When going online
 - After an online change
 - Upon phase switching of the Sercos bus
 - After opening/operating a visualization
 - After a program change
2. Can the error be reproduced?
 - Upon a certain action
 - After a certain period
 - Cyclically in a certain interval
 - Sporadically
3. Can an IndraWorks project archive be provided?
4. Providing control files
5. Which firmware and IndraWorks version?
6. What periphery is connected to the control?

10.2 Error categories

10.2.1 Exception in IEC application

Definition The error entry F0390011, followed by an error text, is output on the control display if there is an exception in an IEC application. This text provides information on the specific IEC task of the specific IEC application that caused the error.

Example for a division by 0 occurring in the "MotionTask" of the application "AppPresse1":

EXCEPTION DivisionByZero AppPresse1 MotionTask

Cause A common cause is an incorrect IEC user program. Such an error can be caused by the self-programmed user code or the PLC libraries.

When throwing an exception, the application is switched to the "STOP" state. Depending on the configuration of the PLC settings (see ["Safe state" on page 190](#)), the I/Os are either set to the safe state or to the configured initial values. Axes are stopped.

An application can only be started after an exception if the control was reset (cold, warm, origin).

Troubleshooting tools

Real cases have shown that exceptions are often caused due to an incorrect pointer programming and incorrect array access.

Most non-initialized pointers are dereferenced, thus causing a possible exception. When using arrays, array limits are often not kept in loop cycles and

Error cases

thus following data memory is overwritten. This can cause unpredictable aftereffects (SysError, control reboot).

By integrating special check functions, the complete IEC application can be monitored at runtime with regard to valid pointer access, division by 0 as well as falling below or exceeding array limits and can trigger a certain error response (division by 0) causing the application to stop. However, integrating these functions causes an increase in the cycle time of the PLC task. The cycle time rise depends on the application and increases with the number of access by pointers, arrays, divisions, etc.



During commissioning, integrate the check functions "CheckBounds" and "CheckPointer" to find the above described errors in the user program. After commissioning, remove these functions to reduce the PLC cycle times.

The implementation of the check functions is only an example and can be adapted to the application.

There are two options to check the application using check functions.

- Integrating the RIL_CheckRtv library into the library manager
All checks (pointer, arrays, divisions, etc.) are always performed by integrating this library. This increases the PLC program runtime significantly and might cause cycle time problems
- Targeted integration of check functions for certain checks
Add check functions to enable the checking of pointer, division and array access. The "RIL_CheckRtv" library may not be integrated into the project.

Check functions are added via the context menu of the "Application" node:
Add ► POU Templates ►

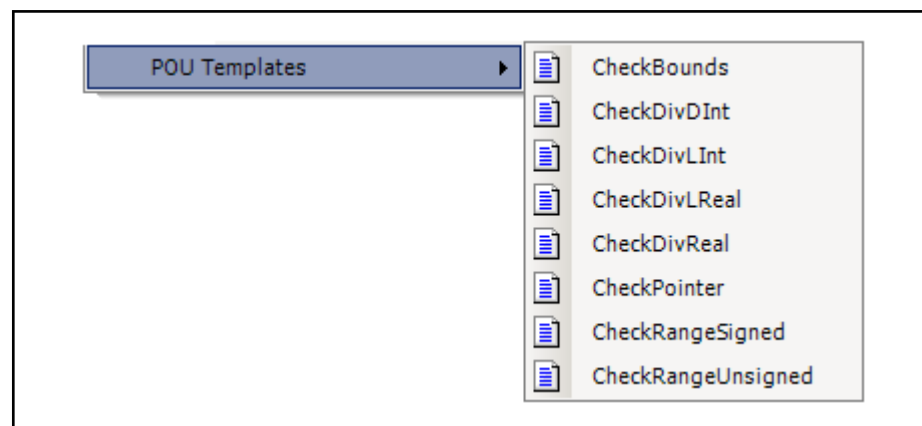


Fig. 10-1: Adding check functions via the PLC context menu

The following check functions are available:

- CheckBounds
Check index limits when arrays are accessed
- CheckDivInt
Check for division by 0 of INT-compatible data types
- CheckDivLInt
Check for division by 0 of LINT-compatible data types

Error cases

- CheckDivLReal
Check for division by 0 of LREAL data types
- CheckDivReal
Check for division by 0 of REAL data types
- CheckPointer
Check pointer access to NULL
- CheckRangeSigned
Check for signed, value-limited variables
- CheckRangeUnsigned
Check for unsigned, value-limited variables

If there is an exception, the error location in the IEC user program is displayed if this location is not in a compiled library or in FB_Init/Exit methods.

Starting from the check function, the call sequence can be retrieved, variable values checked and adapted to the user program via the call stack (menu "Debug/Call List").

10.2.2 SysError

Definition All errors not occurring in an IEC user task (e.g. MotionTask, PlcTask) cause a SysError.

The "SysError" text is output on the control display, followed by an 8-digit error code starting with "F9" as well as by the first eight characters of the name of the task in which errors were caused.

If the control is in the SysError state, all I/Os are switched to the safe state. All axes are stopped. No communication is possible with the engineering tool, IMST, HMI devices, etc.

Cause This error is normally a software error in the firmware.

It can also occur if the periphery (e.g. function modules, I/O) responses incorrectly and if unexpected states occur in the firmware components.

In certain cases, a SysError can also be caused by the user program in the IEC application if the error does not occur in an IEC user task (e.g. MotionTask, PlcTask).

This is the case if there is a division by 0 in an FB_Init method. A SysError (in the communication task "BlkDrvTcp" or "BlkDrvUdp") is already caused during application download and will be present again after the control restart (then in the PLC initialization task "StartTask").

Data memories overwritten by the IEC user program can be another reason. This is caused for example by overwriting array limits in case of array access (writing).

The following operations and functionalities can cause a SysError:

- Downloading application
- Making online change
- Monitoring functionality (forcing variables)
- Operating the integrated commissioning visualization
- Loading a boot project after control restart

Troubleshooting tools

1. "ExcData" is displayed after restart if a SysError occurred. During that period, an "ExcData" directory is created on the storage medium on the

Error cases

USER partition with the following files required for the support to trouble-shoot:

- PLC boot application files *.app and *.crc
 - PLC configuration files *.cfg
 - PLC retain memory file PlcRetain.save
 - Exception log file ExcLog.dat
 - NV-RAM image Para.bin
 - Log swap file LogBook.bin
2. The control can load this directory from the context menu of the "Control" node in the engineering tool using the device data explorer to provide the directory to the Bosch Rexroth Support.
 3. If there is again a SysError after control restart, reboot the control. Before backing up the error-relevant files ("ExcData" display), the BOOT-STOP menu can be opened to perform further measures. Press "Esc" and <Enter> at the control display until "BOOTSTOP" is shown in the display.

See also [chapter 10.3 "Procedure in case of SysError" on page 438](#).

10.2.3 Spontaneous control reboot

Definition In case of this error, the control starts again at runtime without outputting an error text.

Cause The main causes are insufficient power supply of the control, EMC affects, wiring errors or a control hardware error.

A reboot can also be initiated if the memory is overwritten by a defective user program (e.g. writing across array limits) or a firmware error.

To reduce the number of causes created by an incorrect user program, integrate check functions (see [chapter 10.2.1 "Exception in IEC application" on page 435](#)).

10.3 Procedure in case of SysError

If a SysError (F9 error) occurs, provide troubleshooting information as follows.

If the error display in the control display flashes (e.g. "Sys-Err F9240001 BlkDrvUd"), switch off the control and switch it on again.

After boot phase 1.04, "ExcData" is shown in the control display. All required files are collected during this phase and saved in the "ExcData" folder on the USER partition of the CF card.

Booting the control continues.

If no errors occurred during control startup, the backed up files can be loaded from the "ExcData" folder to the system partition via FTP by the control and provided to the Bosch Rexroth Support for troubleshooting purposes.

The application can be subsequently loaded and started via the advanced display menu.

If the control startup ends again in a SysError, go to the BOOTSTOP menu after switching off and on and enable the "Deny load boot application(s)" option.

If the startup is now trouble-free, the error cause is in the PLC boot project. For troubleshooting, load the "ExcData" folder to the user partition using the control and forward it together with the IndraWorks project archive to the Bosch Rexroth Support.

If the startup ends again with a SysError, delete boot application and PLC configuration via the BOOTSTOP menu. After startup, the PLC application is applied to the control. Retain and persistent data are initialized again.

If this is also not successful, reset the control to delivery state. Start CLEAN-UP via BOOTSTOP menu (see [chapter 3.3.3 "Deleting control memory" on page 37](#)).

Load the control configuration and the PLC application again to the control. Retain and persistent data are initialized again.

10.4 BOOTSTOP Menu

When reaching the BOOTSTOP menu, the control startup of the IndraLogic XLC is stopped. Different commands can be executed and the control startup can subsequently be continued without the control being switched off and on.

After the control (CMLx5) is switched on, press the outer buttons <Esc> and **Enter** below the display when BOOT 1.02 is displayed and keep them pressed until BOOTSTOP is displayed.



If the control was switched on again after a SysError, data is automatically backed up. During this data backup, the control display shows the text "ExcData". It is possible to go to the BOOTSTOP menu before.

Press (at least 3 seconds) individual keys or shortcuts to open the available commands.

A selected command is started after pressing "Enter" (at least 1 second).

A selected command can be canceled after pressing <Esc> (at least 1 second). The display returns to the BOOTSTOP menu.

If a command is executed, "SUCCESS" or "ERROR" is displayed for three seconds. The display returns to the BOOTSTOP menu if the executed command does not cause an automatic control reboot.

Error cases

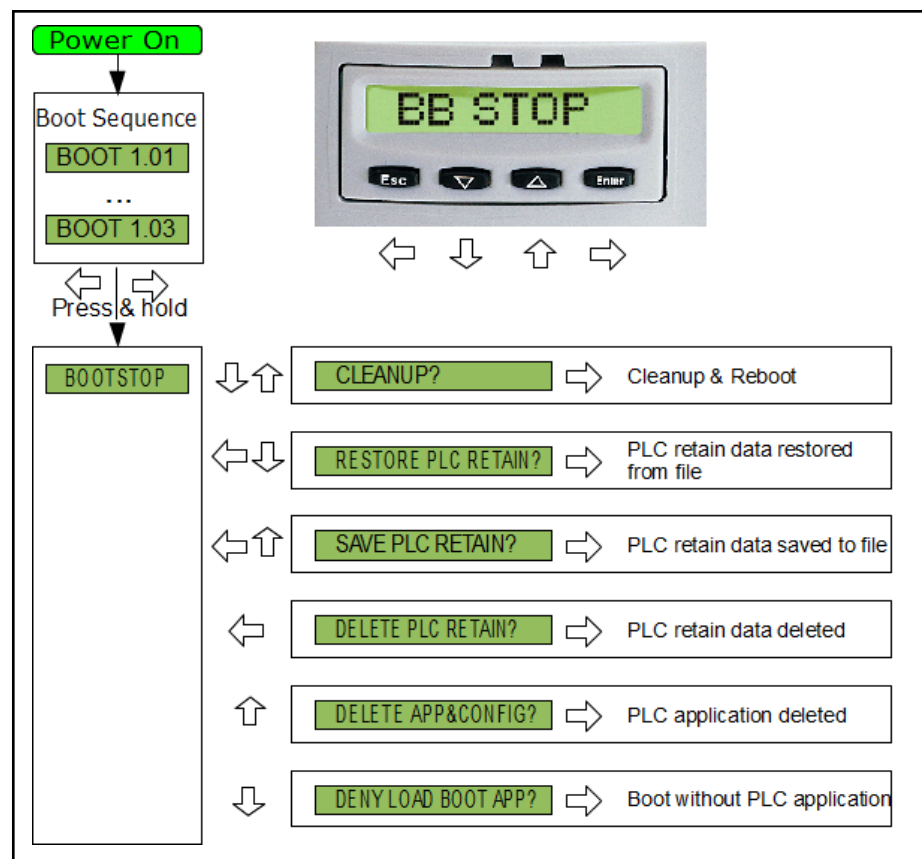


Fig. 10-2: BOOTSTOP menu structure

The following commands are available:

- Continue system startup

Keys: <Esc>, <Enter>

Command display in the display: "CONTINUE STARTUP?"

The system continues the control boot
- Reset control

Keys: <Up>, <Down>

Command display in the display: "CLEANUP?"

The control is reset. Parameterizations, boot project files as well as the complete NV RAM memory of the control are deleted (see [chapter 3.3.3 "Deleting control memory" on page 37](#))
- Backing up retain and persistent retain memory of the PLC

Keys: <Esc>, <Up>

Command display in the display: "SAVE PLC RETAIN?"

The complete PLC memory (retain, persistent retain) is saved in the "PlcRetain.save" file on the storage medium
- Restoring PLC retain and persistent-retain memory

Keys: <Esc>, <Down>

Command display in the display: "RESTORE PLC RETAIN?"

The complete PLC memory (retain, persistent retain) is restored from the "PlcRetain.save" file



To restore the retain memory, it has to be loaded subsequently with the corresponding boot application. Otherwise, the loaded boot application is switched to the "STOP" state with the exception Retainmismatch. Execute a "cold reset" or a "reset origin" to force the initialization of the retain or persistent retain data.

- Deleting PLC retain and persistent retain memory
Key: <Esc>
Command display in the display: "DELETE PLC RETAIN?"
The complete PLC memory (retain, persistent retain) is deleted. Loading a boot application subsequently using a retain or retain persistent memory, it is switched to the "STOP" state with the exception "Retainmismatch". Execute a "cold reset" or a "reset origin" to force the initialization of retain or persistent retain data and to allow the application start
- Deleting PLC application and configuration files
Key: <Up>
Command display in the display: "DELETE APP& CONFIG?"
When deleting the application files and the PLC configuration file on the data carrier, no PLC application can be loaded at next control reboot. This command can be useful if an exception occurs in the initialization part of a PLC application (e.g. FB_Init) that will cause a SysError (F9 error)
- Deny loading PLC boot application(s)
Key: <Down>
Command display in the display: "DENY LOAD BOOTAPP?"
At next control boot, a boot application (application files and valid entry in the PLC configuration file) on the storage medium is not loaded. This command can be useful if a SysError (F9 error) occurs when booting the control. If the error occurs in the PLC initialization task "StartTask", loading the boot application can be denied by this command to check whether the control boot is successful. If a SysError occurs in the "StartTask", it is not caused in the boot project, but in the PLC initialization. Contact the Bosch Rexroth Support

10.5 Core Dump

Core dump refers to the backup of the complete RAM on the CF card of the control. This is only required for certain errors and requested by the support.

The control triggered a SysError. The control display shows a SysError alternating with the error code.

1. Initiating a core dump.

Press <Up> and <Enter> simultaneously for at least 5 seconds to create a core dump.

The control reboots automatically after a core dump is initiated.

2. Core dump backup.

The core dump is stored on the CF card on the /ata0b/ partition after reaching the phase BOOT1.04. Attention: Already stored core dumps are overwritten without prompt! The archiving progress is shown on the display.

Error cases

After a successful core dump backup, COREDUMP FINISHED is displayed and booting is paused.

If the core dump backup was not successful, COREDUMP FAILED is displayed and booting is paused.

3. Control restart.

Irrespective of whether the core dump backup failed or was finished successfully, restart booting with <Esc> to initialize the control again.

The control boots.

4. Core dump reading.

The previously generated core dump can now be retrieved via the IndraWorks File Explorer or an FTP client. The file name is "vxcore1" and can be found on the /ata0b/ partition. It is also reasonable to back up the files "ExcLog.dat" and "ExcLog.bak".

User-defined languages for parameter names, units, diagnostics and menu texts

11 User-defined languages for parameter names, units, diagnostics and menu texts

11.1 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.

11.2 File management

For the online application, store the files on the user partition of the Compact Flash card in the folder "Languagefile".



If the control firmware is updated, all data is deleted on the user partition!

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.

11.3 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. 11-1: 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.

User-defined languages for parameter names, units, diagnostics and menu texts

- The first node is "packagelist" and different packages are declared below
 - The packages contain nodes with "idx", "key" and "additional"
 - 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 packages have to be named as follows:
 - A: "Axis parameters"
 - C: "Control parameters"
 - K: "Kinematic parameters" (only IndraMotion MLC)
 - O: "Oscilloscope parameters"
 - N: "ProgLimSwitch parameters"
 - M: "Touch probe parameters"
 - 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 11.5 "Node names for units" on page 446](#).
- 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 "Menue"
 - 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](#)
- [Node Names for Display Texts](#)

11.4 Function

Select the language via the parameter C-0-0001.

Furthermore, German, English, user-defined and "empty" (easier editability) XML files can be created for parameters, diagnostics and display.

User-defined languages for parameter names, units, diagnostics and menu texts

C-0-0001 can assume the following values:

- 0: Loads German from the installation storage location
- 1: Loads German 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 limited to 60 characters, diagnostics are limited to 61 characters and units are limited to 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.
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.

User-defined languages for parameter names, units, diagnostics and menu texts

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.

11.5 Node names for units

No.	Node name	Unit English	No.	Node name	Unit German
1	Unit_1_100K	1/100K	42	Unit_MasterUnit	nicht_aktiviert (not_enabled)
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/rev
11	Unit_DrehMomKonst_Lin	N/A	52	Unit_mNms2_rad	mNm/(rad/s ²)
12	Unit_Grad	Degree	53	Unit_ms	ms
13	Unit_Grad_s	Degree/s	54	Unit_N	N
14	Unit_Grad_ss	Degree/s ²	55	Unit_N_Aeff	N/Aeff
15	Unit_Grad_sss	Degree/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/rpm
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	Pole pairs
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/rev	68	Unit_ss	s ²
28	Unit_Inkr	Incr	69	Unit_TP	TP
29	Unit_Inkr_ss	Incr/s ²	70	Unit_TP_U	TP
30	Unit_Inkr_sss	Incr/s ³	71	Unit_Traegheit_Lin	Kg
31	Unit_Inkr_Umdr	Incr/rev	72	Unit_U	rev

User-defined languages for parameter names, units, diagnostics and menu texts

No.	Node name	Unit English	No.	Node name	Unit German
32	Unit_inlbf	inlbf	73	Unit_U_min	rpm
33	Unit_kBaud	kBaud	74	Unit_U_s	rev/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	Vel * 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)			

11.6 Node names for menu texts (display texts)

No.	Node name	Text English	Characters
1	Menue_MIState	ML:OK	8
2	Menue_PlcState	PLC: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	Device	8
11	Menue_Info	Info	8
12	Menue_OHCCtrlFanReset	ResetOHC	8
13	Menue_Hardware	Hardware	8
14	Menue_Plac	PLC APP	8
15	Text_IpAddress	IP-Address	60
16	Text_SubnetMask	Subnet-Mask	60
17	Text_GatewayAddress	Gateway-Address	60
18	Text_MacAddress	Mac-Address	60
19	Text_Firmware	Firmware	60
20	Text_HwMatNr	Material number	60
21	Text_HwTypeCode	Type code	60
22	Text_HwMatIndexNr	Index	60
23	Text_HwSerNr	Serial number	60

User-defined languages for parameter names, units, diagnostics and menu texts

No.	Node name	Text English	Characters
24	Text_Temperature	Temperature	60
25	Text_OHCctrl	OHC Control	60
26	Text_OHCctrlFan	OHC Fan	60
27	Text_ShowDHCPState	DHCP	60
28	Text_ShowDHCPStateEnabled	ENABLED	60
29	Text_ShowDHCPStateDisabled	DISABLED	60
30	Text_ChangeDHCPStateEnable	ENABLE	60
31	Text_ChangeDHCPStateDisable	DISABLE	60
32	Text_DHCPisActiv	DHCP is enabled, no changes possible	60
33	Text_LoadBaseParameter	Command: Load Network default	60
34	Text_LoadBaseParameterDone	Command execution done	60
35	Text_Axis	Axis	60
36	Text_Sercos	Sercos-Device	60
37	Text_Kinematic	Kinematic	60
38	Error_IpAddress	ERROR: Getting IP-Address	60
39	Error_SubnetMask	ERROR: Getting Subnet-Mask	60
40	Error_GatewayAddress	ERROR: Getting Gateway-Address	60
41	Error_MacAddress	ERROR: Getting Mac-Address	60
42	Error_BaseParameter	ERROR: During execution of command	60
43	Error_SerialNumber	ERROR: Getting serial number	60
44	Error_OHCctrl	ERROR: Getting OHC STRG time	60
45	Error_Temperature	ERROR: Getting temperature	60
46	Error_TemperatureScaling	ERROR: Getting temperature scaling	60
47	Error_FwVersion	ERROR: Getting firmware version	60
48	Error_TypeCode	ERROR: Getting type code	60
49	Error_Plc	ERROR: Getting PLC application name	60

12 Remote access via modem

12.1 General information

12.1.1 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

Tab. 12-1: 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 12.1.2 "Connection via internet" on page 449](#)
- To connect point-to-point (analog or digital telephone network, GSM or UMTS mobile network), see [chapter 12.1.3 "Point-to-point connection" on page 449](#)

More information

Description	Parts number
Rexroth IndraWorks I-Remote, Remote Control Software	R911310612

Tab. 12-2: More information

12.1.2 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.

There are many tunnel variants (for instance: PPTP, OpenVPN, IPsec). These different tunneling procedures are based on different protocols and differ in configuration complexity and security.

12.1.3 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

Remote access via modem

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.

12.1.4 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 12.2.4 "Possible remote maintenance tasks" on page 451](#)
- For a dial-up connection between service PC and MoRoS, see [fig. 12-2 "Dial-up connection between service PC and MoRoS" on page 452](#)

12.2 Dial-up connection between service PC and client PC

12.2.1 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. 12-1 "Dial-up connection between service PC and client PC" on page 450](#).

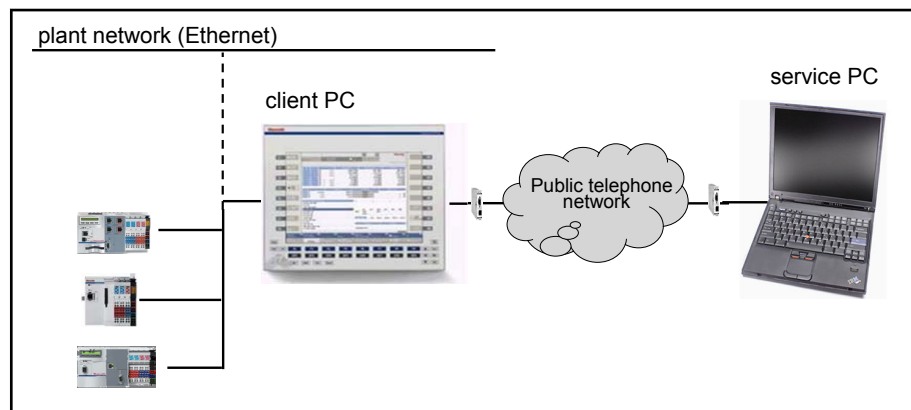


Fig. 12-1: Dial-up connection between service PC and client PC

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.

12.2.2 Required equipment

To operate a dial-up connection between service PC and client PC, the following equipment is required:

- Suitable remote maintenance software

Remote access via modem

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

12.2.3 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 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.

12.2.4 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.

12.3 Dial-up connection between service PC and MoRoS

12.3.1 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. For an exemplary connection establishment, see [fig. 12-2 "Dial-up connection between service PC and MoRoS" on page 452](#).

Remote access via modem

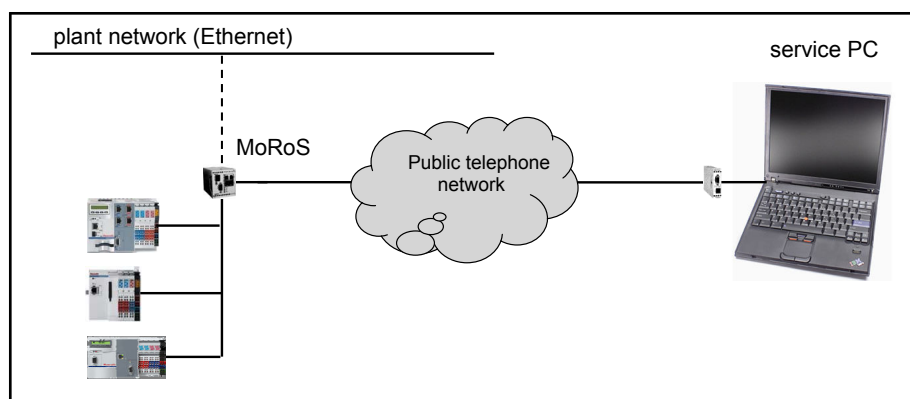


Fig. 12-2: Dial-up connection between service PC and MoRoS

12.3.2 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

12.3.3 Remote maintenance tasks

In general, all IndraWorks online functions are possible. Not included are drive commissioning and drive diagnostics via the serial interface.

12.3.4 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, the PC has 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".

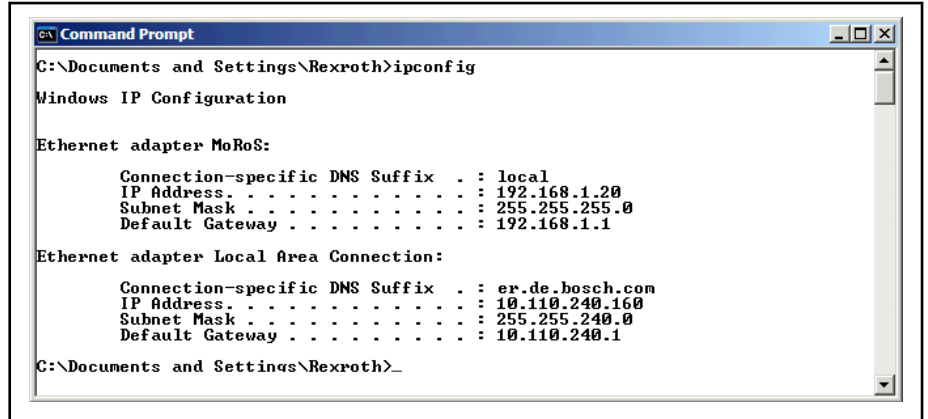


Fig. 12-3: Output of the "ipconfig" command in the command prompt at set up MoRoS

If no other addresses are displayed, use these for configuration.

If no useful addresses are displayed, check the cable connection between PC and MoRoS first. If the cable connection is 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.

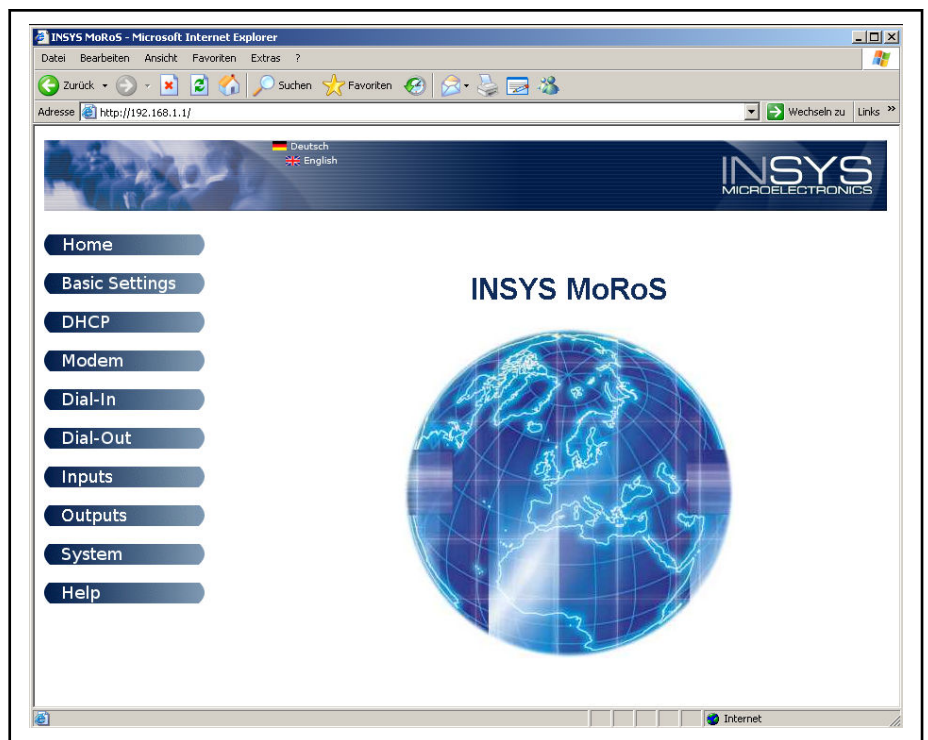


Fig. 12-4: Homepage of the MoRoS configuration interface

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

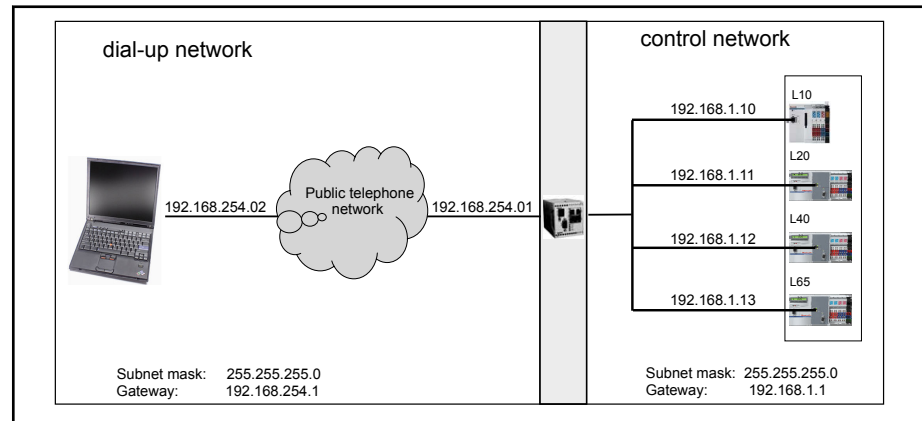


Fig. 12-5: Coupling two spatially separated subnets via router with built-in modem (MoRoS)

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 by the MoRos.

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

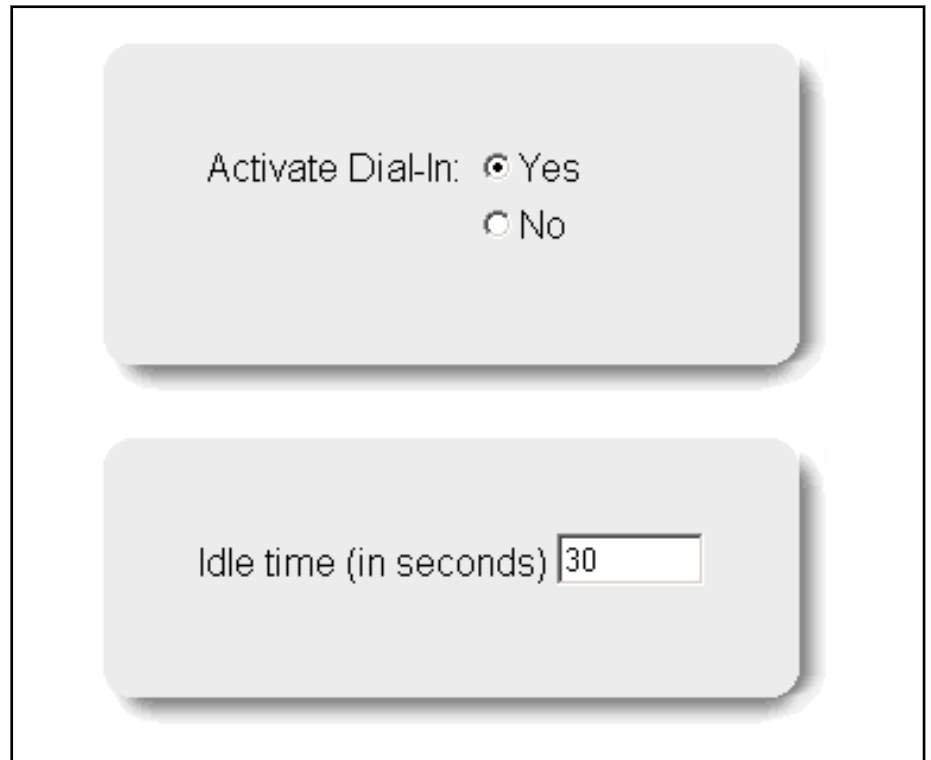
Fig. 12-6: DHCP settings

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 also set in this dialog if not specified by the device dialing-in.

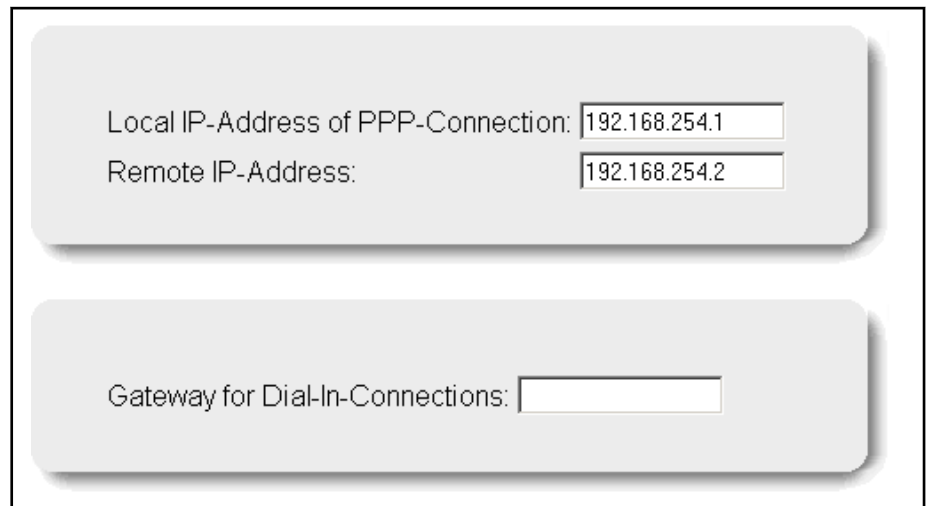


Activate Dial-In: ☒ Yes
☐ No

Idle time (in seconds)

Fig. 12-7: Enable incoming connection

IP settings



Local IP-Address of PPP-Connection:
Remote IP-Address:
Gateway for Dial-In-Connections:

Fig. 12-8: IP settings of the dial-up connection

Remote access via modem

Access data

Fig. 12-9: Access data of the incoming connection

Network configuration of the controls

The IndraLogic XLC controls use static IP addresses. Refer to [chapter 3.3.4 "Setting the IP address" on page 38](#) on how to change the network settings of the control.

In the example from [fig. 12-5 "Coupling two spatially separated subnets via router with built-in modem \(MoRoS\)" on page 454](#), 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

Tab. 12-3: 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. 12-10: Starting window of the connection wizard

3. Select connection type and method in the following dialogs.

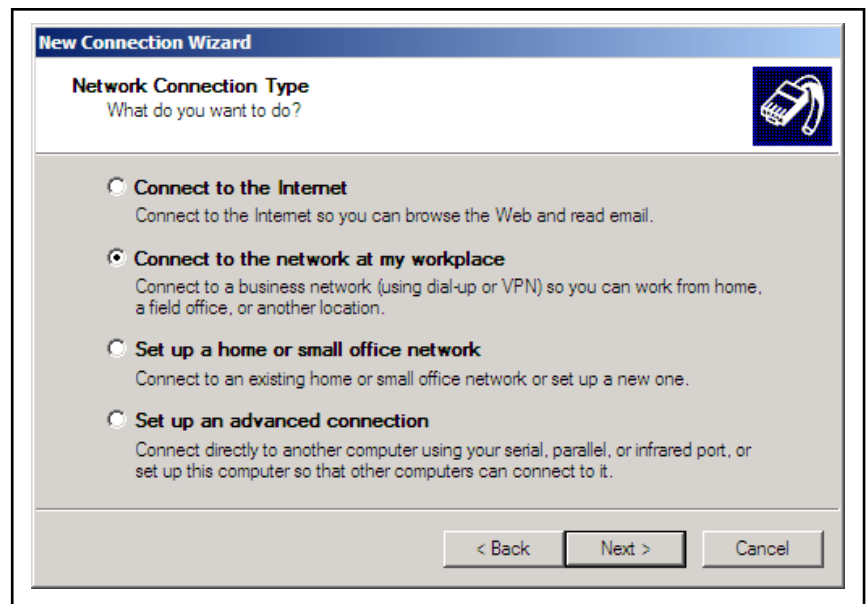


Fig. 12-11: Selecting network connection type

Remote access via modem

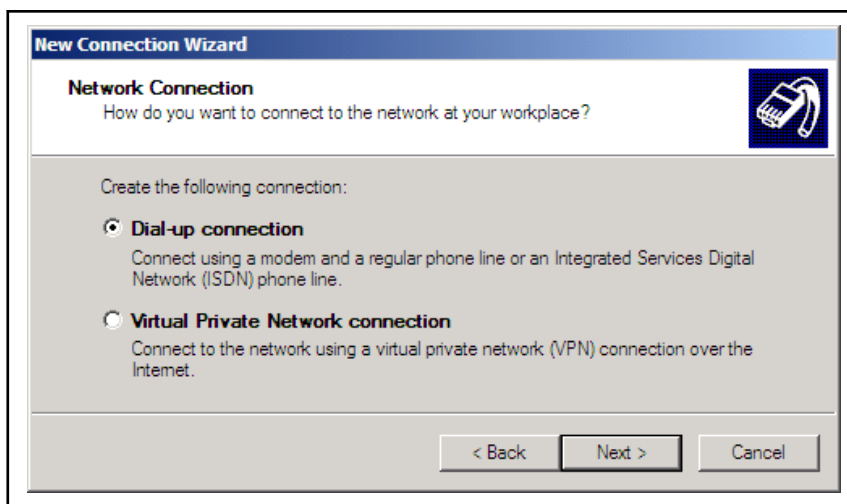


Fig. 12-12: Selecting network connection

4. Specify a name for the connection to be established.

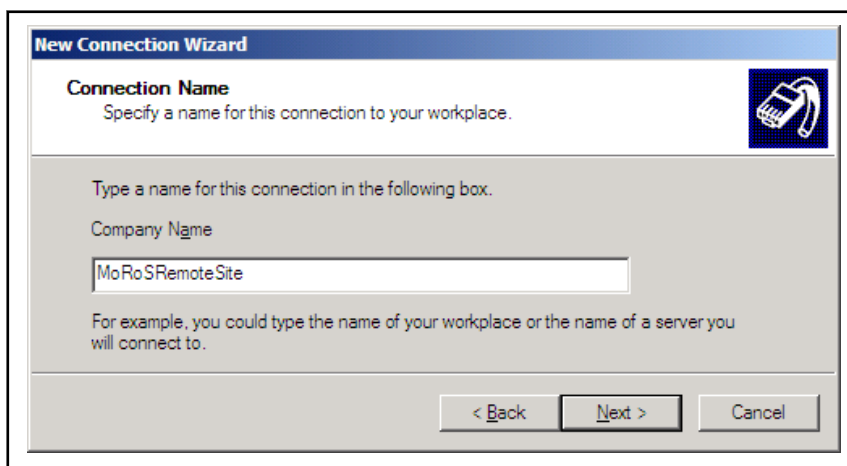


Fig. 12-13: Specifying connection name

5. Enter the phone number under which MoRoS is reachable.



Fig. 12-14: Entering phone number

6. Close the wizard with **Finish**.

Remote access via modem

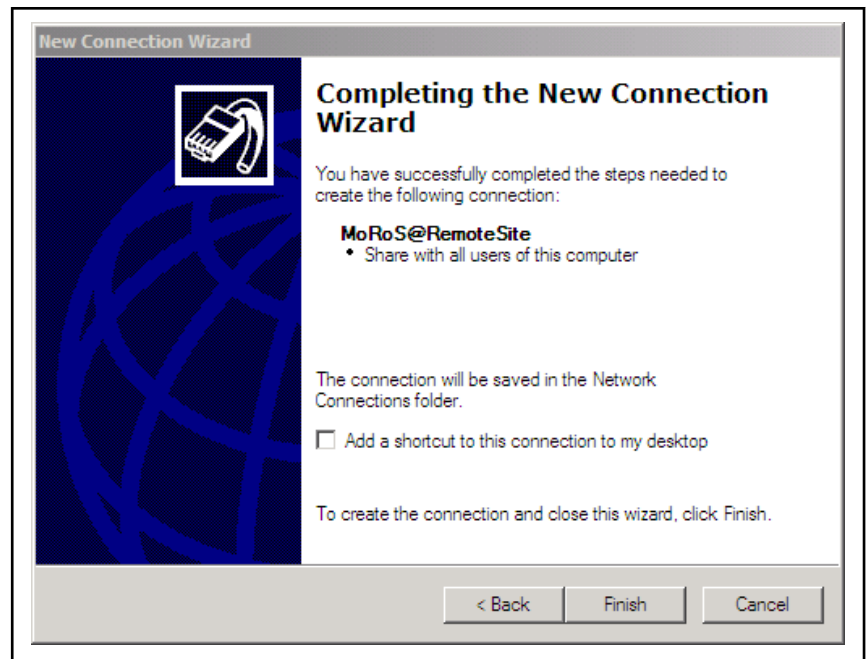


Fig. 12-15: Closing connection wizard

7. The "Connect" dialog opens. Access data can now be entered.

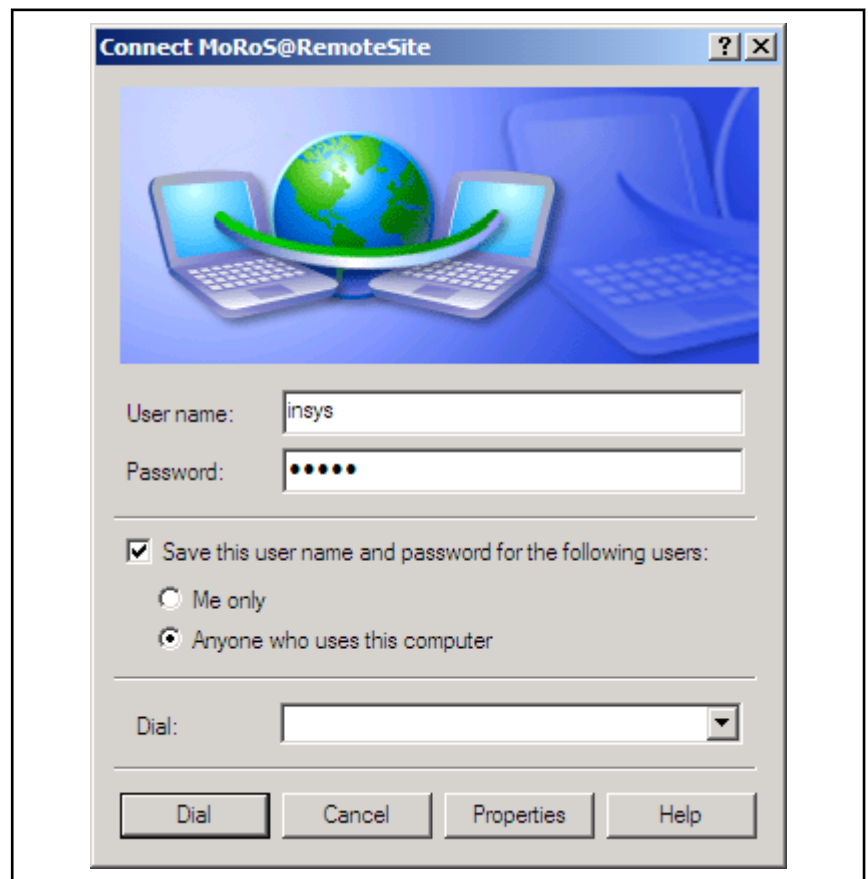


Fig. 12-16: Entering access data

The connection is established via the **Dial** button.

Remote access via modem

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 [tab. 12-3 "Network configuration of the controls" on page 456](#)) 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.

13 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 Hotline** and **Service Helpdesk** 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
- Type plate specifications of the affected products, in particular type codes and serial numbers
- Your contact data (phone and fax number as well as your e-mail address)

Index

A

- Abbreviations..... 24
- About this documentation..... 15
 - Validity of the documentation..... 15
- Activate watchdog..... 184
- Active
 - Watchdog..... 184
- Add POU to task..... 184
- Adjust command/actual configuration..... 329
- Adjust the behavior of the axis synchroniza-
tion dialog..... 335
- ANSI Z535.6-2006..... 16
- Archive project..... 48
- Axes – Introduction and overview..... 207
- Axes in IndraMotion
 - Operation modes for synchronous mo-
tions with electronic gear function..... 351
 - Single-axis modes..... 344
- Axis
 - Active..... 341
 - Deactivated..... 340
 - Parameterization mode..... 344
 - Parked..... 340
- Axis - dialog
 - Scaling/units extended..... 246
- Axis - Dialog
 - Cam table, gear settings..... 265
 - Dimensional reference - Optional encoder.. 253
 - Drive Halt..... 251
 - E-STOP function..... 256
 - Initial values..... 259
 - MotionProfile, gear settings..... 271
- Axis - Dialogs
 - Overview..... 234
- Axis – context menu
 - Diagnostics, error/diagnostic memory..... 306
- Axis – Context menu
 - Communication, cyclic Sercos data chan-
nel..... 299
 - Communication, signal control word..... 302
 - Communication, signal status word..... 300
 - Diagnostics..... 304
 - Diagnostics, average value filter display..... 307
 - Diagnostics, clear errors..... 306
 - Diagnostics, IDN list of invalid operating
data..... 305
 - Diagnostics, status..... 304
 - Initial commissioning..... 296
- Axis – Dialog
 - Axis configuration..... 290
 - Brake..... 242
 - Cam table, cam table phase offset..... 266
 - Cam table, cam table selection..... 267
 - Cam table, dynamic phase offset..... 267
- Cam table, dynamic synchronization -
Phase synchronization..... 268
- Cam table, master axis position offset..... 265
- Closed-loop control..... 249
- Compensation functions/corrections..... 261
- Compensation functions/corrections,
backlash on reversal correction..... 262
- Drive-integrated command value generator. 263
- Error reaction can be switched off..... 254
- Error response..... 254
- Error response, drive..... 254
- Error response, power section..... 255
- Establish dimensional reference..... 252
- FlexProfile..... 275
- FlexProfile, cam table phase offset..... 278
- FlexProfile, dynamic synchronization -
Phase synchronization..... 281
- FlexProfile, gear settings..... 278
- FlexProfile, general information..... 275
- FlexProfile, master axis position offset..... 277
- FlexProfile, overview..... 279
- Gear with phase synchronization - Gear
settings..... 288
- Gear with phase synchronization - Gen-
eral information..... 286
- Gear with velocity synchronization - Gear
settings..... 284
- Gear with velocity synchronization - Gen-
eral information..... 282
- Measuring systems - Settings motor en-
coder..... 242
- Measuring systems - Settings of optional
encoder..... 244
- Motion limit values..... 257
- MotionProfile..... 269
- MotionProfile, cam table phase offset..... 272
- MotionProfile, dynamic phase offset..... 273
- MotionProfile, dynamic synchronization -
Phase synchronization..... 274
- MotionProfile, general information..... 269
- MotionProfile, master axis position offset.... 270
- MotionProfile, MotionProfile overview..... 273
- Motor..... 240
- Motor temperature monitoring..... 241
- Operation mode settings..... 238
- Position switching point..... 292
- Settings of status messages..... 250
- Touch probe..... 293
- Axis (IndraDrive) - Dialog
 - Cam table..... 263
 - Cam table, general information..... 263
 - Mechanical gear..... 248
- Axis (IndraDrive) – Dialog
 - Compensation functions/corrections..... 261
 - Drive-integrated safety engineering..... 292

Index

Gear with phase synchronization - Dynamic synchronization.....	289	Configure the measuring encoder.....	231
Gear with phase synchronization - Master axis position offset.....	288	Container libraries.....	171
Gear with velocity synchronization - Dynamic synchronization.....	285	Context menu of the project folder.....	47
Gear with velocity synchronization - Master axis position offset.....	284	Control	
Axis commissioning.....	359	"Motion" node.....	87
Enable axis.....	361	Context menu items.....	94
FlexProfile.....	364	Function modules folder.....	91
Overview.....	359	Inline I/O folder.....	90
Phase offset.....	364	Onboard I/O folder.....	90
Position synchronization.....	363	Overview on nodes.....	82
Positioning.....	362	Profibus/M folder.....	90
Safety functions.....	360	Sercos node.....	91
Stop.....	365	Control - Context menu	
Velocity control.....	362	Diagnostics, task list/configuration.....	124
Velocity synchronization.....	363	Control - "Logic" node	
Axis error messages.....	367	Application – Toolbar.....	85
Axis Modes.....	340	Context menu.....	84
Axis name, encoder axis.....	230	Context menu, "Application" subfolder.....	86
Axis name, real axis.....	215	Subfolders.....	83
Axis name, virtual axis.....	224	Control - "Motion" node	
Axis number, encoder axis.....	230	Context menu.....	88
Axis number, real axis.....	215	Control - "Sercos " node	
Axis number, virtual axis.....	224	Context menu.....	92
Axis properties.....	220	Control - "Sercos" node	
Axis synchronization dialog.....	335	Sercos - Toolbar.....	93
Axis types.....	207	Control – Context menu	
Axis warning messages.....	374	Cam explorer.....	104
B		Communication, interfaces.....	101
Byte order.....	203	Communication, TCP settings.....	100
C		Communication, user control ID.....	101
Cam Pool.....	48	Compare/merge.....	151
Cam table.....	263	Compare/merge, axes.....	152
CamBuilder connection.....	431	Compare/merge, control.....	152
CMP ID.....	97	Compare/merge, function modules.....	152
Command value decoupling		Device data.....	139
Display.....	342	Device data, archive.....	139
Mode of operation.....	341	Device data, Explorer.....	145
Commission and operate.....	33	Diagnostic, MLPI connections.....	125
Commissioning and operation.....	169	Diagnostics, advanced properties.....	119
Compare control and project configuration.....	338	Diagnostics, clear errors.....	120
Complaints.....	25	Diagnostics, device status.....	117
Complete backup		Diagnostics, error/diagnostic memory.....	120
Complete backup.....	160	Diagnostics, invalid parameters.....	123
Complete data backup.....	160	Diagnostics, task list/configuration.....	124
Complete project backup		Export.....	148
Restore a project.....	165	Import.....	149
Configuration file		Language settings.....	103
Network settings.....	41	Parameters.....	109
Configuration task.....	184	Print.....	155
Configure cyclic data, Sercos III drive as PLC device	311	Print preview.....	153
		Runtime licenses.....	153
		Simulation.....	147
		Synchronize, compare control and project configuration.....	338
		Control error messages.....	369
		Control IndraLogic XLC – Context menu	
		Device data, restore.....	143
		Control properties.....	157

Control warning messages.....	376
Create control	48
Create encoder axis (real master axis).....	228
Create axis In the Project Explorer.....	228
Create function module.....	52
Create Inline I/Os.....	56
Create Onboard I/Os.....	54
Create PLC program	
Compile program.....	170
Declare variables.....	169
Edit instruction.....	170
Load PLC program to control.....	171
Open PlcProg object.....	169
Create Profibus I/Os.....	63
Create Profinet I/O Controller.....	63
Create project.....	46
Create real axis.....	213
Create Sercos III I/Os.....	64
Create virtual axis (virtual master axis).....	222
Criticism.....	25
Customer Feedback.....	25
Cycle times.....	323
Cyclic	
Task.....	184
Cyclic Sercos data channel.....	299
D	
Delete application.....	96
Device category.....	99
Device data	
Archive.....	139
Explorer.....	145
Restore.....	143
device editor	
Log.....	97
Device editor.....	96
Applications.....	96
Information.....	99
PLC settings.....	98
Device name.....	99
Device status.....	117
Device type.....	99
Device version.....	99
Device-specific library.....	171
Diagnostics	
Profinet I/O controller.....	90
Diagnostics, error/diagnostic memory	
Area control.....	122
Area View.....	122
Context menu.....	123
Filters.....	121
Messages.....	120
Diagnostics, Error/diagnostic memory	
Overview.....	120
Documentation of library function blocks.....	171
Documentation structure.....	15
Drive address, Sercos.....	214, 230

Drive as PLC device.....	308
Drive firmware.....	136
Drive firmware management.....	136
Drive password – Change.....	298

E

Electronic cams.....	379, 401
Cam solutions.....	380
Electronic cam.....	382
FlexProfile, basic functionalities.....	383
FlexProfile, configuration.....	402
FlexProfile, configuration using CamBuilder	408
FlexProfile, configuration using IndraWorks	403
FlexProfile, configuration using PLC func-	
tion blocks.....	402
FlexProfile, event control.....	398
FlexProfile, Flex step.....	387
FlexProfile, motion laws.....	385
FlexProfile, profile configuration.....	384
FlexProfile, select coordinate system.....	391
FlexProfile, switch and synchronize.....	392
FlexProfile, synchronous operating states...	383
FlexProfile, time master.....	385
Electronic motion laws.....	383
Encoder axis - Context menus	
Menu items.....	232
Encoder axis - Dialogs	
Overview.....	232
Encoder axis - General information.....	228
Error cases.....	435
BOOTSTOP menu.....	439
Error categories.....	435
Introduction and overview.....	435
Procedure in case of SysError.....	438
Error causes	
Core dump.....	441
Error response of the axis.....	213
Event.....	184
Task.....	184
Event logging.....	97
Event task.....	184
Exchanging firmware.....	134
External event for tasks.....	184
External library.....	171

F

Feedback.....	25
Find parameters.....	113
Firmware management.....	134
Firmware update.....	134, 136
FlexProfile	
Centralized and decentralized cam concept	383
IndraDrive.....	275
Use cases.....	382
FlexProfile editor.....	402
FlexProfile, assign MotionProfile of an axis.....	401
Free-running task.....	184

Index

Function module error messages..... 371

G

General information on the Profinet I/O..... 63

General information, definition of terms and introduction..... 449

H

Hazard warnings..... 16

Helpdesk..... 461

Hidden libraries..... 171

Hotline..... 461

I

I/O..... 200

I/O configuration..... 200

IndraDrive..... 263

IndraLogic XLC – Add Sercos bus..... 41

IndraLogic XLC – Basic sequences, general information..... 33

IndraLogic XLC – Delete control memory..... 37

IndraLogic XLC Cross-over connection..... 41

IndraLogic XLC display and operating keys..... 38

IndraLogic XLC IP address – Initial setting..... 38

IndraLogic XLC L25/L45/L65 – Hardware commissioning..... 33

IndraLogic XLC Lxx – Control startup..... 34

IndraLogic XLC menu overview to operate display and keyboard..... 39

IndraWorks

Interface..... 45

IndraWorks – Startup..... 42

Information messages..... 374

Information on device..... 99

Initial Values - Real Axis..... 260

Install and start IndraWorks..... 42

Install IndraWorks

Technical requirements..... 42

Internal library..... 171

Interval for task..... 184

IP communication..... 68

Automatic IP address generation..... 76

Basics..... 70

Configure bridge..... 77

Default setting..... 70

Firewall..... 78

IndraWorks commissioning dialogs..... 74

More information on the use..... 79

Overview..... 68

Parameters..... 77

Ports..... 78

TCP..... 78

UDP..... 78

L

Late binding..... 171

Library..... 171

Library directory..... 171

Library management

Convert IndraLogic 1.x libraries..... 176

Create libraries..... 175

Install on the system and integrate into a project..... 173

Integrate external code..... 177

Library overview..... 172

Library versions..... 174

Referenced libraries..... 174

Unique access to library function blocks (namespace)..... 174

Library menu

Container..... 171

Device-specific reference..... 171

External..... 171

Library repository..... 171

Library version..... 171

Link axis - Dialog

Settings: scaling/units..... 245

Load basic parameters

IndraDrive..... 297

Load offline parameters to device..... 337

Log..... 97

Logger in runtime system..... 97

M

M-0-0000, Dummy, M-parameter..... 416

M-0-0001, Logic touch probe number..... 416

M-0-0002, Touch probe name..... 416

M-0-0005, Input selection..... 416

M-0-0006, Signal selection..... 417

M-0-0007, Control word..... 417

M-0-0009, Osci: list of all M-parameters that can be oscilloscoped..... 419

M-0-0010, List of all M-parameters..... 419

M-0-0016, Project identification number..... 419

M-0-0021, Status word..... 419

M-0-0024, Command: Activation..... 420

M-0-0130, Measured value, positive edge..... 421

M-0-0131, Measured value, negative edge..... 421

M-0-0132, Difference value..... 421

M-0-0133, Time difference..... 422

M-0-0140, Time stamp, positive edge..... 422

M-0-0141, Time stamp, negative edge..... 423

M-0-0200, Dead-time compensation, positive edge..... 423

M-0-0201, Dead-time compensation, negative edge..... 423

M-0-0202, Starting position of the expectation window, positive edge..... 423

M-0-0203, End position of the expectation window, positive edge..... 423

M-0-0204, Starting position of the expectation window, negative edge..... 424

M-0-0205, End position of the expectation window, negative edge..... 424

Index

M-0-0206, Maximum number of marker failures.....	424	MLs-E0Y0010.....	372
M-0-0224, Number of marker failures, positive edge.....	424	MLs-E0Y0011.....	372
M-0-0225, Number of marker failures, negative edge.....	425	MLs-E0Y0012.....	373
M-0-0401, Input status.....	425	MLs-E0Y0013.....	373
M-0-0405, Enabling.....	425	MLs-E0Y0014.....	373
M-0-0409, Counter of measured values, positive edge.....	426	MLs-E0Y0015.....	373
M-0-0410, Counter of measured values, negative edge.....	426	MLs-E0Y0100.....	373
M-0-0411, Counter of difference values.....	426	MLs-E0Y0101.....	373
M-0-1000, Command: Load basic touch probe parameters.....	427	MLs-E0Y0102.....	374
M-0-2100, List of all backup operation M-parameters.....	427	MLs-IOX0001.....	374
Max. number of tasks.....	186	MLs-IOX0002.....	374
Memory alignment.....	202	MLs-W0A0020.....	374
Memory organization.....	202	MLs-W0A0021.....	375
ML_AXISDATA_SM.....	300, 301, 303	MLs-W0A0022.....	375
ML_TechBase.library.....	32	MLs-W0A0200.....	375
ML_TechInterface.library.....	32	MLs-W0A0201.....	375
ML_TechMotion.library.....	32	MLs-W0A0202.....	375
MLPI connections.....	125	MLs-W0C0020.....	376
MLs-E0A0001.....	367	MLs-W0C0021.....	376
MLs-E0A0002.....	367	MLs-W0C0022.....	376
MLs-E0A0003.....	367	MLs-W0C0023.....	376
MLs-E0A0005.....	368	MLs-W0C0200.....	377
MLs-E0A0010.....	368	MLs-W0S0013.....	377
MLs-E0A0012.....	368	MLs-W0S0020.....	377
MLs-E0A0013.....	368	MLs-W0S0021.....	377
MLs-E0A0014.....	368	MLs-W0S0023.....	377
MLs-E0A0015.....	368	MLs-W0S0024.....	377
MLs-E0A0020.....	368	MLs-W0S0200.....	378
MLs-E0A0021.....	369	MLs-W0S0201.....	378
MLs-E0A0023.....	369	MLs-W0S0202.....	378
MLs-E0A0024.....	369	MLs-W0S0203.....	378
MLs-E0A0030.....	369	MLs-W0X0001.....	378
MLs-E0A0100.....	369	MLs-W0Y0010.....	379
MLs-E0C0001.....	369	MLs-W0Y0011.....	379
MLs-E0C0002.....	370	Modes for project editing.....	326
MLs-E0C0003.....	370	Modes for project editing.....	328
MLs-E0C0004.....	370	Transition between modes, mode switching.....	328
MLs-E0C0005.....	370	Monitor task.....	177
MLs-E0C0006.....	370	Motion configuration validation.....	
MLs-E0C0010.....	370	Error messages.....	367
MLs-E0C0014.....	370	Motion functionality.....	323
MLs-E0F0010.....	371	Motion mode.....	324
MLs-E0F0011.....	371	Parameterization level.....	325
MLs-E0F0012.....	371	Parameters.....	325
MLs-E0F0013.....	371	Programmable limit switch.....	428
MLs-E0F0020.....	371	Motion limit values.....	
MLs-E0S0010.....	371	Virtual axis.....	257
MLs-E0S0012.....	371	Motion tasks according to VDI2143.....	385
MLs-E0S0013.....	372	MotionProfile.....	
MLs-E0S0014.....	372	IndraDrive.....	269
MLs-E0S0022.....	372	Multi-device.....	126
		Brief description of the multi-device table....	127
		Configuration file to disable field bus slaves	133
		Introduction.....	126
		Multi-device functionality in detail.....	128
		Multi-tasking.....	177
		I/O turn-around times.....	189
		Task configuration.....	179

Index

Task configuration, configuration examples.....	181	Parameterization.....	317
Task editor.....	183	PLC programming.....	320
Task list/configuration.....	186	PowerSupply.....	317
Task Viewer.....	188	Priority.....	184
Multilingualism XLC/MLC.....	443	Task.....	184
File management.....	443	Processing types.....	397
File structure.....	443	Product phases.....	15
Function.....	444	Program call.....	184
Node names for menu texts	447	Project information.....	171
Node names for units.....	446	Properties.....	157, 220
		Purpose of this documentation.....	15
N		R	
Names.....	24	Ready contact.....	34
Names and abbreviations.....	24	Real axes, IndraDrive	
Namespace for library.....	171	Firmware.....	209, 210
Network settings via the configuration file.....	41	Function package.....	209, 210
Non-object-related information messages.....	374	Real axis - dialogs	
Non-object-related warning messages.....	378	Overview.....	216
Notes on how to use deactivated and park- ing axes.....	341	Real axis - general information.....	208
		Real axis – Context menu	
O		Menu items.....	219
OEM partition.....	31	Real axis (IndraDrive).....	208
Offline mode.....	326	Real axis (IndraDrive) - Context menu	
Offline parameterization.....	328	Parameters, Load basic parameters.....	297
Online mode.....	326	Parameters, save mode.....	298
Operating strategy.....	48	Parameters, save parameters in the drive... ..	298
Operation mode settings		Real axis (IndraDrive) - Dialog	
Real axis.....	238	Dimensional reference - Motor encoder.....	252
Ordering number of device.....	99	Real axis (IndraDrive) – Context menu	
Oscilloscope connection.....	433	Parameter, drive password – Change password.....	298
		Referenced libraries	171
P		Remote access via modem.....	449
Parameter editor.....	109	Dial-up connection between service PC and client PC.....	450
Parameter editor toolbar.....	110	Dial-up connection between service PC and client PC, installation and configuration	451
Parameter group.....	111	Dial-up connection between service PC and client PC, possible remote mainte- nance tasks.....	451
Parameter group toolbar.....	112	Dial-up connection between service PC and client PC, required equipment.....	450
Parameterization level.....	325	Dial-up connection between service PC and MoRoS.....	451
Parameterization Sercos III drive as PLC de- vice.....	309	Dial-up connection between service PC and MoRoS, installation and configuration.. ..	452
Parameters - Export.....	113	Dial-up connection between service PC and MoRoS, remote maintenance tasks.....	452
Parameters - import.....	115	Dial-up connection between service PC and MoRoS, required equipment.....	452
Parameters – Save parameters in drives.....	113	General information, accessibility to the control.....	450
Placeholder for library.....	171	General information, connection via inter- net.....	449
PLC logger.....	97	General information, point-to-point con- nection.....	449
PLC program			
Create.....	169		
PLC programming, Sercos III drive as PLC device.....	314		
Pointer addressing.....	203		
Pointer programming.....	202		
POU			
Assign to task.....	184		
Open from task editor.....	184		
POUs for task.....	184		
Power supply as PLC device.....	317		
Configure cyclic data.....	318		

Replace hardware.....	42
Repository.....	171
Response time	
Inline I/O.....	57
Onboard I/O.....	54

S

Safety alert symbol.....	16
Safety error messages.....	372
Safety instructions.....	16
Safety warning messages.....	379
Save mode.....	298
Save parameters in drive.....	298
Save project.....	48
Scan bus configuration.....	93, 333
Sensitivity	
Task watchdog.....	184
Sercos device error messages.....	371
Sercos device warning messages.....	377
Sercos III drive as PLC device.....	309
Service hotline.....	461
Signal control word.....	302
Signal status word.....	300
Signal words.....	16
Signatures	
Applications.....	96
Singleton event.....	184
Special motion laws.....	386, 387
Structure of a project.....	81
Suggestions.....	25
Support.....	461
switch online.....	335
Symbols.....	18
Synchronous operating states.....	383
System overview.....	27
Firmware.....	30
Hardware.....	27
Software.....	31
SYSTEM partition.....	31

T

Target groups.....	15
Target-specific library.....	171
Task configuration	
Max. number of tasks.....	186
Properties.....	186
Task configuration properties.....	186
Task editor	
Assign POU.....	184
Configuration dialog.....	184
Max. number.....	186
Order.....	184
Task list.....	184
Task list/configuration.....	124
Task list/Task Viewer.....	124
Task name.....	184
Task properties.....	184

Task system.....	177
CPU load of the control.....	198
System tasks.....	193
Toolbar "XLC/MLC Info".....	82
Touch probe.....	293, 408
Accuracy.....	411
Data acquisition.....	411
Expectation window.....	415
Features.....	409
Function blocks.....	427
Functional description.....	409
Introduction.....	408
Overview on M-parameters.....	416
Turn around times	
Profibus, Profinet and Inline I/Os.....	191
Turn-around times	
Onboard and Fast I/Os.....	192
Sercos III I/Os.....	191

U

Update	
Application list.....	96
Update options.....	31
User control ID.....	101
USER partition.....	30

V

Validate Motion configuration.....	365
Vendor name.....	99
Version control.....	128
Version control of the control.....	159
Introduction.....	159
Special features.....	159
Version of a library.....	171
Virtual axis - Dialogs	
Overview.....	225
Virtual axis – Context menu	
Menu items.....	226
Virtual axis – General information.....	222

W

Warning messages.....	374
Warnings.....	16
Watchdog for tasks.....	184

Notes

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



R911336352