

PLC Designer



Application Sample i700 - Lenze teachware case -----

Software Manual

EN



13479663

Lenze

1	About this documentation	3
1.1	Document history	5
1.2	Conventions used	5
1.3	Notes used	6
2	Safety instructions	7
3	Conditions	8
3.1	System requirements	8
3.2	Setting up communication to the Controller	8
4	What is the Application Sample?	10
4.1	Target of the Application Sample	10
4.2	Overview of the features of the Application Sample	11
4.3	Elements of the Application Sample	11
4.3.1	The L_SMC_AxisBasicControl block	11
5	Overview - The structure of the Application Sample	12
5.1	A11_ProgramCalls	13
5.2	A20_Visualisation	13
5.3	A70_POUs	14
5.4	A90_Resources	15
5.4.1	Use of the oscilloscope function: Example of AxisX	15
5.4.2	EPM_S202: Digital inputs	16
5.4.3	EPM_S303: Digital outputs	16
5.4.4	EPM_S400: Analog inputs	16
6	Opening the Application Sample	17
6.1	Creating a new project - opening the Application Sample	18
6.2	Updating the Controller in the project (optional)	18
6.3	Going online	19
6.3.1	Transferring the project to the control system - "logging in"	19
6.4	Downloading and starting the PLC program	19
6.5	Simulation	20
7	Getting started - operating the Application Sample	21
7.1	The buttons of the visualisation	22
7.2	Commissioning an axis	23
7.3	Motor data parameterisation	23
7.4	Parameterising the axis data	24
7.5	Axis control during manual operation: manual control	25
7.6	Axis control in the automatic mode: "CyclicMove"	26
7.7	Coupling/decoupling of an axis: "GearingAxisY"	27
7.8	Holding brake: Application "BrakeAxisZ"	28
7.9	EtherCAT diagnostics: "Diagnostic"	29
	Index	30

1 About this documentation

This documentation describes the sample project for the i700 servo inverter for commissioning a Lenze automation system. The used automation system consists of a PLC for "Controller-based Automation" and drive components connected (via a bus system).



Read the mounting instructions accompanying the controller first before you start working!

The mounting instructions include safety instructions which must be observed!



Tip!

Current information on the Lenze products can be found at: <http://www.Lenze.com>

This documentation is part of the "Controller-based Automation" manual collection. The manual collection consists of the following documents:

Documentation type	Subject
System manuals	System overview/sample topologies • Controller-based Automation • Visualisation
Communication manuals	Bus systems • Controller-based Automation EtherCAT® • Controller-based Automation CANopen® • Controller-based Automation PROFIBUS® • Controller-based Automation PROFINET®
Online helps/ Software manuals	Lenze Engineering tools • »PLC Designer«: Programming • »Engineer«: Configuring drives • »VisiWinNET® Smart«: Visualisation • »Backup & Restore«: Back up/restore data

Technical documentation for Lenze products

Further information on Lenze products which can be used in connection with "Controller-based Automation" can be found in the following documentation:

Mounting & wiring
 Mounting instructions • Controllers • Communication cards (MC-xxx) • I/O system 1000 (EPM-Sxxx) • Inverter • Communication modules
Using sample application/application template
 Online help/software manuals • Application Sample i700 - Lenze teachware case • Application samples - Inverter Drives 8400/Servo Drives 9400 • ApplicationTemplate Lenze standard • ApplicationTemplate PackML standard
Parameterisation, configuration, commissioning
 Online help/software manuals • L-force Controller • i700 servo inverter • Servo Drives 9400 HighLine/PLC/ - regenerative power supply module • Inverter Drives 8400 StateLine/HighLine/TopLine • I/O system 1000 (EPM-Sxxx)
 Online help/communication manuals • Bus systems • Communication modules

Icons

-  Printed documentation
-  Online help in the Lenze Engineering tool/
software manuals and communication manuals are provided as PDF files in the download area of the Lenze website.

Target group

This documentation addresses to all persons who plan, commission, and program a Lenze automation system on the basis of the Lenze "ApplicationTemplate" as part of the "Controller-based Automation".

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the Lenze device and the software version of the engineering tools installed (here: »PLC Designer«), the representation of the actual screen display may deviate.

Information on validity

The information in this documentation is valid for the following Lenze software:

Software	From software version
»PLC Designer«	3.9.x

About this documentation

Document history

1.1 Document history

Version			Description
1.0	09/2012	TD11	First edition
1.1	01/2015	TD11	Updated to the »PLC Designer« V3.9.

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal number	0x	For hexadecimal numbers, the "0x" prefix is used. Example: 0x60F4
Binary number	0b	For binary numbers, the "0b" prefix is used. Example: 0b00010111
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the drive onwards are marked accordingly in this documentation. Example: This function extension is available as from software version V3.0!
Program name	» «	»PLC Designer«
Window	<i>italics</i>	The <i>Message window...</i> / the <i>Options</i> dialog box...
Variable names		By setting <i>bEnable</i> to TRUE...
Control element	bold	The OK... button / The Copy... command / The Properties... tab / The Name ... input field
Sequence of menu commands		If the execution of a function requires several commands in a row, the individual commands are separated by an arrow: Select File→Open to...
Shortcut	< bold >	Use <F1> to open the online help.
		If a key combination is required for a command, a "+" is placed between the key identifiers: With <Shift>+<ESC>...
Hyperlink	<u>Underlined</u>	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	 5	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

About this documentation

Notes used

1.3 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	Danger of personal injury through dangerous electrical voltage Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	Danger of personal injury through a general source of danger Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Stop!	Danger of property damage Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Please observe the following safety instructions when you want to commission a controller or system using the »Engineer«.



Read the documentation supplied with the controller or the individual components of the system carefully before you start commissioning the devices with the »Engineer«!

The device documentation contains safety instructions which must be observed!



Danger!

According to today's scientific knowledge it is not possible to ensure absolute freedom from defects of a software product.

If necessary, systems with built-in controllers must be provided with additional monitoring and protective equipment complying with the relevant safety regulations (e.g. law on technical equipment, regulations for the prevention of accidents) in each case, so that an impermissible operating status does not endanger persons or facilities.

During commissioning persons must keep a safe distance from the motor or the machine parts driven by the motor. Otherwise there is a risk of injury by the moving machine parts.



Stop!

If you change parameters in the »PLC Designer« during an online connection to the device is established, the changes are directly accepted in the device!

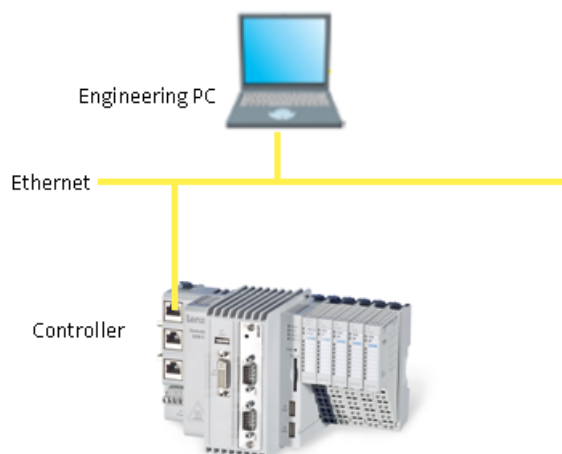
A wrong parameter setting can cause unpredictable motor movements. By an unintended direction of rotation, a too high speed, or jerky operation, the driven machine parts may be damaged!

Conditions

System requirements

3 Conditions

3.1 System requirements



	Engineering PC	Controllers
Hardware	PC/notebook	Controller 3200 C/p500 (Logic) from firmware V3.9 onwards
Operating system	Windows 7	Windows CE
Required Lenze software	»PLC Designer« from V3.9	Runtime software (Logic)
Further requirements	-	Bus system • EtherCAT bus system • EtherCAT node

3.2 Setting up communication to the Controller

- Connect the Engineering PC with the controller via a network cable. The »PLC Designer« accesses the controller via Ethernet.
- Make the IP settings with the »PLC Designer« afterwards.



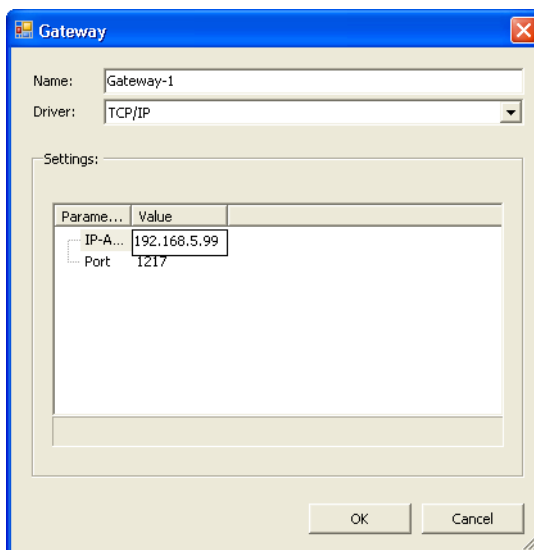
How to check the communication settings:

1. Go to the *Device* view and double-click the desired Controller.
2. Make the desired settings on the **Communication settings** tab.
 - Click the **Add gateway** button to insert a gateway.

Conditions

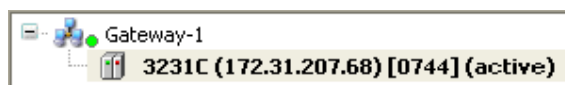
Setting up communication to the Controller

- Enter the desired IP address of the controller.



[3-1] Example: Enter the IP address of the Controller

3. Click OK to add the controller as gateway.
4. By double-clicking the desired channel (or clicking the **Set active path** button) set the channel selected in the device tree below the gateway as active path for control.
 - Thus, all communication actions directly refer to this channel.
 - The currently active path is represented in **bold** in the list and "(active)" is attached:



5. A device represented in *italics* is set as active path but has not been found during the last network scan.



Note!

- During initial commissioning, observe the following predefined IP addresses of the controllers:: **192.168.5.99**



Further information can be found in the following documentation:

- **Controller - Parameter setting & configuration**

What is the Application Sample?

Target of the Application Sample

4 What is the Application Sample?

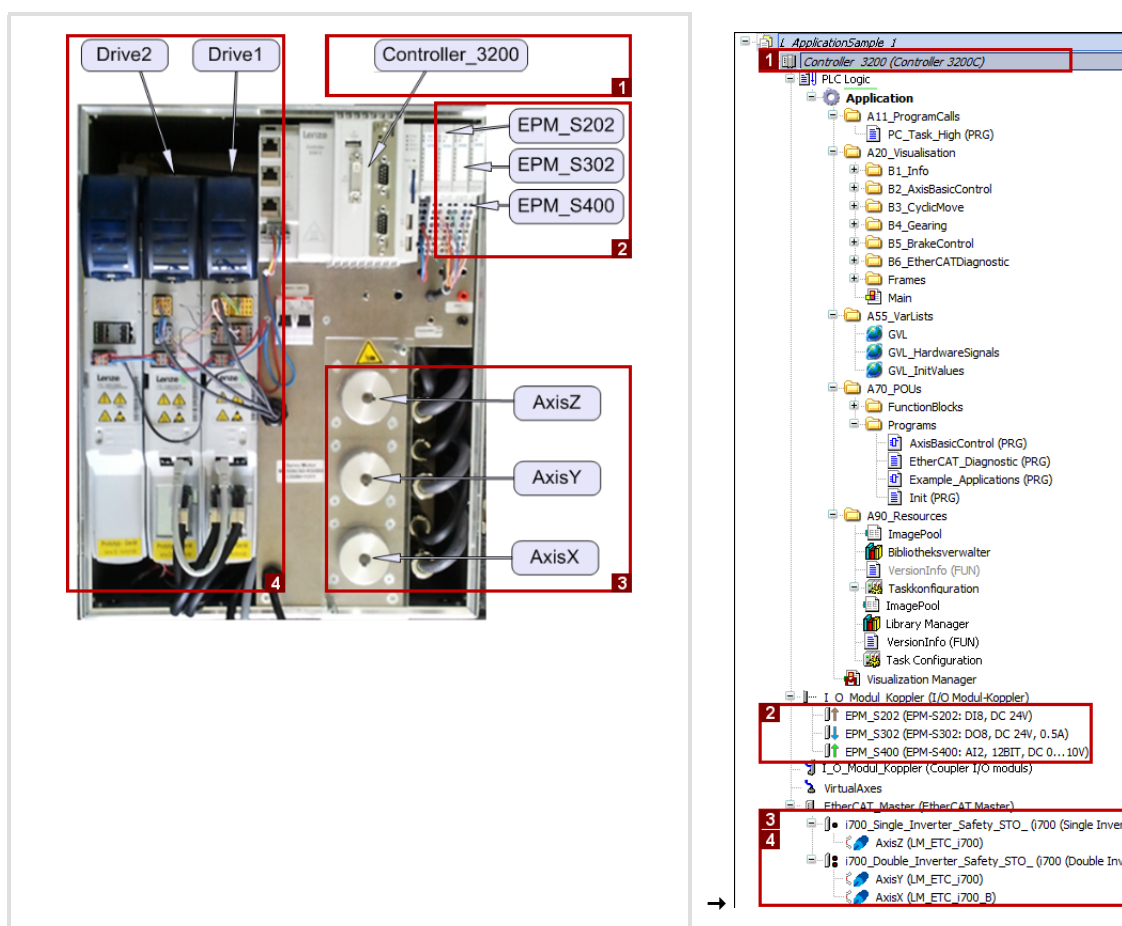
The "Application Sample i700" is a sample project in the »PLC Designer« for commissioning the i700 Lenze teachware case the purpose of which is to facilitate the introduction to the "Controller-based Automation" Lenze system.

- The ready-made sample project enables easy commissioning/diagnostics of the system which consists of a Controller (Controller 3200 C) with a connected I/O system 1000 and i700 servo inverters.

4.1 Target of the Application Sample

The Application Sample i700...

- ... facilitates the introduction to the Lenze "Controller-based Automation" on the basis of the Lenze Teachware case i700.
- ... includes simple examples of commissioning the i700 servo inverters.
- ... can optionally be extended by other hardware configurations.



[4-1] Structure of the Lenze Teachware case i700: It includes the Lenze devices contained in the **Application Sample i700**.

What is the Application Sample?

Overview of the features of the Application Sample

4.2 Overview of the features of the Application Sample

Visualisations

Folder/symbol	Function
AxisBasicControl	Visualisation for the quick commissioning of the controllers of the Lenze application case via the L_SMC_AxisBasicControl function.
CyclicMove	Example of application integration in the L_SMC_AxisBasicControl FB.
GearingAxisY	Example of integration of PLCopen functionality for coupling/decoupling: Second axis→first axis
BrakeAxisZ	Example of application integration of a holding brake by means of PLCopen functions.
Diagnostic	Example of diagnostics of the EtherCAT master and the status of the controllers.
	"Getting started": Information on the use/structure of the Application Sample

4.3 Elements of the Application Sample

The Application Sample ...

- ... enables easy and quick commissioning of a Lenze Automation system by means of examples.
- ... illustrates the possible applications of the **L_SMC_AxisBasicControl** function block.

4.3.1 The L_SMC_AxisBasicControl block

The **L_SMC_AxisBasicControl** block ...

- ... enables easy traversing of controllers by using PLCopen functionality.
- ... can be controlled via:
 1. The visualisation during manual operation
 2. The interfaces of the block (via a PLC program)

Controlling the L_SMC_AxisBasicControl block

The Application Sample shows the control possibilities by means of examples:

1. Example of commissioning via visualisation:

The  **AxisBasicControl** visualisation can be used to control the  **AxisBasicControl** program.

2. Example of controlling via the interfaces of the **AxisBasicControl** FB:

The  **ExampleApplications** program uses the following blocks which control the interfaces of the corresponding AxisBasicControl block (X axis, Y axis, Z axis):

- CyclicMove
- Gearing
- Brake-Control

The "programs" are created as function blocks to illustrate by means of an example how software functions can be reused in the »PLC Designer«.

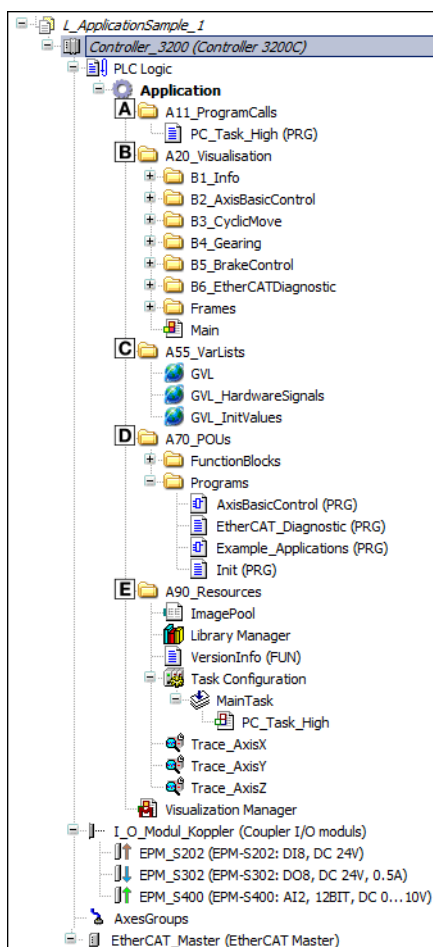
5 Overview - The structure of the Application Sample

The Application Sample i700 is a project template which provides support during commissioning and helps you entering the world of Lenze "Controller-based Automation" with the »PLC Designer«.

- The Application Sample is included in the »PLC Designer« setup as project template:

 **L_AppSample_i700.project**

- Lenze devices already contained in the project are:
 - I/O modules on the backplane bus of the controller (I/O system 1000),
 - Three axes (AxisX...Z) on the EtherCAT fieldbus (single axis of i700 servo inverter each).
- The Application Sample i700 has the following (fixed) structure:



A **A11_ProgramCalls**

- ... contains all program calls.

B **A20_Visualisation**

- ... contains the visualisations for the device-independent functions.

C **A55_VarLists**

- ... contains the declarations of the global variables:
 - Variables for switching over the visualisation: **GVL**
 - I/O variables: **GVL_Hardsignals**
 - Initialisation variables: **GVL_InitValues**

D **A70_POUs**

- ... contains the programs for:
 - BrakeControl/CyclicMove/EtherCAT/Gearing

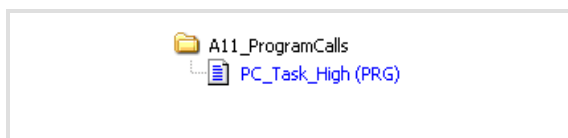
E **A90_Resources**

- ... contains all system data such as:
 - task settings,
 - used libraries,
 - predefined traces,
- Visualization manager.

Overview - The structure of the Application Sample

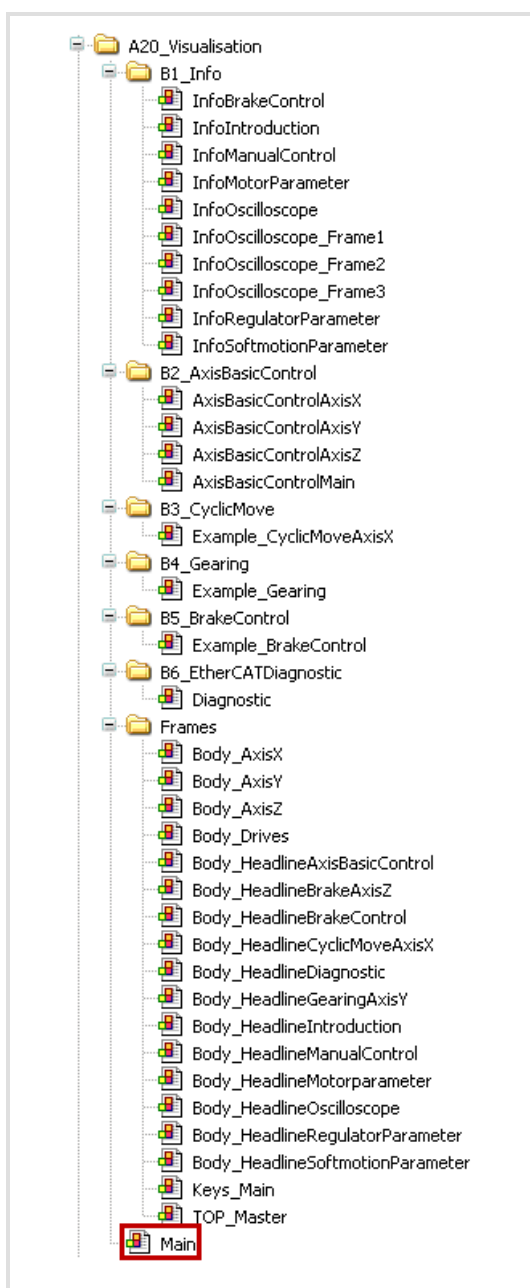
A11_ProgramCalls

5.1 A11_ProgramCalls



- The **PC_Task_High** program calls all high-priority programs of the application.

5.2 A20_Visualisation



B1_Info: "Getting started"

- ... contains the stored information on the structure and use of the Application Sample.

B2_AxisBasicControl

- ... contains the visualisations for the individual instances of the **L_SMC_AxisBasicControl** block.

B3_CyclicMove

- ... contains the visualisation of the **CyclicMove** block.

B4_Gearing

- ... contains the visualisation of the **Gearing** block.

B5_BrakeControl

- ... contains the visualisation of the **BrakeControl** block.

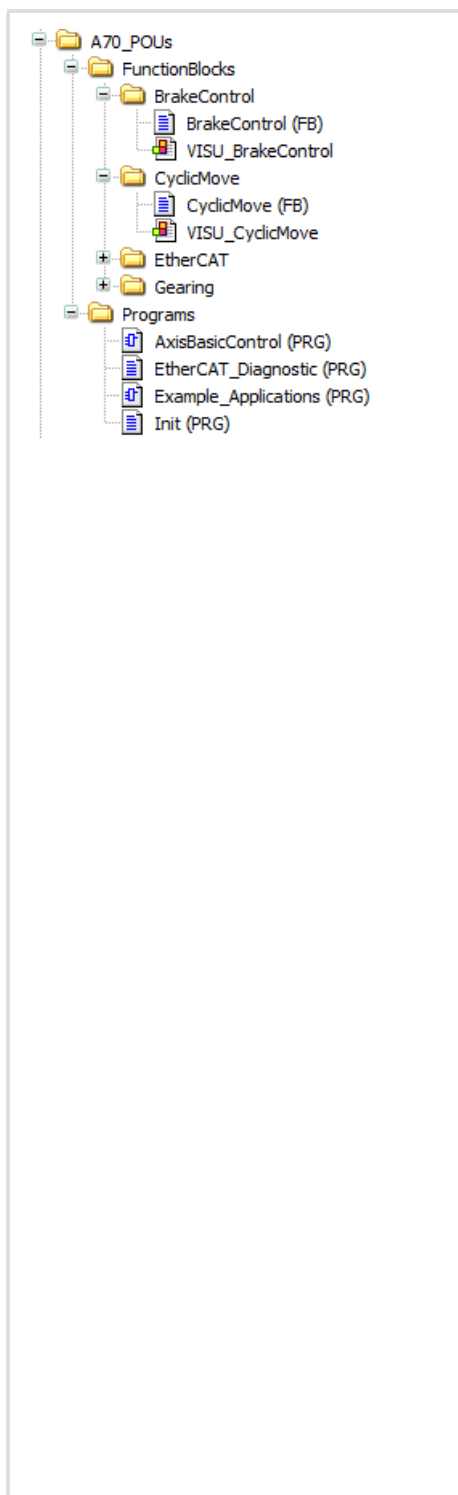
B6_EtherCATDiagnostic

- ... contains the visualisation of the EtherCAT diagnostic of the application.

Main calls the user interface of the main page of the visualisation which serves to run the entire visualisation.

- ... contains all visualisations of subfolders **B1-B6**.

5.3 A70_POUs



BrakeControl

- ... contains the instanceable sample program, **BrakeControl (FB)**, including the brake function, which can be used with the following function blocks:
 - **L_SM3_AxisBasicControl**
 - **SMCBrakeControl**
 - **SMCBrakeState**
- ... contains the corresponding visualisation.

CyclicMove

- ... contains the instanceable sample program, **CyclicMove (FB)**, for using the **L_SMC_AxisBasicControl** function block.
- ... contains the corresponding visualisation.

EtherCAT

- ... contains the following programs:
 - MasterHandling:** Reinitialise EtherCAT bus.
 - ReadStatus:** Read the status of the EtherCAT slaves.
 - SetState:** Set the state of the EtherCAT slaves.

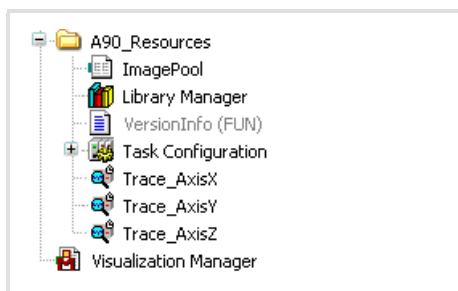
Gearing

- ... contains the instanceable sample program, **Gearing (FB)**, for synchronising the Y axis to the X axis via PLC open functions.

Programs

- ... contains the following programs:
 - **AxisBasicControl:**
Calls three instances of the **L_SMC_AxisBasicControl** function block.
 - **Example_Applications:**
Calls instances of the **Gearing**, **BrakeControl**, **CyclicMove** function blocks.
 - **EtherCATDiagnostic:**
Calls the **MasterHandling**, **ReadStatus**, **SetState** programs.
 - **Init:**
Writes initial values to those function block inputs which are not interconnected.

5.4 A90_Resources



BrakeControl

- ... contains the graphics referenced in the visualisations.

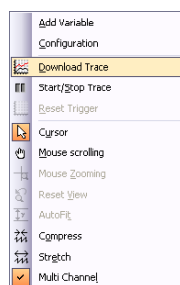
Trace_AxisX...Z

- ... pre-configured oscilloscope function of the respective axis.

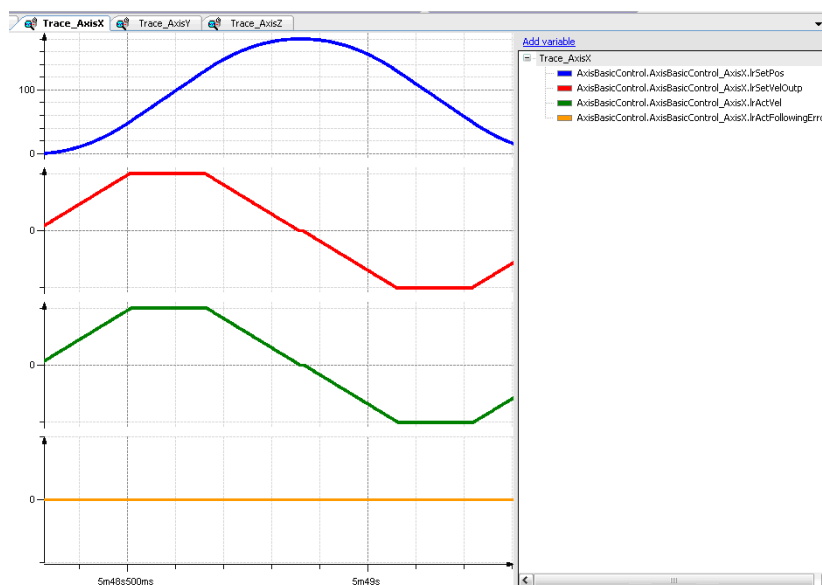
5.4.1 Use of the oscilloscope function: Example of AxisX

How to proceed:

1. Double-click the desired  **Trace** in the device view.
2. Select the **Download Trace** command in the context menu of the oscilloscope to load the data to the Controller.
 - Alternatively, execute the command via the menu **Trace→Download Trace**.



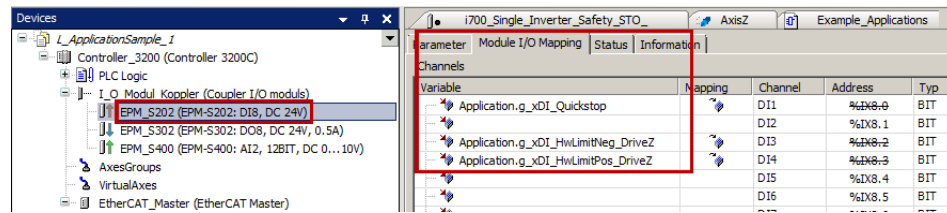
- The **Add Variable** option makes evaluation easier by providing all signals in separate channels.



Overview - The structure of the Application Sample

A90_Resources

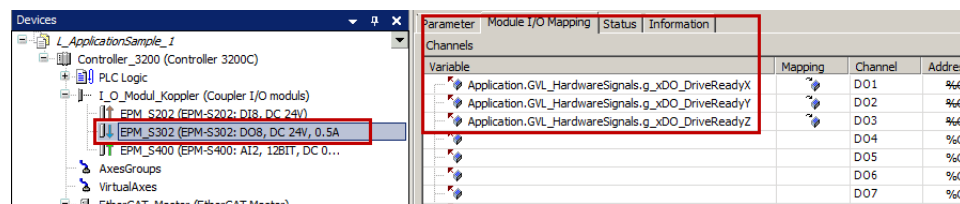
5.4.2 EPM_S202: Digital inputs



The hardware configuration of the sample project contains eight digital inputs. The corresponding Lenze Teachware case i700 contains a control panel for controlling the inputs of the I/O system.

- Digital input 1 "DI1": Mapped to the global variable for quick stop (QSP).
- Digital input 3 "DI3": Mapped to the global variable of the negative hardware limit switch for Axis Z.
- Digital input 4 "DI4": Mapped to the global variable of the positive hardware limit switch for Axis Z.

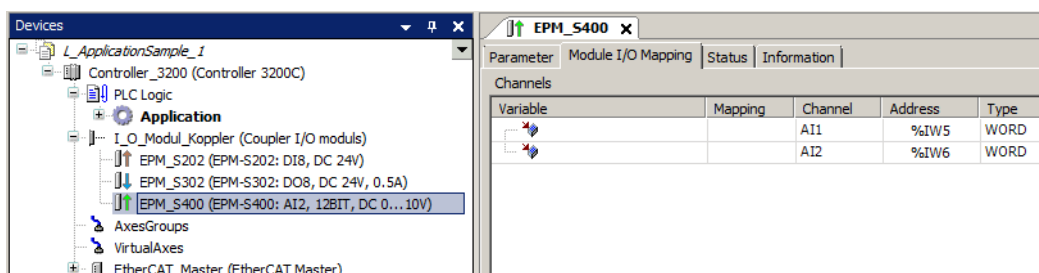
5.4.3 EPM_S303: Digital outputs



The hardware configuration of the sample project contains eight digital outputs. The corresponding Lenze Teachware case i700 contains a control panel for evaluating the outputs of the I/O system.

- Global variable *DriveReadyX* is mapped to digital output 1 "DO1".
- Global variable *DriveReadyY* is mapped to digital output 2 "DO2".
- Global variable *DriveReadyZ* is mapped to digital output 3 "DO3".

5.4.4 EPM_S400: Analog inputs



The hardware configuration of the sample project contains two analog inputs. The corresponding Lenze Teachware case i700 contains a control panel for controlling the inputs of the I/O system.

- The sample project does not contain/require a pre-configured mapping of the analog inputs.

6 Opening the Application Sample

The Application Sample includes a sample program with three axes and a predefined visualisation.

General procedure

The main steps are presented in the table below:

Step	Activity
1.	▶ Creating a new project - opening the Application Sample (📖 18)
2.	▶ Updating the Controller in the project (optional) (📖 18)
3.	▶ Going online (📖 19)
4.	▶ Downloading and starting the PLC program (📖 19)
5.	▶ Getting started - operating the Application Sample (📖 21)



Further information about the parameterisation and configuration of the is provided in the following communication manuals:

- "Controller-based Automation" EtherCAT - *commissioning & configuration*

The commissioning steps in detail

The following section provides a detailed description of every commissioning step.

Please follow the instructions below carefully to commission your automation system.

Opening the Application Sample

Creating a new project - opening the Application Sample

6.1 Creating a new project - opening the Application Sample

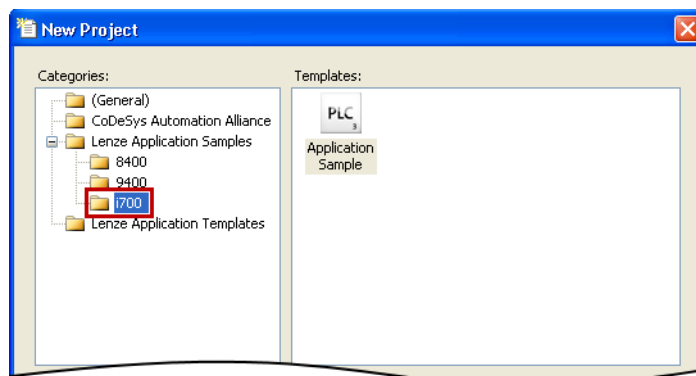
The Application Sample is available as a predefined »PLC Designer« project (*.project, ). To call the Application Sample, a new project must be created with the Application Sample being the template.



How to proceed:

1. Creating a new project:

- File →  **New project**



- Select category  *Lenze Application Samples\Lenze conventional\i700*
- Open template **Application sample**

6.2 Updating the Controller in the project (optional)

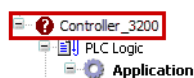
The Controller must optionally be updated in the »PLC Designer« if the project has older firmware information than the hardware to be used.

- If the Controller is marked with the symbol  after opening the project, the device must be updated.
- Use the »WebConfig« to determine the firmware used by the Controller to be able to select the corresponding setting in the »PLC Designer«.



How to proceed:

1. Select the desired controller:



- Execute the **Update device** command in the context menu.
- Double-click the desired controller in the **Update device** dialog box to update the controller in the device view.

Opening the Application Sample

Going online

6.3 Going online

To establish an online connection to the Controller, the communication settings (**Set Active Path**) must be adapted first. ▶ [Setting up communication to the Controller](#) (📖 8)

6.3.1 Transferring the project to the control system - "logging in"

The "Login" command transfers the PLC program to the controller.

- Transfer the project to the Controller by "logging in" to the control system:

Menu command **Online** →  **Login**

6.4 Downloading and starting the PLC program

- Select the **Online** → **Load** menu command to load the PLC program to the Controller.
- Select the **Online** → **Run** menu command to start the PLC program.
 - As an alternative, you can execute the **Debug** → **Start** menu command or press <F5>.



Tip!

If the project is to be loaded automatically after a restart of the device, you can install it as "Boot project".



How to install the project as boot project:

1. Select the **Online** → **Generate boot project for L-force Controller** menu command.

Opening the Application Sample

Simulation

6.5 Simulation



Note!

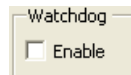
"Watchdog" monitoring must be deactivated for all tasks before the simulation mode can be used with the Application Sample.

- If the "Watchdog" monitoring is active, an error message will be issued and the simulation mode cannot be activated.



How to deactivate the "Watchdog" monitoring:

1. Double-click the *A90_Resources* folder in the device view.
 - Double-click **Task Configuration** → **MainTask** .
2. On the **Configuration** tab, deactivate the *Watchdog* by removing the ☒ checkmark:



7 Getting started - operating the Application Sample

The Application Sample i700 includes visualisations for the operation of the i700 which can be controlled via the  **Main** visualisation.

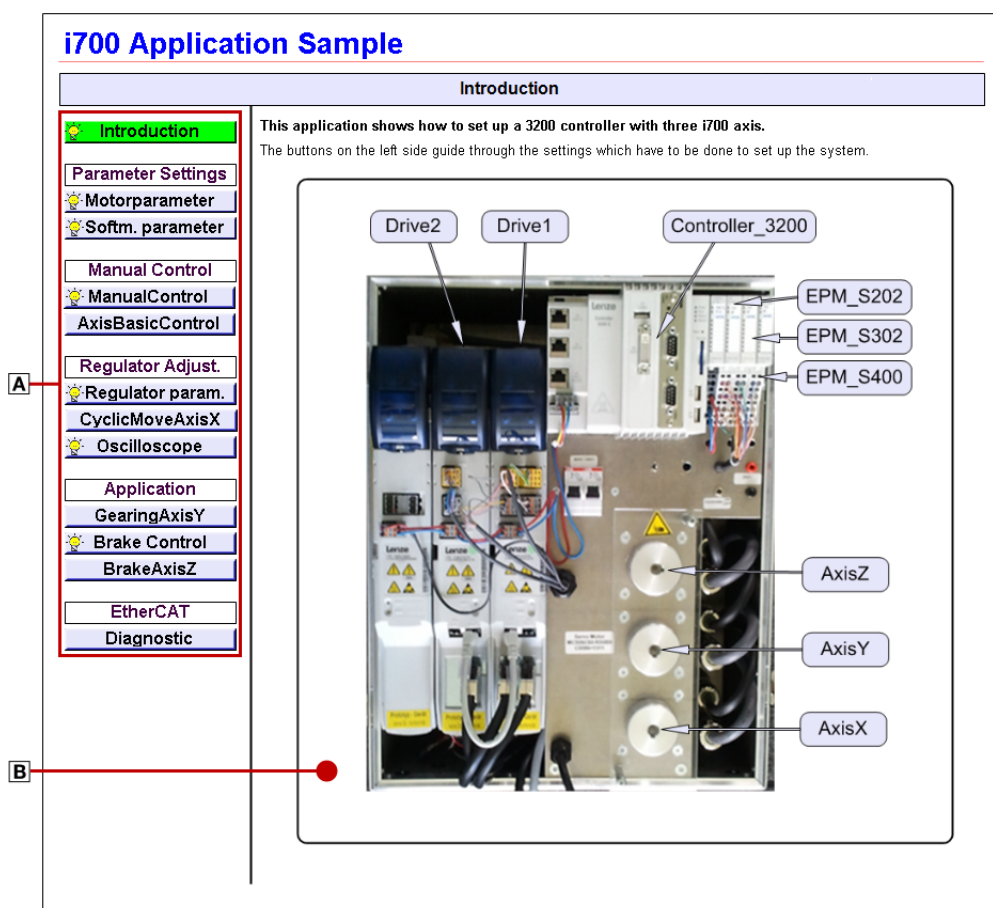


How to proceed:

1. In the *Device View*, select the  *A20_Visualisation* folder.
2. Double-click the  **Main** visualisation.

Welcome page - Main visualisation

The user interface of the visualisation is divided into the following areas:



A Buttons:

- Select the desired visualisation.

B Operating range of the machine modules

- Displays the visualisation page selected.

Getting started - operating the Application Sample

The buttons of the visualisation

7.1 The buttons of the visualisation



Tip!

The information pages of the visualisation are marked with the symbol .

- The pages provide information on what to do to get the system started: "Getting started".

Parameter Settings
 Motorparameter
 Softm. parameter
Manual Control
 ManualControl
AxisBasicControl
Regulator Adjust.
 Regulator param.
CyclicMoveAxisX
 Oscilloscope
Application
GearingAxisY
 Brake Control
BrakeAxisZ
EtherCAT
Diagnostic

Setting parameters

Motor parameters: information page "Getting started"

- **Step 1:**
How to parameterise the motor data

Softm. parameters: information page "Getting started"

- **Step 2:**
How to parameterise the SoftMotion data

Manual control

ManualControl: information page "Getting started"

- **Step 3:**
How to move the axes (*AxisX/AxisY/AxisZ*) in the manual mode (manual jog)

AxisBasicControl: user interface

- Axis control via the visualisation of the **L_SMC_AxisBasicControl** FB.

Controller settings

Regulator param.: information page "Getting started"

- **Step 4:**
How to change the controller settings

CyclicMoveAxisX: user interface

- Visualisation of the program **CyclicMoveAxisX** ▶ [Axis control in the automatic mode: "CyclicMove"](#)

Application

Oscilloscope: information page "Getting started"

- **Step 5:**
How to optimise the controller settings

GearingAxisY:

- Visualisation of the **GearingAxisY** program

BrakeControl: information page "Getting started"

- **Step 6:**
How to use the holding brake

BrakeAxisZ: user interface

- Visualisation of the **BrakeAxisZ** program

EtherCAT

Diagnostic: EtherCAT diagnostic page

- Query/set EtherCAT state (bus/slave)

Getting started - operating the Application Sample

Commissioning an axis

7.2 Commissioning an axis



Note!

If the Lenze Teachware case i700 is used, steps **1-4** are not required!

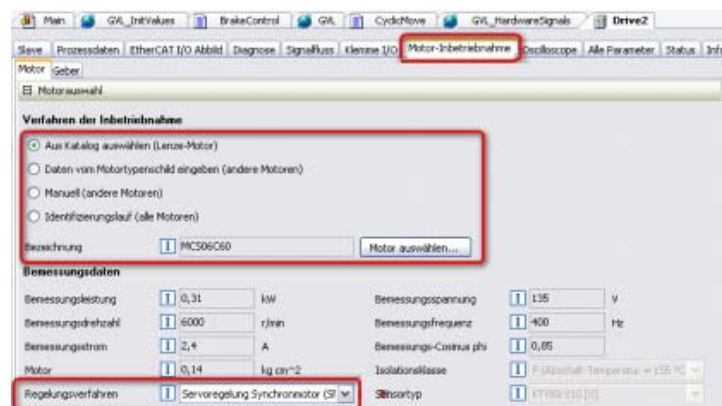
- The parameters are predefined and already included in the project.



How to proceed:

1. Motor data parameterisation
2. SoftMotion data parameterisation
3. Axis control during manual operation
4. Controller adjustment by means of oscilloscope function

7.3 Motor data parameterisation



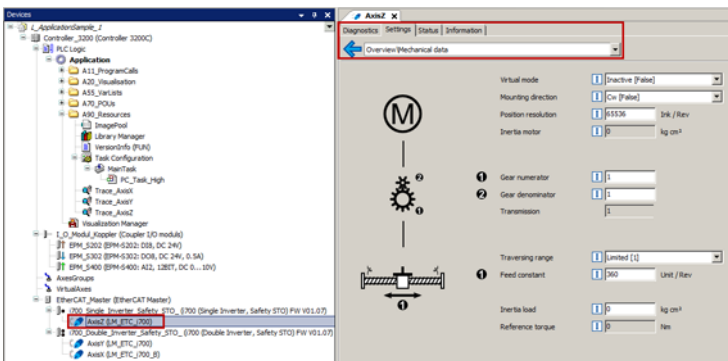
How to proceed:

1. Select the desired controller from the device tree.
2. Select the **Motor Commissioning** tab.
3. Select the desired motor.
4. Select the desired control mode.

Getting started - operating the Application Sample

Parameterising the axis data

7.4 Parameterising the axis data



The screenshot shows the 'AxisZ' configuration window with the 'Overview/Mechanical data' tab selected. The left pane displays a device tree with 'AxisZ' highlighted. The right pane shows various mechanical parameters for the axis, including virtual mode, mounting direction, position resolution, inertia motor, gear ratio, transmission, traversing range, feed constant, inertia load, and reference torque.

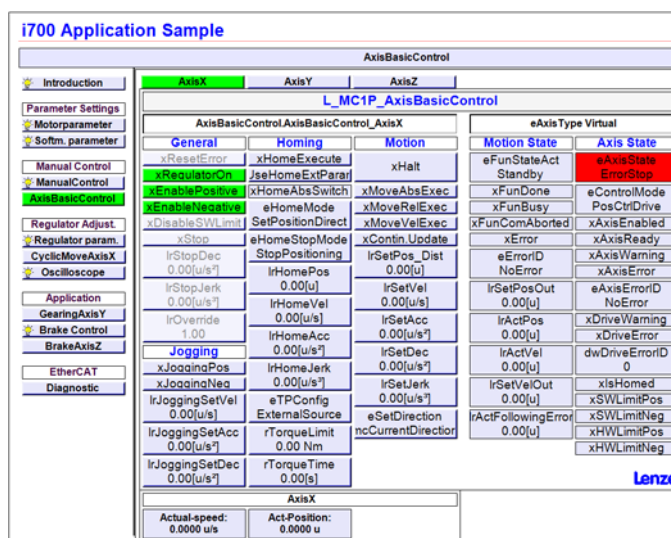
 **How to proceed:**

1. Select the desired axis in the device tree. Example: **AxisZ**
2. Select the **Overview\Mechanical data** tab. Select desired settings such as the traversing range of the axis.
3. Select the **Overview\Basic settings** tab. Select desired settings such as...
... the response of the following error monitoring.
....the suitable scaling of the units/mounting direction.

Getting started - operating the Application Sample

Axis control during manual operation: manual control

7.5 Axis control during manual operation: manual control



How to proceed:

1. Click the **AxisBasicControl** button.
2. Select the desired axis: **AxisX/AxisY/AxisZ**.
3. Click the **Internal Control** button.
4. Acknowledge pending errors by clicking the **xResetError** button.
5. Click the **xRegulatorOn** button.
 - Now the controller has been enabled and can be controlled via the buttons.

Example: Click the **xJoggingPos** button to control the axis in positive direction of rotation.

Getting started - operating the Application Sample

Axis control in the automatic mode: "CyclicMove"

7.6 Axis control in the automatic mode: "CyclicMove"

Cyclic Move AxisX

AxisX

Actual-speed: -27.1000 u/s Act-Position: 176.3008 u

CyclicMove

Example_Applications.CyclicMoveAxisX

xRegulatorOn IrActPos: 176.36[u]

xStart IrActVel: -26.90[u/s]

IrSetPos_Dist: 180.00[u]

IrSetVel: 100.00[u/s]

IrSetAcc: 100.00[u/s²]

IrSetDec: 100.00[u/s²]

IrWaitTime: 1.000[ms]

How to proceed:

1. Click the **CyclicMoveAxisX** button.
2. Enable the controller: Click the **xRegulatorOn** button.
3. Start the positioning process: Click the **xStart** button.
 - The positioning parameters can be edited anytime directly in the visualisation.

Getting started - operating the Application Sample

Coupling/decoupling of an axis: "GearingAxisY"

7.7 Coupling/decoupling of an axis: "GearingAxisY"

i700 Application Sample

Gearing AxisY

Introduction

Parameter Settings

Motorparameter

Softm. parameter

Manual Control

ManualControl

AxisBasicControl

Regulator Adjust.

Regulator param.

CyclicMoveAxisX

Oscilloscope

Application

GearingAxisY

Brake Control

BrakeAxisZ

EtherCAT

Diagnostic

AxisX

Actual-speed: 0.0000 u/s
Act-position: 0.0000 u

CyclicMove

Example_Applications.CyclicMoveAxisX

xRegulatorOn

xStart

IrSetPos_Dist
0.00[u]

IrActPos
0.00[u]

IrSetVel
0.00[u/s]

IrActVel
0.00[u/s]

IrSetAcc
0.00[u/s²]

IrSetDec
0.00[u/s²]

IrWaitTime
0.000[ms]

AxisY

Actual-speed: 0.0000 u/s
Act-position: 0.0000 u

MC_GearIn

Example_Applications.Gearing1.MC_GearIn1

Execute

Contin.Update

RatioNumerator
1

RatioDenominator
1

Master/ValueSource
mcSetValue

Acceleration
3600.00[u/s²]

Deceleration
3600.00[u/s²]

Jerk
0.00[u/s³]

BufferMode
mcAborting

MC_GearOut

Example_Applications.Gearing1.MC_GearOut1

Execute

Done

Busy

Active

CommandAborted

Error

ErrorID
NoError

MC_Halt

Example_Applications.Gearing1.MC_Halt1

Execute

Deceleration
0.00[u/s²]

Jerk
0.00[u/s³]

BufferMode
mcAborting

Done

Busy

Active

CommandAborted

Error

ErrorID
NoError

Lenze

How to proceed:

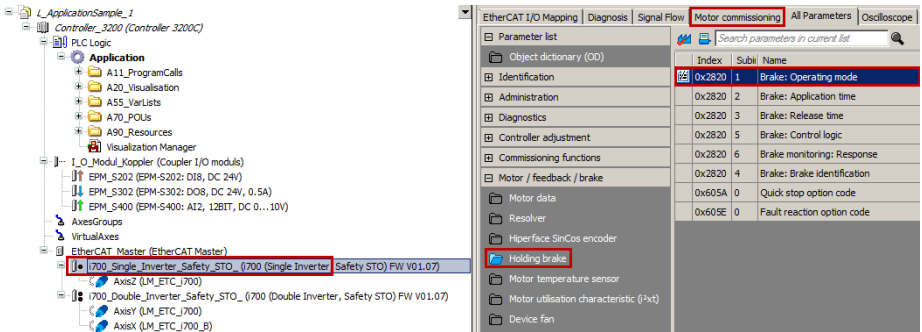
1. Click the **GearingAxisY** button.
2. Enable the controller: Click the **xRegulatorOn** button.
3. Start the positioning process: Click the **xStart** button.
4. Set the acceleration/deceleration: Enter the desired values in **MC_GearIn**.
5. Set the desired deceleration of **MC_Halt**.
6. Coupling/decoupling
 - Start "Coupling": Click **Execute** in **MC_GearIn**.
 - "Decoupling": Click **Execute** in **MC_GearOut**.
7. To bring **Axis Y** to a standstill after decoupling, click **Execute** in **MC_Halt**.

The parameters of the function blocks can be edited anytime directly in the visualisation.

Getting started - operating the Application Sample

Holding brake: Application "BrakeAxisZ"

7.8 Holding brake: Application "BrakeAxisZ"



Index	Subindex	Name
0x2820	1	Brake: Operating mode
0x2820	2	Brake: Application time
0x2820	3	Brake: Release time
0x2820	5	Brake: Control logic
0x2820	6	Brake monitoring: Response
0x2820	4	Brake: Brake identification
0x605A	0	Quick stop option code
0x605E	0	Fault reaction option code

For **Axis Z** a holding brake is defined by default, which releases at controller enable.

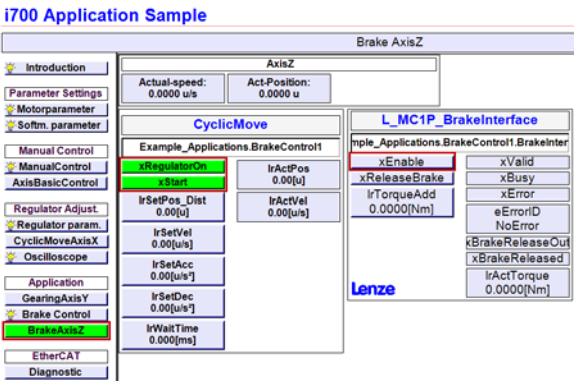
- The holding brake can be additionally switched over to other states (for example for purposes regarding the application) via FB **L_MC1P_BrakeInterface**.

 **Preparation:** First of all, the operating mode of the brake (index **0x2820/1**) must be set to a value of "1", the standard value is "0".

Note: When you quit the example, you must reset the brake setting to the initial value!

Movement of Axis X during automatic operation

After changing the operating mode of the brake, the function of the holding brake can be controlled via the **BrakeAxisZ** visualisation.



How to proceed:

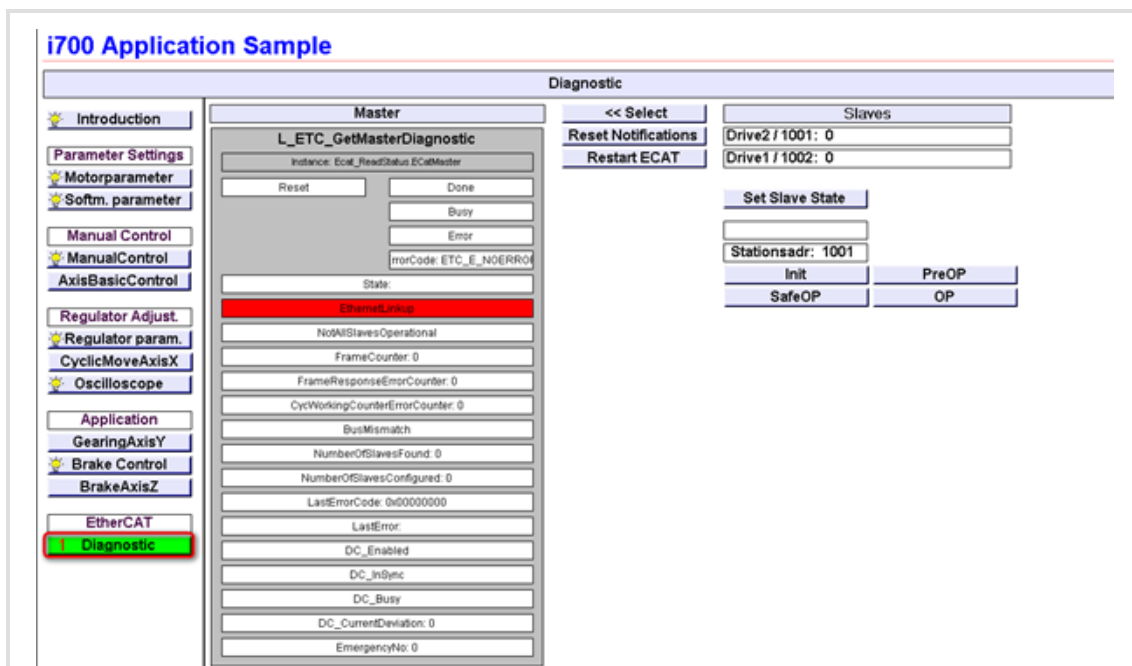
1. Click the **BrakeAxisZ** button.
2. Select brake mode for **L_MC1P_BrakeControl** :
 - Open brake: **xReleaseBrake** = **TRUE**
3. Release brake: click **xEnable** button of **L_MC1P_BrakeInterface**.
4. **xBrakeReleased** shows the current brake status.
5. Enable the controller: Click the **xRegulatorOn** button.
6. Start the positioning process: Click the **xStart** button.

The positioning parameters can be edited anytime directly in the visualisation.

Getting started - operating the Application Sample

EtherCAT diagnostics: "Diagnostic"

7.9 EtherCAT diagnostics: "Diagnostic"



The **Diagnostic** visualisation displays information on the EtherCAT state:

- State EtherCAT bus
... displays the current state of the bus. It shows e.g. whether the bus has been restarted after the communication to the slaves was interrupted (disconnected cable connection between Controller and node).
- EtherCAT slave status
... displays the state of the corresponding slave.
- **Set Slave State** serves to set the desired state for the corresponding slave.

Index

A

Application notes (representation) [6](#)

Application Sample i700 [10](#)

C

Conventions used [5](#)

CyclicMove [26](#)

D

Document history [5](#)

Downloading and starting the PLC program [19](#)

E

Elements of the Application Sample [11](#)

E-mail to Lenze [31](#)

Establishing communication with the controller [8](#)

F

Feedback to Lenze [31](#)

G

GearingAxisY [27](#)

Getting started - operating the Application Sample [21](#)

I

i700

SoftMotion parameters [24](#)

L

Logging in [19](#)

M

Manual control [25](#)

MM_DcI [12](#)

MM_IO [12](#)

MM_Par [12](#)

MM_PD [12](#)

MM_Vis [12](#)

N

Notes used [6](#)

O

Objective of the Application Sample [10](#)

S

Safety [7](#)

Safety instructions (representation) [6](#)

Screenshots [4](#)

Simulation [20](#)

Starting the Application Sample i700 [17](#)

T

Target group [4](#)

Transferring the project to the control system (logging in) [19](#)

V

Validity of the documentation [4](#)



Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Automation GmbH

Hans-Lenze-Str. 1

D-31855 Aersen

Germany

☎ +49 (0)51 54 / 82-0

📠 +49 (0)51 54 / 82-28 00

✉ Lenze@Lenze.de

🌐 www.Lenze.com

Service

Lenze Service GmbH

Breslauer Straße 3

D-32699 Extertal

Germany

☎ 00 80 00 / 24 4 68 77 (24 h helpline)

📠 +49 (0)51 54 / 82-11 12

✉ Service@Lenze.de