

ILx2K EtherNet/IP

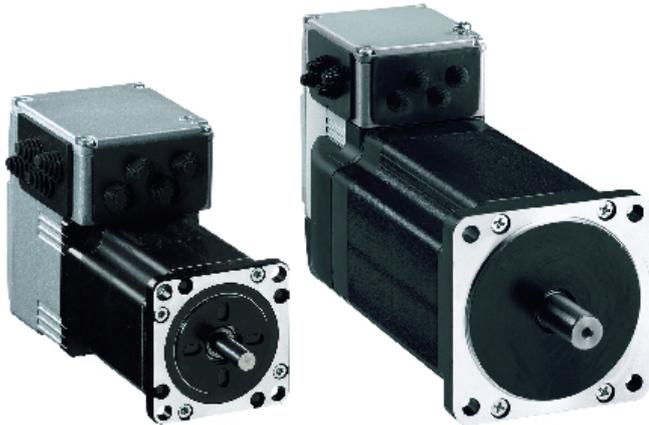
Lexium Integrated Drive

Fieldbus manual PLCopen

Add-On Instructions for

Rockwell RS Logix5000

V0.08, 11.2009



BERGER LAHR

www.schneider-electric.com

Schneider
Electric

Important information

This manual is a part of the PLCopen Add-On Instructions for Rockwell RSLogix 5000.

Carefully read this manual and observe all instructions.

Keep this manual for future reference.

Some products are not available in all countries.

Please consult the latest catalog for information on the availability of products.

Subject to technical modifications without notice.

All details provided are technical data which do not constitute warranted qualities.

Most of the product designations are registered trademarks of their respective owners, even if this is not explicitly indicated.

Table of contents

1	Introduction	4
1.1	Scope of supply	4
2	Creating an application	5
2.1	Export / import of the Add-On Instructions and data types	5
2.2	Creating the global data structure per drive	7
2.3	I/O configuration	7
2.4	Loading and testing the I/O configuration	10
2.5	Addressing the drives	12
2.5.1	Rotary switches for setting the IP address	12
2.5.2	Setting the IP address via the integrated Web server	12
2.5.3	Setting the IP address via Lexium CT	14
3	Add-On Instructions	17
3.1	Function block names	17
3.2	Signal diagrams for inputs and outputs	17
3.2.1	Control inputs	17
3.2.2	Control outputs	18
3.2.3	Signal diagrams for function blocks with Execute control input	19
3.2.4	Signal diagram for function blocks with control input Enable	20
3.3	Description of the function blocks	21
3.4	Usage of the function blocks	23
3.4.1	Initialization	23
3.4.2	Jog	23
3.4.3	Homing	23
3.4.4	Operating mode Profile Position	24
3.4.5	Operating mode Profile velocity	24
3.4.6	Stopping	24
3.4.7	Fast position capture	25
3.4.8	Reading parameters	25
3.4.9	Writing parameters	25
3.4.10	Inputs/outputs	26
3.4.11	Error handling	26
3.5	Detailed description of function blocks	27
3.5.1	MC_UpdateInputData_ETH_ILx2	27
3.5.2	MC_UpdateOutputData_ETH_ILx2	28
3.5.3	MC_Power_ETH_ILx2	29
3.5.4	MC_Jog_ETH_ILx2	31
3.5.5	MC_Home_ETH_ILx2	33
3.5.6	MC_SetPosition_ETH_ILx2	36
3.5.7	MC_MoveAbsolute_ETH_ILx2	37
3.5.8	MC_MoveRelative_ETH_ILx2	40
3.5.9	MC_MoveAdditive_ETH_ILx2	42
3.5.10	MC_MoveVelocity_ETH_ILx2	44
3.5.11	MC_VelocityControl_ETH_ILx2	46
3.5.12	MC_Stop_ETH_ILx2	48
3.5.13	MC_TouchProbe_ETH_ILx2	50

PLCopen AddOnInstructions ILx2 EthernetIP

3.5.14	MC_AbortTrigger_ETH_ILx2.....	52
3.5.15	MC_ReadActualPosition_ETH_ILx2	54
3.5.16	MC_ReadActualVelocity_ETH_ILx2	55
3.5.17	MC_ReadStatus_ETH_ILx2.....	56
3.5.18	MC_ReadParameter_ETH_ILx2	59
3.5.19	MC_WriteParameter_ETH_ILx2	61
3.5.20	MC_SetLimitSwitch_ETH_ILx2	63
3.5.21	MC_ReadDigitalInputs_ETH_ILx2	65
3.5.22	MC_Reset_ETH_ILx2	66
3.5.23	MC_ReadAxisError_ETH_ILx2	68
4	Communication monitoring	70
4.1	Reading the I/O LED state	70
4.2	Monitoring individual I/O connections	71
5	Appendix	72
5.1	Error numbers	72

Information on this edition

Version 0.0/1

Manual created

Version08

Input assembly changed from 44 bytes to 40 bytes. This change is done in the AddOnInstruction Version 0.201 and higher.

1 Introduction

Programs according to the IEC 61131-3 standard are created with the function blocks as per PLCopen Motion specification in the RSLogix 5000 programming environment provided by Rockwell. The function blocks control one or several Schneider Electric Lexium ILx2 drives via the EtherNet/IP fieldbus. The function blocks are universal so that the library is suitable for the following drives:

- ILS2K...
- ILE2K...
- ILA2K...

1.1 Scope of supply

Check the delivery for completeness to enable easy and complete integration into the Rockwell RLogix 5000 programming system.

The version number is a part of the library name; Vxxx stands for the corresponding version number, for example V1001.

- RSLogix project with all available Add-On Instructions and a sample program

SE_Motion_ILx2_ETH_Vxxx.ACD

- Device description SEILx2K_0011E.EDS.eds
- This manual

2 Creating an application

The Add-On Instructions have been created for the RSLogix 5000 programming software. The following steps are required to make the Lexium ILx2 drives known to the RSLogix 5000 software and to link the process data of the drives and the data structures of the library. Perform the steps in the sequence listed.

The project "SE_Motion_ILx2_ETH_Vxxxx.ACD" can be used as a basis for a new application.

If the Add-On Instructions are to be integrated into an existing RSLogix 5000 project, they may be exported from the project "SE_Motion_ILx2_ETH_Vxxxx.ACD" and imported into the existing project.

2.1 Export / import of the Add-On Instructions and data types

The RSLogix 5000 software does not allow you to export or import several Add-On Instructions at the same time. Therefore, each function block and each data type must be exported and imported separately.

- Open the project "SE_Motion_ILx2_ETH_Vxxxx.ACD" with the RSLogix 5000 software.

- Export the 3 data types

AXIS_REF_EIP_ILx2



Bild 1) Required data types

- Right-click the data type

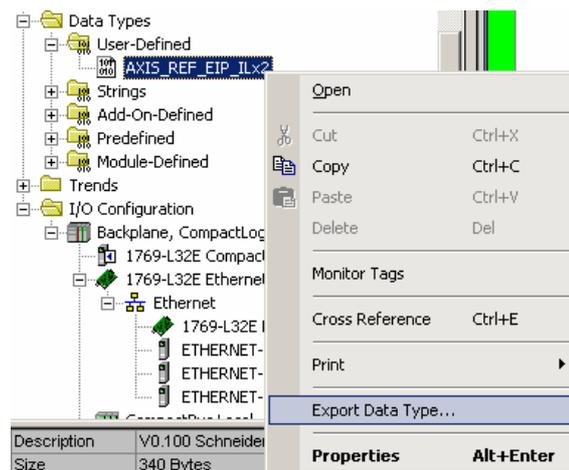


Bild 2) Menu item "Export Data Type..." from the context menu

- Save the *.L5X files to a known folder.
- Export the Add-On Instructions.

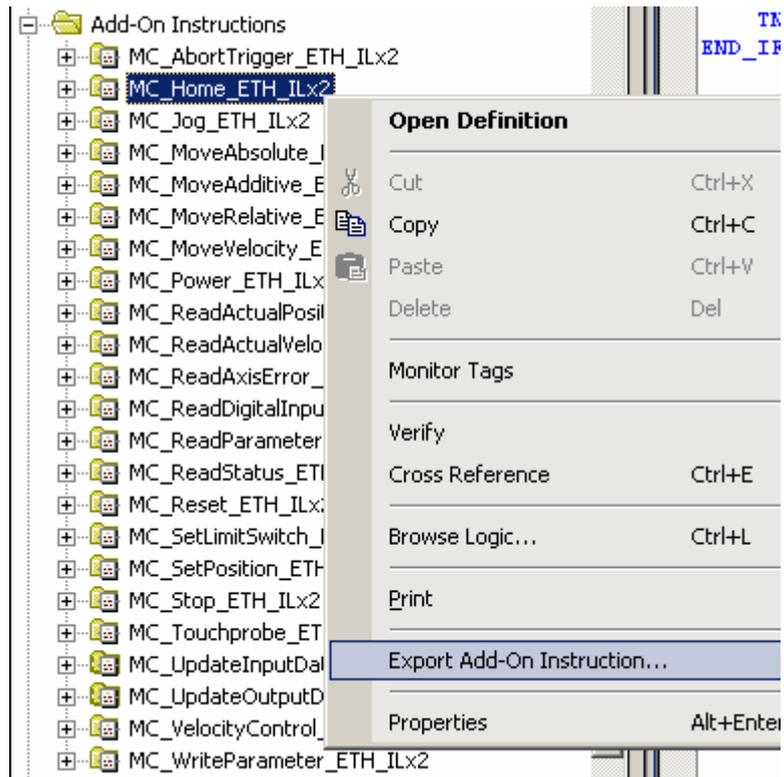


Bild 3) Menu item "Export Add-On-Instruction..." from the context menu

- Save the *.L5X files to a known folder.
 To save on disk space, you do not have to export all available Add-On Instructions.
 The required functions and operating modes determine the Add-On Instructions required for the application.
 For example, if you do not need fast position capture via the Capture input, the Add-On Instructions "MC_AbortTrigger_ETH_ILx2" and „MC_Touchprobe_ETH_ILx2“ do not need to be exported/imported.
- Now, first import the data types and then the Add-On Instructions from the *.L5X files.

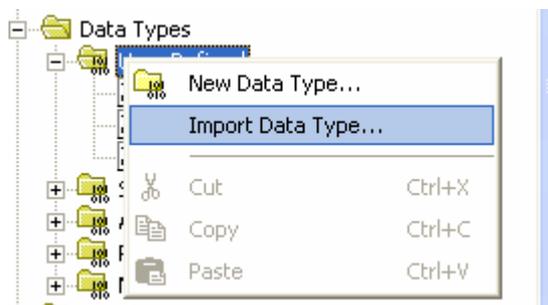


Bild 4) Menu item "Import Data Type..." from the context menu



Bild 5) Menu item "Import Add-On Instruction..." from the context menu

2.2 Creating the global data structure per drive

- Now create a data structure in the "Controller Tags" or the "Program Tags" for each Lexium ILx2 drive you want to operate via EtherNet/IP.

The data structure must be of the type `AXIS_REF_EIP_ILx2` sein.

Name	Alias For	Base Tag	Data Type	Style	Description
+AxisRef_1			AXIS_REF_EIP_ILx2		V0.100 Schneider Electric Motion
+AxisRef_2			AXIS_REF_EIP_ILx2		V0.100 Schneider Electric Motion
+AxisRef_3			AXIS_REF_EIP_ILx2		V0.100 Schneider Electric Motion

Bild 6) Data structure for each drive (example: 3 drives)

2.3 I/O configuration

The input and output devices managed in the I/O configuration. The Lexium ILx2 drives with EtherNet/IP interface are added here.

- Add the Lexium ILx2 drives to the I/O configuration. To do so, select the EtherNet/IP master, right-click it and choose "New Module..." from the context menu.

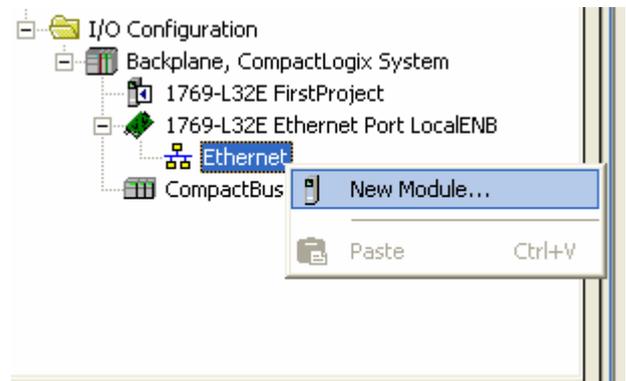


Bild 7) New Module...

- In the window, open the "Communications" node and select the entry "ETHERNET-MODULE – Generic Ethernet Module". Confirm by clicking "OK"

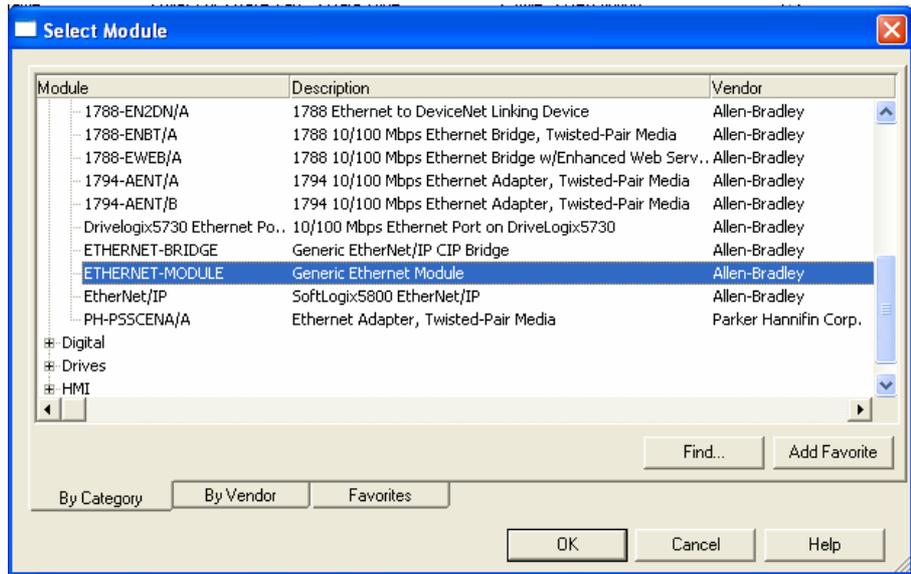


Bild 8) Adding an EtherNet/IP drive

- After you have confirmed with OK, the software opens the window "Module Properties". The name of the derive can be chosen as required. Enter the following Connection Parameters:

Input Assembly Instance = 113 Size 40 (8-Bit)
Output Assembly Instance = 103 Size 40 (8-Bit)
Configuration Assembly Instance = 3 Size 0 (8-Bit)

Comm_Format = Data-SINT

Enter the IP address or the host name of the drive.
Please refer to the product manual for information on addressing the drive.

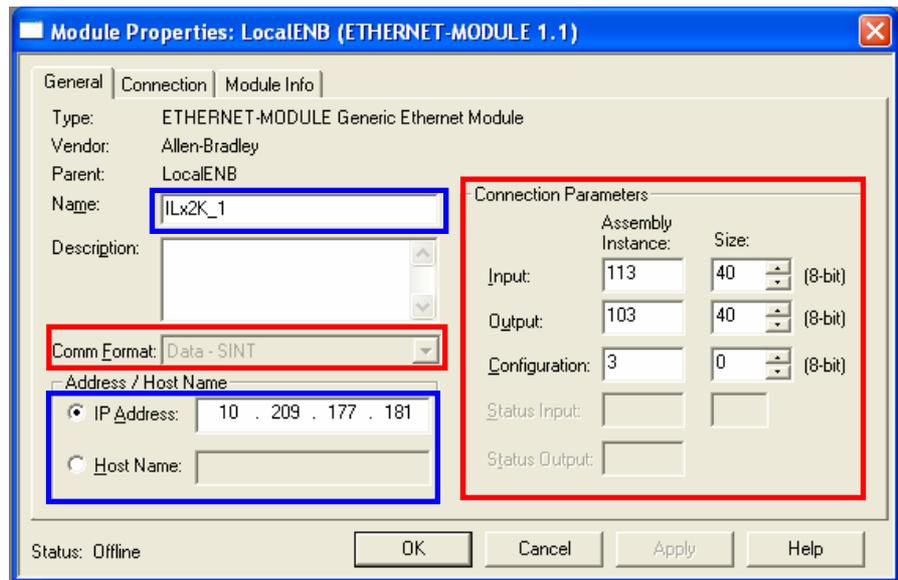
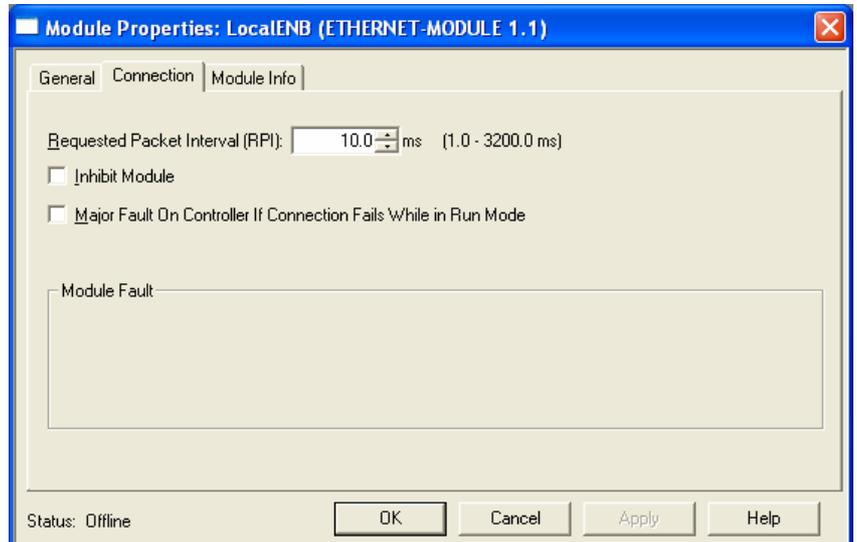


Bild 9) EtherNet/IP Module Properties / General

PLCopen AddOnInstructions ILx2 EthernetIP

- Now display the "Connection" tab and specify the desired "Requested Packet Interval".
The default is 10ms. Please note that this value must not be too small, depending on the number of EtherNet/IP devices.



- Repeat the previous steps until you have added all drives to the EtherNet/IP network.
- The Controller Tags now displays the process data of the drive.

The screenshot shows a window titled "Controller Tags - Compact(controller)". It has a "Scope:" dropdown menu set to "Compact" and buttons for "Show..." and "Show All". Below this is a table with the following data:

Name	Alias For	Base Tag	Data Type
+ ILx2K_0:C			AB:ETHERNET_MODULE:C:0
+ ILx2K_0:I			AB:ETHERNET_MODULE_SINT_40Bytes:I:0
+ ILx2K_0:O			AB:ETHERNET_MODULE_SINT_40Bytes:O:0

Bild 10) Drive with its process data

2.4 Loading and testing the I/O configuration

@ DANGER

Unintended consequences of equipment operation

When the system is started, the drives are usually out of the operator's view and cannot be visually monitored.

- Only start the system if there are no persons or obstructions in the hazardous area.

Failure to follow these instructions will result in death or serious injury.

@ WARNING

Unexpected movement

Drives may perform unexpected movements because of incorrect wiring, incorrect settings, incorrect data or other errors.

Interference (EMC) may cause unpredictable responses in the system.

- Carefully install the wiring in accordance with the EMC requirements.
- Switch off the voltage at the inputs `STO_A` (`PWRR_A`) and `STO_B` (`PWRR_B`) to avoid an unexpected start of the motor before switching on and configuring the drive system.
- Do NOT operate the drive system with unknown settings or data.
- Perform a comprehensive commissioning test.

Failure to follow these instructions can result in death or serious injury.

PLCopen AddOnInstructions ILx2 EthernetIP

The following steps transmit the project to the controller and start it.

- Load the project to the controller and start it with "RUN".
- The status LED I/O indicates whether or not the I/O configuration is running properly. -> If this is the case, the LED is "steady on".



Bild 11) Status LEDs of the Rockwell controller (I/O OK)

- If one or several assemblies / fieldbus devices are not available or if there is an error, the LED flashes.

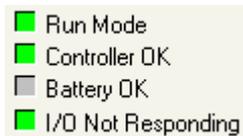


Bild 12) Status LEDs of the Rockwell Controller (I/O Not Responding)

- The communication status of an individual drive can be monitored in online mode in the RSLogix 5000 software.

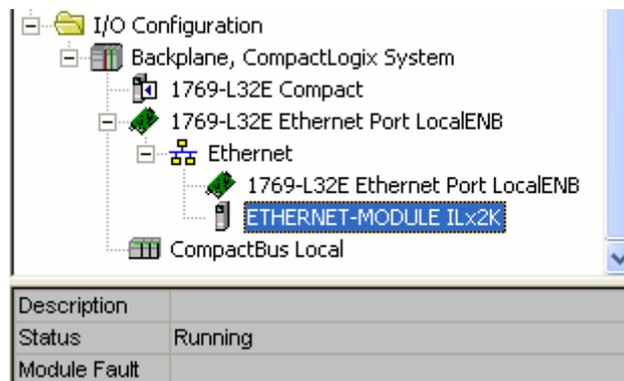


Bild 13) ILx2 communication status - Running

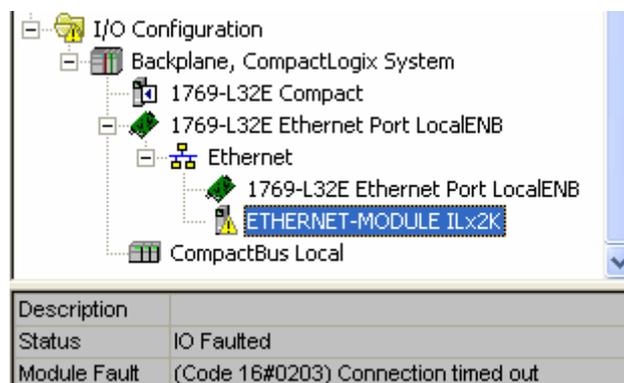


Bild 14) ILx2 communication status – IO error

2.5 Addressing the drives

Drives can be addressed in various ways.
Please refer to the product manual, section "Commissioning Steps" for detailed descriptions of setting the address.
The following section provides a short description of the various options.

2.5.1 Rotary switches for setting the IP address

The drive has 2 rotary switches for setting the IP address.
If the rotary switches are set to "Stored IP", you can set and modify the IP address via the integrated web server or the "Lexium CT" commissioning software.
The drive is shipped with the default IP 192.168.100.10 .
For more information see the product manual.

2.5.2 Setting the IP address via the integrated Web server

The IP address can only be changed if the rotary switches are set to "Stored IP".

If a different addressing mode is selected, changes to the communication parameters are reset the next time the device boots.

It is also possible to start the integrated web server if an Ethernet/IP master already communicates with the drive.

- Use a web browser to connect to the drive.
In the address field of your Internet browser, enter the current IP address of the drive.

http:// 192.168.100.10 (default)

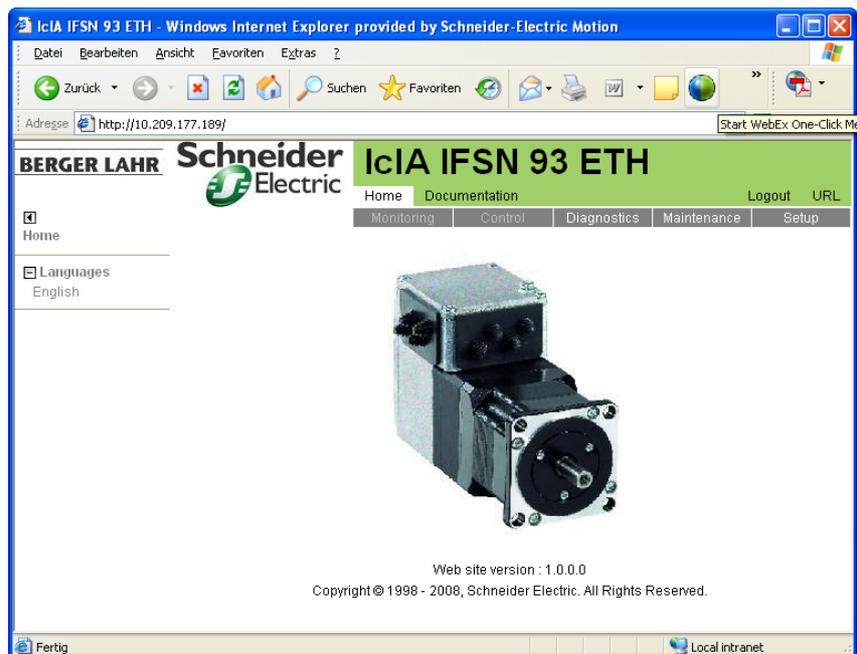


Bild 15) Start page ILx2 – Web-Server

PLCopen AddOnInstructions ILx2 EthernetIP

- Menu: Maintenance -> Communication -> Ethernet
- User name: USER
Password: USER



Verbindung zu 10.209.177.189 herstellen

cgi/printParameter.cgi

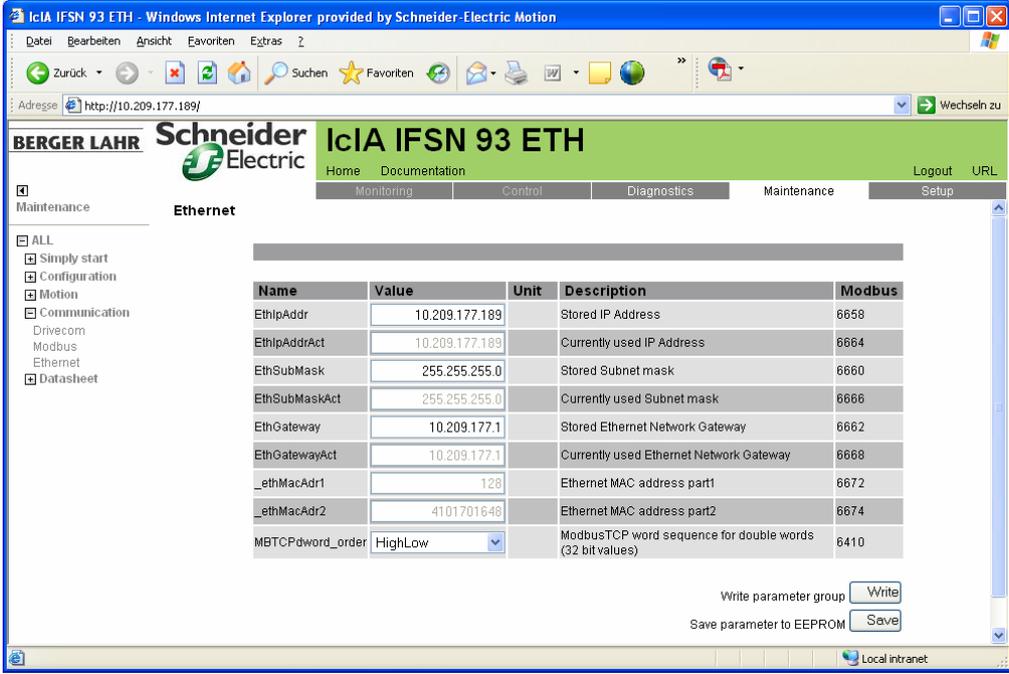
Benutzername: USER

Kennwort:

Kennwort speichern

OK Abbrechen

Bild 16) Logging in



ICA IFSN 93 ETH - Windows Internet Explorer provided by Schneider-Electric Motion

Adresse: http://10.209.177.189/

BERGER LAHR **Schneider Electric** **ICA IFSN 93 ETH**

Home Documentation Logout URL

Monitoring Control Diagnostics Maintenance Setup

Ethernet

Name	Value	Unit	Description	Modbus
EthIpAddr	10.209.177.189		Stored IP Address	6658
EthIpAddrAct	10.209.177.189		Currently used IP Address	6664
EthSubMask	255.255.255.0		Stored Subnet mask	6660
EthSubMaskAct	255.255.255.0		Currently used Subnet mask	6666
EthGateway	10.209.177.1		Stored Ethernet Network Gateway	6662
EthGatewayAct	10.209.177.1		Currently used Ethernet Network Gateway	6668
_ethMacAdr1	128		Ethernet MAC address part1	6672
_ethMacAdr2	4101701648		Ethernet MAC address part2	6674
MBTCPdword_order	HighLow		ModbusTCP word sequence for double words (32 bit values)	6410

Write parameter group Write

Save parameter to EEPROM Save

Bild 17) Communication parameters

- The Ethernet parameters can be changed on this page.
- Click the button "Write" to transmit the changed parameters to the drive.
- Click the button "Save" to save the parameters to the EEPROM.
- The changed communication parameters do not become active until you switch on the drive the next time!

2.5.3 Setting the IP address via Lexium CT

The current version of the Lexium CT commissioning software can be downloaded from the Internet at :

www.schneider-electric-motion.com

A complete description of the software can be found in the "Lexium CT Product Manual".

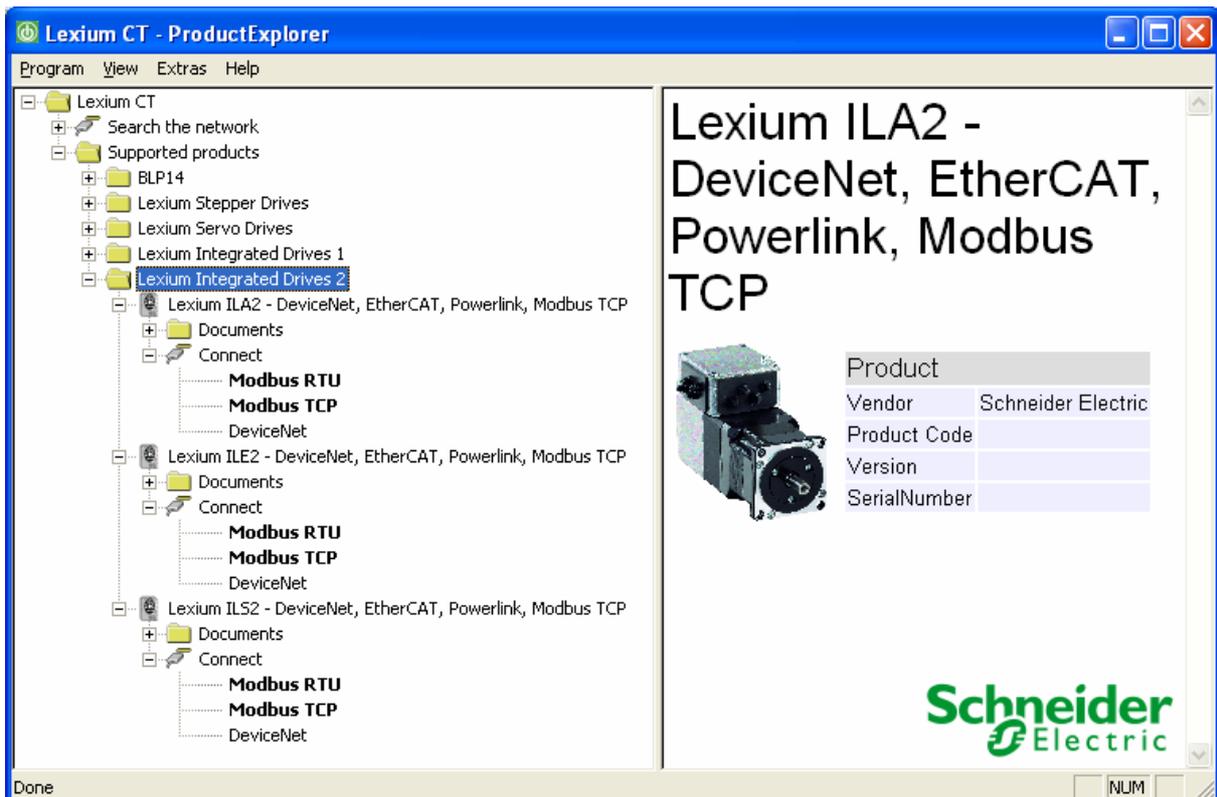


Bild 18) Lexium CT ProductExplorer

- Open the node "Lexium Integrated Drives 2" and select your type of drive.
ILA2, ILE2 or ILS2.
- There are 2 types of communication:
 1. Ethernet connection (Modbus TCP)
 2. Serial RS485 connection (Modbus RTU) with an RS 485 interface.

PLCopen AddOnInstructions ILx2 EthernetIP

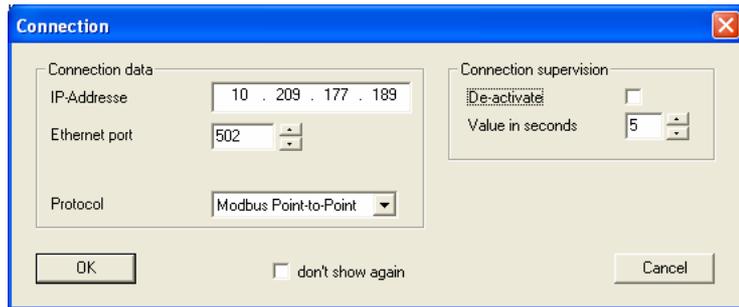


Bild 19) Modbus TCP connection

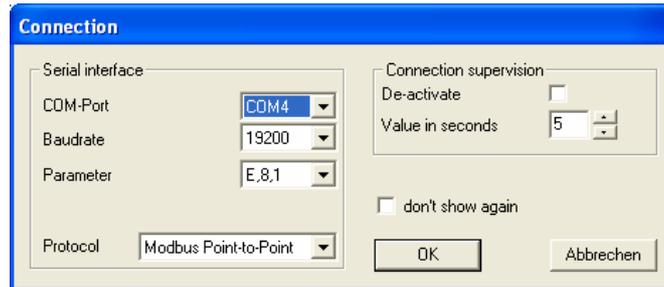


Bild 20) Modbus RTU connection (RS485 interface required)

- When connecting, the software loads all parameters from the drive.
- The Ethernet parameters are contained in the parameter group Communication -> Ethernet.

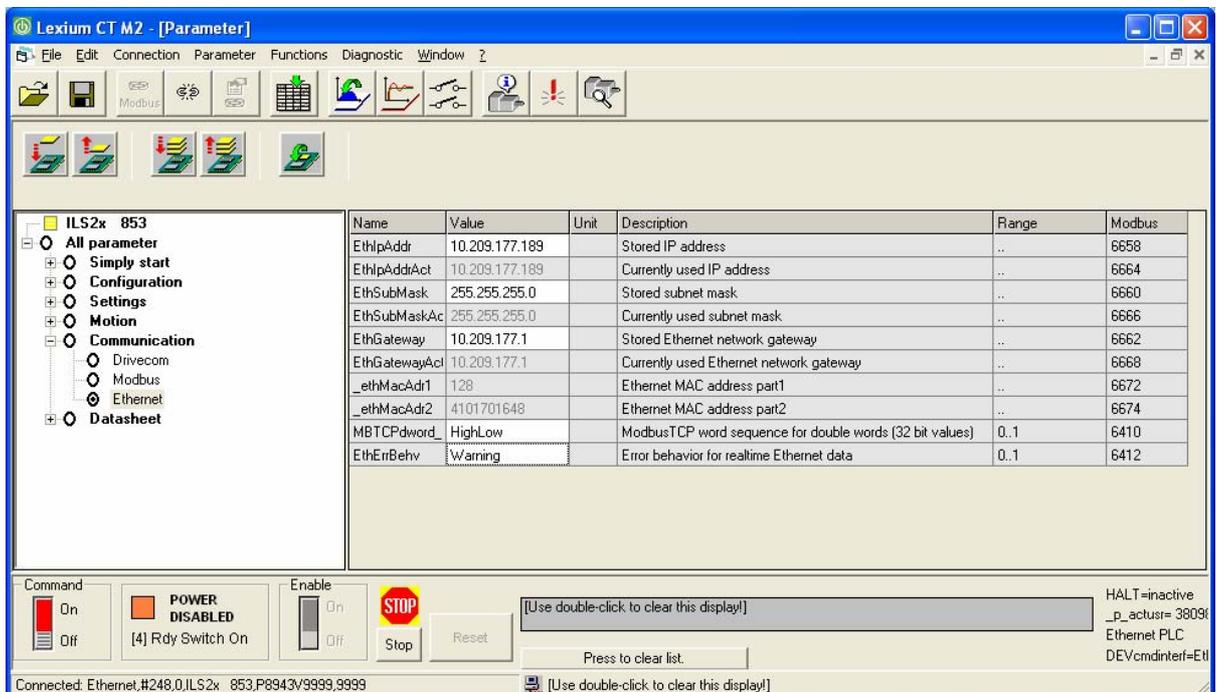


Bild 21) Lexium CT: Communication parameters -> Ethernet

- After you have changed the parameters, you must transmit them to the drive. Then the parameters must be saved to the EEPROM.

- The changed communication parameters do not become active until you switch on the drive the next time.

3 Add-On Instructions

The Add-On Instructions only control one ILx2 drive each. There are no function blocks that control several drives simultaneously or one drive in dependence on another drive. However, it is possible to create several instances of an Add-On Instruction; each instance then controls another drive independently.

3.1 Function block names

Function blocks whose names begin with the prefix Prefix *MC_* are compliant with the specification developed by the [PLCopen User Organization](#). They comply with a global standard for programming Motion Control applications.

Function blocks whose names do not contain the prefix Prefix *MC_* still comply with this standard; however, these function blocks are not yet covered by the specifications.

To identify the drives and fieldbuses for which the function blocks are used, the postfixes *_ETH* for the EtherNet/IP fieldbus and *_ILx2* for the Lexium drives are used. This allows for unique identification.

Two typical examples of function block names:

- *MC_Power_ETH_ILx2*
- *MC_MoveAbsolute_ETH_ILx2*

3.2 Signal diagrams for inputs and outputs

The signal behavior of the function blocks is uniform. There are two types of function blocks:

- Function blocks that are edge-controlled via the *Execute* input.
- Function blocks that are level-controlled via the *Enable* input.

3.2.1 Control inputs

Control input Execute

The control input *Execute* has two functions:

- A rising edge starts the execution of the function block. Input parameters such as position and velocity are taken over and the operating mode is started.
- The control outputs are enabled or disabled: As long as *Execute* = TRUE, the signal outputs signal the current state of the function block. This means that as long *Execute* = TRUE, exactly one of the outputs *Done*, *Busy*, *CommandAborted* or *Error* is also TRUE.

On the other hand, if *Execute* = FALSE before the execution of the function block is completed, *Busy* = TRUE until the execution is completed. Upon execution, one of the outputs *Done*, *CommandAborted* or *Error* for one call becomes TRUE and then

FALSE.

Control input Enable

The control input Enable starts and stops the execution of the function block (exception: MC_Power_ETH_ILx2). With TRUE, the function block is repeated. With FALSE, the execution is immediately stopped and the control outputs Valid, Busy, CommandAborted and Error are immediately set to FALSE.

3.2.2 Control outputs

Control output Done

The control output Done signals the end of the execution. This is the case, for example, when the target position of a point-to-point movement is reached.

In a number of function blocks, the output *Done* has a different name, for example *InVelocity* in the case of the function block for Profile Velocity operating mode. *InVelocity* signals that the required velocity has been reached. The execution of the function block is not terminated with this; the movement continues to be monitored.

Control output Valid

The control output Valid signals that the outputs of the function block contain valid values. For example, in the case of the function block MC_ReadParameter_ETH_ILx2, the outputs *Value* and *Length* are valid as soon as *Valid* becomes TRUE.

Control output Busy

The control output Busy signals that the function block is being executed. The execution of the function block is not completed until after *Busy* = FALSE; only after that, one of the outputs *Done*, *Error* or *CommandAborted* signals whether the function block was completed with or without an error or whether it was aborted.

Control output CommandAborted

The control output CommandAborted signals that another function block has taken over axis control. This terminates the execution of the function block that signals *CommandAborted*. For example, the function block MC_MoveVelocity_ETH_ILx2 can only be terminated by cancellation.

Signal output Error

The signal output Error signals the end of the execution if an error has occurred.

3.2.3 Signal diagrams for function blocks with Execute control input

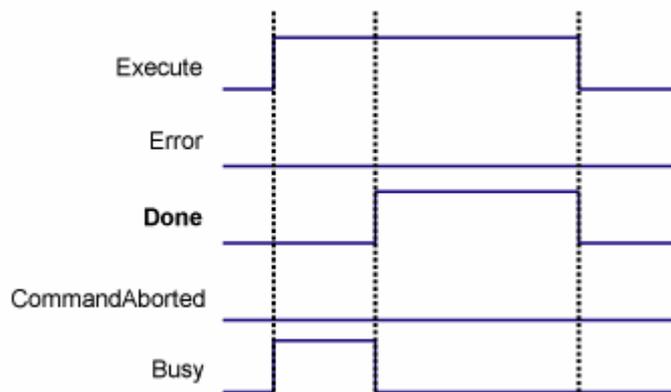


Bild 22) Execution completed without error

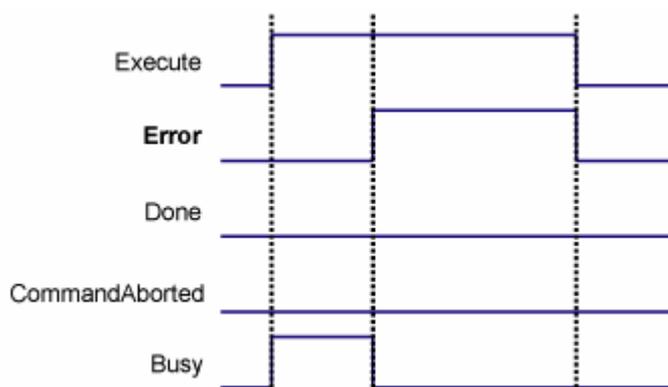


Bild 23) Execution terminated with error

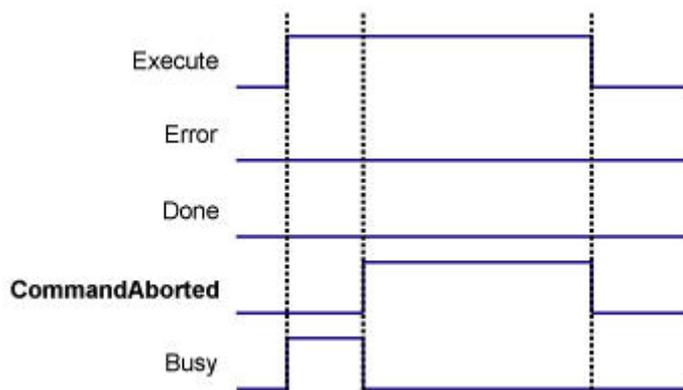


Bild 24) Cancellation of execution; another function block has taken over axis control

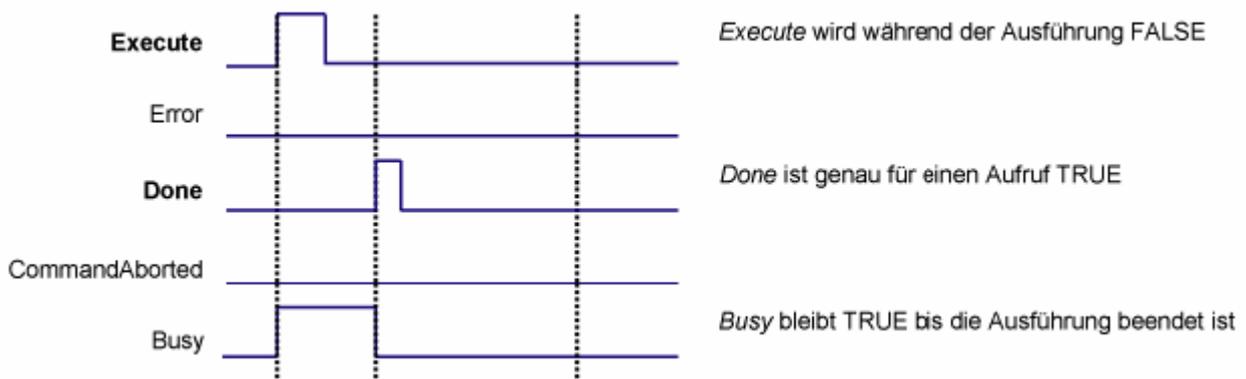


Bild 25) Execution completed without error after *Execute* was set to FALSE during execution

3.2.4 Signal diagram for function blocks with control input Enable

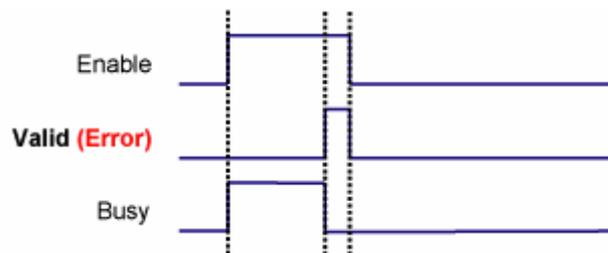


Bild 26) Single execution without (**with**) error [execution requires more than one call]

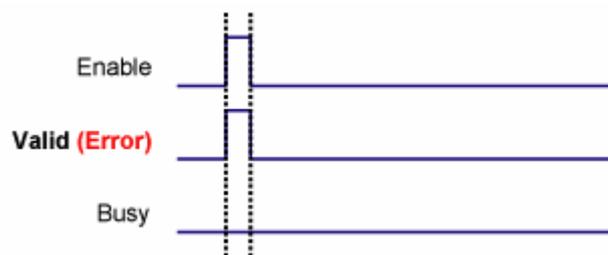


Bild 27) Single execution without (**with**) error [execution requires only one call]

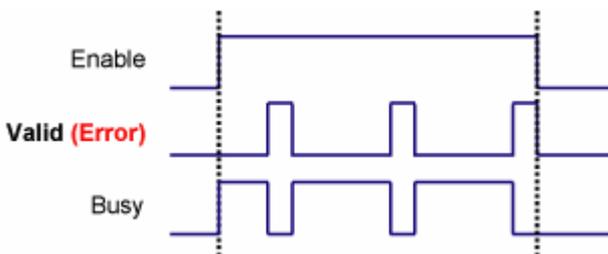


Bild 28) Multiple execution without (**with**) error [execution requires more than one call]

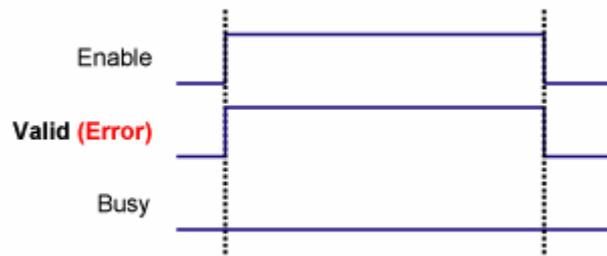


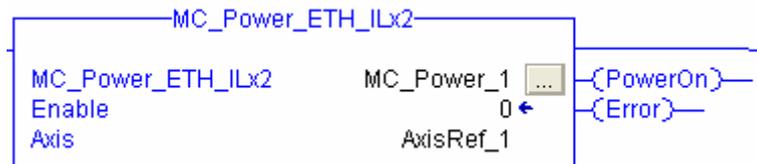
Bild 29) Multiple execution without (with) error [execution requires only one call]

3.3 Description of the function blocks

The following chapter provides descriptions of the function blocks, structured by the use of the function blocks. The chapter after that describes each function block in detail; the list is sorted alphabetically by function block name.

Structure of the detailed description of the function blocks:

Graphical representation



Inputs/outputs

Description of the inputs and outputs with specification of the type, the range of permissible values and the meaning of the values. For example:

Input variables

Variable	Possible values, meaning
Enable	Type BOOL (value range TRUE, FALSE) initial value: FALSE TRUE: Switch on motor current. FALSE: Switch off motor current.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Status	Type: BOOL (value range TRUE, FALSE) initial value: FALSE Indicates the status of the motor current. The status is signaled by the drive after each change. FALSE: Motor current is off. TRUE: Motor current is on.
Error	Type: BOOL (value range:TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Switching the motor current on/off . TRUE at the *Enable* input switches the motor current on. As soon as the motor current is switched on, the *Status* output is set. FALSE at the *Enable* input switches the motor current off . As soon as the motor is without current, the *Status* output is reset. If errors occur during execution, the *Error* output is set.

Note

When the power supply to the drive is switched off, the motor current is also switched off. Since the drive can no longer signal this state transition, *Status* remains TRUE. When the power supply to the drive is switched on again, the fact that the motor is without current is signaled and the *Status* output changes to FALSE.

Switching on the motor current causes a transition to one of the operating states *Standstill* or *ErrorStop*, depending on whether there is an error or not. The operating state is read with MC_ReadStatus_ETH_ILx2.

The motor current can be switched off in any state. Any function block that is being executed at this point is aborted.

3.4 Usage of the function blocks

3.4.1 Initialization

The initialization function block switches the drive to a state in which other function blocks can be used. Enables and disables the power stage of the drive.

Function block	Task
MC_Power_ETH_ILx2	Enables and disables the power stage of the drive.

3.4.2 Jog

The function block for jog movements moves the drive in Jog operating mode. A short signal at the inputs moves the drive by a short distance in positive or negative direction. In the case of a continuous signal, the drive starts a continuous movement.

Function block	Task
MC_Jog_ETH_ILx2	Jog in positive or negative direction

3.4.3 Homing

The operating mode Homing establishes an absolute position reference between the motor position and a defined axis position. Homing is possible by:

- Reference movement or
- Position setting

The reference movement defines the position reference by means of a movement to a switch. During the reference movement, the drive moves to a defined point on the axis. The position is defined by a mechanical switch. Depending on the motor type you use, the follow types of switches are possible:

- Limit switches in positive and negative directions
- Reference switch
- Index pulse of the motor encoder system

In addition, it is possible to combine the limit switch and the index pulse. Please refer to the product manual for information on the types of reference movement supported by your Lexium ILx2 drive.

When the position is reached, a position reference is automatically created. This way, the position becomes the absolute user position.

The search speed, the speed for moving away from the switch as the distance from the switching edge and the distance for moving away from the switch can be adjusted for the reference movement. A reference movement must be completed for the new reference point to be valid. If the reference movement is aborted, it must be started

again.

Position setting defines a position reference with reference to the current motor position. Position setting lets you set an axis position as the reference point to which the subsequent position relate. The reference point for reference positions is moved to the new position setting position.

Position setting is only possible when the motor is at a standstill. Position setting can be used to carry out continuous relative positioning without overtraveling the range.

Function block	Task
MC_Home_ETH_ILx2	Trigger reference movement
MC_SetPosition_ETH_ILx2	Position setting

3.4.4 Operating mode Profile Position

In the Profile Position operating mode, the motor is positioned from a point A to a point B by means of a function block. The positioning distance is specified in with reference to the zero point of the axis (absolute) or with reference to the original target position or the current motor position (relative). Prior to positioning, the reference point must be defined by homing.

Function block	Task
MC_MoveAbsolute_ETH_ILx2	Absolute positioning
MC_MoveRelative_ETH_ILx2	Relative positioning with reference to the current motor position
MC_MoveAdditive_ETH_ILx2	Relative positioning with reference to the original position

3.4.5 Operating mode Profile velocity

In Profile Velocity operating mode, a reference speed for the motor is set and a movement without a target position is started. The motor moves at this speed until a different speed is set or the operating mode is terminated by execution of another function block.

Transitions between two speeds of rotation are defined via the profile generator in the drive. The profile generator calculates the transition to the new speed on the basis of the parameters for acceleration, deceleration, reference speed and actual speed.

Function block	Task
MC_MoveVelocity_ETH_ILx2	Activating the operating mode Profile Velocity

3.4.6 Stopping

Every operating mode can be canceled by stopping the drive. This does not generate an error. The canceled function block terminates the execution by setting the output *CommandAborted* = TRUE. The drive switches to the operating state "Stopping" and, after the standstill and the reset of the input *Execute* in, to the state "Standstill". New movements are not possible before this is completed.

PLCopen AddOnInstructions ILx2 EthernetIP

Function block	Task
MC_Stop_ETH_ILx2	Stop drive

3.4.7 Fast position capture

The motor position can be captured with an accuracy depending on the motor type via 2 parameterizable channels. See the appropriate section in the product manual for details on fast position capture. Only the integrated drives ILA2E and ILS2E feature the fast position capture function.

Possible triggering events:

- Rising edge or falling edge at signal input CAP1
- Rising edge or falling edge at signal input CAP2

Function block	Task
MC_TouchProbe_ETH_ILx2	Adjust and start position capture
MC_AbortTrigger_ETH_ILx2	Cancel position capture

3.4.8 Reading parameters

The following function blocks allow for easy reading of specific parameters. In addition, there is a universal function block which enables a read access to all parameters of the Lexium ILx2E drive. See the product manual for detailed descriptions of all parameters of the drive.

The function blocks for reading the reference position and the reference speed directly from the profile generator only deliver different values than the other function blocks for reading position and speed in the case of the Lexium ILA2E drive. This is due to the fact that the current positions and speeds are determined via the motor encoder in the case of servo motors..

Function block	Task
MC_ReadActualPosition_ETH_ILx2	Read the the current position in user-defined units
MC_ReadActualVelocity_ETH_ILx2	Reads the current speed in user-defined units
MC_ReadStatus_ETH_ILx2	Reads the current status of the drive
MC_ReadParameter_ETH_ILx2	Reads the specified parameters from the drive

3.4.9 Writing parameters

The following function blocks allow for easy writing of specific parameters. In addition, the parameters can be reset to the factory settings or stored permanently in the drive. In addition, there is a universal function block which enables a write access to all parameters of the Lexium ILx2E drive. See the product manual for detailed descriptions of all parameters of the drive.

Function block	Task
----------------	------

Function block	Task
MC_WriteParameter_ETH_ILx2	Set parameter

3.4.10 Inputs/outputs

The digital inputs and digital outputs of the Lexium ILx2E can be read with the following parameters.

Function block	Task
MC_ReadDigitalInput_ETH_ILx2	Read inputs

3.4.11 Error handling

For error handling purposes, each function block has an *Error* output which is set when an error occurs. The function block MC_ReadAxisError_ETH_ILx2 is called for detailed analysis of the cause of the error. The error cell is deleted and available for future error messages with MC_Reset_ETH_ILx2.

The error cell contains the error code and the error class of an error . A new error is entered provided the error cell is free. If the error cell not free, the previous error message is not overwritten; instead, the new error message is ignored.

Possible error sources comprise:

- Function block errors
- Drive errors

Drive errors can be the result of, for instance, invalid input values. Drive errors are usually caused by events such as reaching a limit switch. MC_Reset_ETH_ILx2 also resets the error in the drive.

Function block	Task
MC_ReadAxisError_ETH_ILx2	Read error
MC_Reset_ETH_ILx2	Delete error

3.5 Detailed description of function blocks

3.5.1 MC_UpdateInputData_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
CommunicationOK	Type: BOOL (value range: FALSE,TRUE initial value: FALSE The communication status is not checked by the AddOnInstructions. The use has to check this status by the instruction GSV (GetSystemValue). The ststus "RUNNING" can be assigned by a tag of the type BOOL to this input. If the communication status is not checked, the value "TRUE" has to be assigned to this input.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis> initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.
ScannerIn	Type: ETHERNET MODULE INPUT DATA Initial value: empty Data array of the Ethernet input data from the Controller Tag list.

Output variables

This function block has no output variable.

Task of the function block

This function block must be called at lfirst in the PLC cycle before another Add-On Instruction for the drive is called. This function block

copies the ScannerInputTags into the Axis_Ref structure which is used by any other AddOnInstruction.

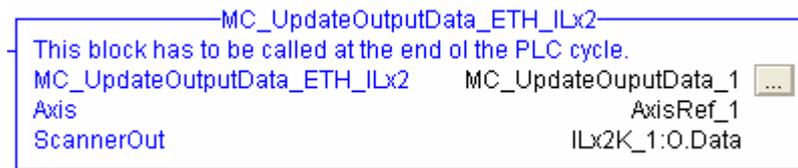
The Axis_Ref structure enables access to the drive through several Add-On Instructions.

Note

This function must be called cyclically as the first AddOnInstruction.

3.5.2 MC_UpdateOutputData_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
ScannerOut	Type: ETHERNET MODULE OUTPUT DATA Initial value: empty Data array of the Ethernet output data from the Controller Tag list.

Task of the function block

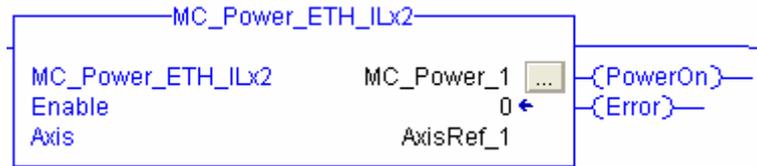
This block copies the output data from the AxisRef structure to the EthernetIP controller output tags.

Note

This function must be called cyclically as the last AddOnInstruction.

3.5.3 MC_Power_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range TRUE, FALSE) initial value: FALSE TRUE: Switch on motor current. FALSE: Switch off motor current.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Status	Type: BOOL (value range TRUE, FALSE) initial value: FALSE Indicates the status of the motor current. The status is signaled by the drive after each change. FALSE: Motor current is off. TRUE: Motor current is on.
Error	Type: BOOL (value range:TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Switching the motor current on/off . TRUE at the *Enable* input switches the motor current on. As soon as the motor current is switched on, the *Status* output is set. FALSE at the *Enable* input

switches the motor current off . As soon as the motor is without current, the *Status* output is reset. If errors occur during execution, the *Error* output is set.

Note

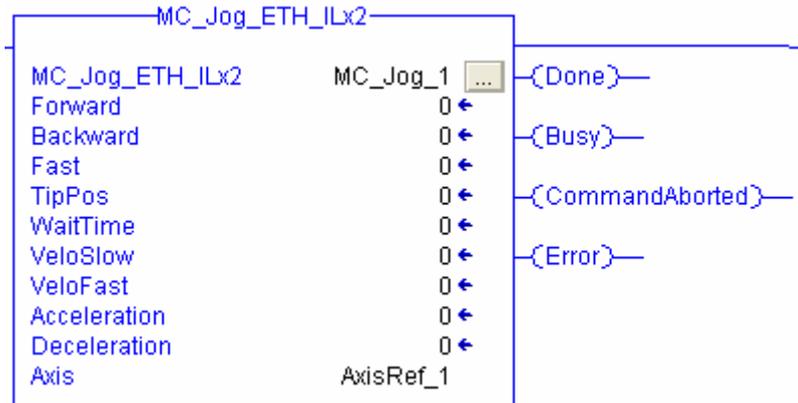
When the power supply to the drive is switched off, the motor current is also switched off. Since the drive can no longer signal this state transition, *Status* remains TRUE. When the power supply to the drive is switched on again, the fact that the motor is without current is signaled and the *Status* output changes to FALSE.

Switching on the motor current causes a transition to one of the operating states *Standstill* or *ErrorStop*, depending on whether there is an error or not. The operating state is read with `MC_ReadStatus_ETH_ILx2`.

The motor current can be switched off in any state. Any function block that is being executed at this point is aborted.

3.5.4 MC_Jog_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Forward	Type BOOL (value range TRUE, FALSE) initial value: FALSE FALSE: No movement in positive direction TRUE: Start of the movement in positive direction
Backward	Type BOOL (value range TRUE, FALSE) initial value: FALSE FALSE: No movement in negative direction TRUE: Start of the movement in negative direction
Fast	Type BOOL (value range TRUE, FALSE) initial value: FALSE The speed can also be selected during movements from two values: FALSE: Speed VeloSlow is selected. TRUE: Speed VeloFast is selected.
TipPos	Type DINT (value range: 0..2147483647) initial value: 20 0: The motor immediately starts a continuous movement. >0: Distance [usr] by which the motor moves after the start before it switches to continuous

Variable	Possible values, meaning
	movement after the delay time (WaitTime) has elapsed.
WaitTime	Type UINT (value range: 1ms..32767ms) initial value: 500ms Delay time [ms], which starts after the motor has moved a defined distance (TipPos) and after which the motor switches to continuous movement.
VeloSlow	Type DINT (value range: 1..13200) initial value: 60 Speed [min-1] for movement if Fast = FALSE.
VeloFast	Type: DINT (value range: 1..13200) initial value: 180 Speed [min-1] for movement if Fast = TRUE.
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600 Value for steepness of acceleration ramp [1min-1/s]
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed.

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	FALSE: Execution terminated; the function block is not active.
CommandAborted	Type BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range:TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

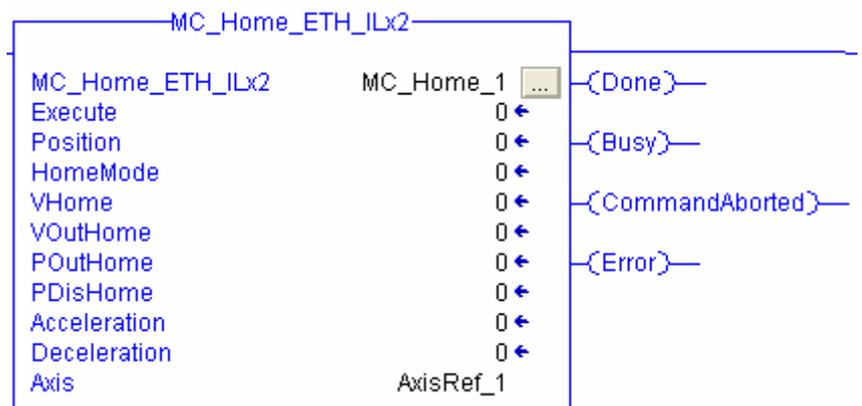
A jog movement is started with TRUE at *Forward* or *Backward*.

If *Forward* and *Backward* are FALSE, the operating mode is terminated and *Done* is set.

If *Forward* and *Backward* are TRUE, the operating mode remains active, the jog movement is stopped and *Busy* remains set.

3.5.5 MC_Home_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the

Variable	Possible values, meaning
	<p>execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE .</p> <p>After termination of the execution, Execute determines the behavior of the outputs:</p> <p>FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call.</p> <p>TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.</p>
Position	<p>Type DINT</p> <p>(value range:-2147483648..2147483647) initial value: 0</p> <p>Position setting to position setting position (setting the absolute position) in user-defined units. Position is the current motor position after successful homing.</p>
HomeMode	<p>Type: UINT</p> <p>(value range 1..35) initial value: 1</p> <p>Please see the product manual for the homing methods supported by the drive.</p>
Vhome	<p>Type: UINT</p> <p>(value range: 1..13200) initial value: 60</p> <p>Speed for searching for the reference switch or a limit switch [min-1]. The drive stops when the switching edge has been detected.</p>
VoutHome	<p>Type: UINT</p> <p>(value range: 1..3000) initial value: 6</p> <p>Speed for moving away from the switch to the switching edge [min-1]. The maximum distance for searching for the switching edge can be limited with the parameter POutHome.</p>
PoutHome	<p>Type: DINT</p> <p>(value range: 0..2147483647) initial value: 0</p> <p>0: Monitoring of moving away from switch inactive</p> <p>> 0: Distance for search for switching edge during movement away from switch [usr], i.e. maximum movement for searching the switching edge. If the switching edge is not found in this distance, the reference movement is aborted with an error.</p>
PdisHome	<p>Type: DINT</p>

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	(value range: 1..2147483647) initial value: 200 Distance from switching edge to reference point in user-defined units When the switching edge is reached, the drive continues to move until the distance is covered.
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600 Value for steepness of acceleration ramp [1min-1/s]
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE, FALSE) initial value: FALSE

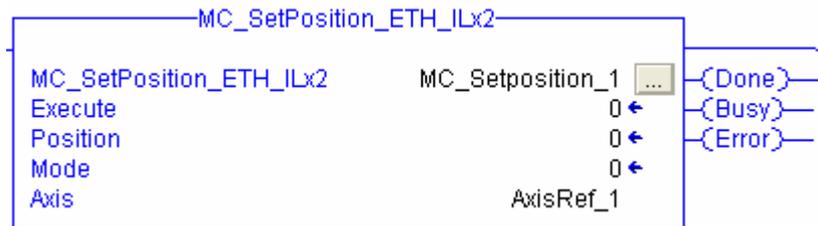
Variable	Possible values, meaning
	TRUE: Execution was terminated with an error.
	FALSE: No error has (yet) occurred during execution.

Task of the function block

Configuration and start of the reference movement.

3.5.6 MC_SetPosition_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Position	Type: DINT (value range: -2147483648..2147483647) initial value: 0 Position setting to position setting position in user-defined units
Mode	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE FALSE: Set current motor position to position.

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	TRUE: Add position to current motor position.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

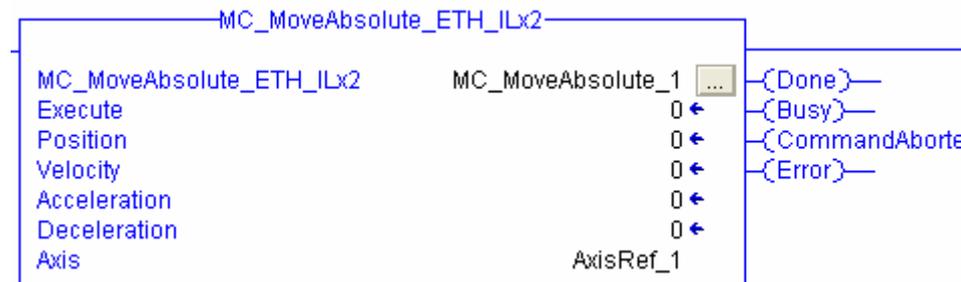
Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Position setting

3.5.7 MC_MoveAbsolute_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	<p>Type BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE .</p> <p>After termination of the execution, Execute determines the behavior of the outputs:</p> <p>FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call.</p> <p>TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.</p>
Position	<p>Type: DINT</p> <p>(value range: -2147483648..2147483647) initial value: 0</p> <p>Value for the absolute target position in user-defined units.</p>
Velocity	<p>Type: INT</p> <p>(value range: 1..13200) initial value: 60)</p> <p>Value for the reference speed of the movement [min-1].</p>
Acceleration	<p>Type: DINT</p> <p>(value range: 1..3000000) initial value: 600</p> <p>Value for steepness of acceleration ramp [1min-1/s]</p>
Deceleration	<p>Type: DINT</p> <p>(value range: 200..3000000) initial value: 750</p> <p>Value for steepness of deceleration ramp [1min-1/s]</p>
Axis	<p>Type: AXIS_REF_EIP_ILx2</p> <p>(value range: <name of axis>) initial value: empty</p> <p>Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.</p>

PLCopen AddOnInstructions ILx2 EthernetIP

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE, FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Positioning to absolute target position *Position* at speed *Velocity*

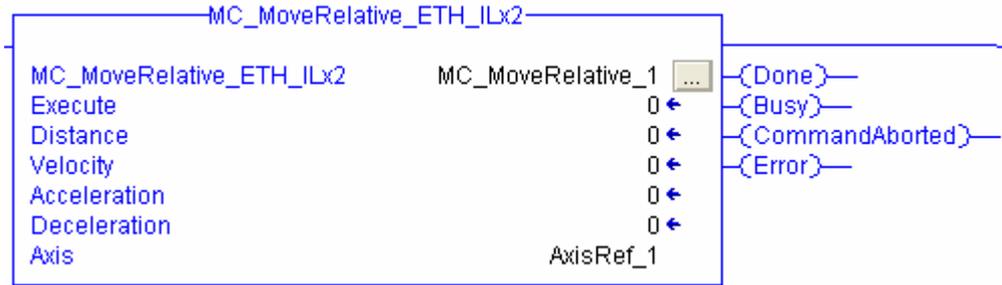
Note

Position overrun

Absolute positioning cannot be started after a position overrun, because the absolute position reference is lost during the position overrun. The *Referenced* output of the function block `MC_ReadStatus_ETH_ILx2` allows you to read whether or not the absolute position reference is still available.

3.5.8 MC_MoveRelative_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Distance	Type: DINT (value range: -2147483648..2147483647) initial value: 0 Value for the distance with reference to the current motor position in user-defined units.
Velocity	Type: INT (value range: 1..13200) initial value: 60) Value for the reference speed of the movement [min-1].
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600 Value for steepness of acceleration ramp [1min-1/s]

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

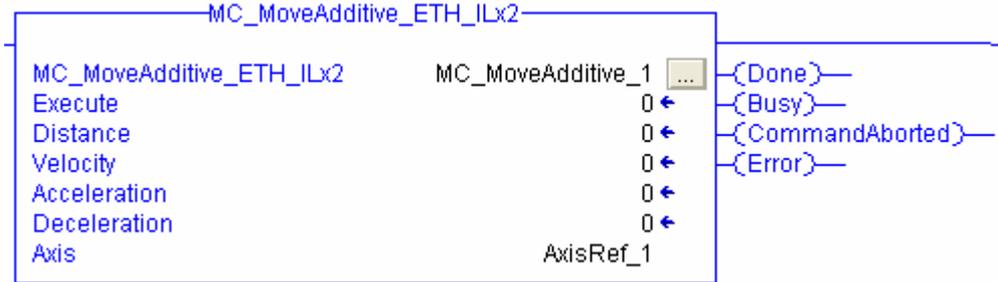
Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Positioning by distance *Distance* at speed *Velocity*

3.5.9 MC_MoveAdditive_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Distance	Type: DINT (value range: -2147483648..2147483647) initial value: 0 Value for the original target position plus additional relative distance in user-defined units.
Velocity	Type: INT (value range: 1..13200) initial value: 60) Value for the reference speed of the movement [min-1].
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600 Value for steepness of acceleration ramp

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	[1min-1/s]
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

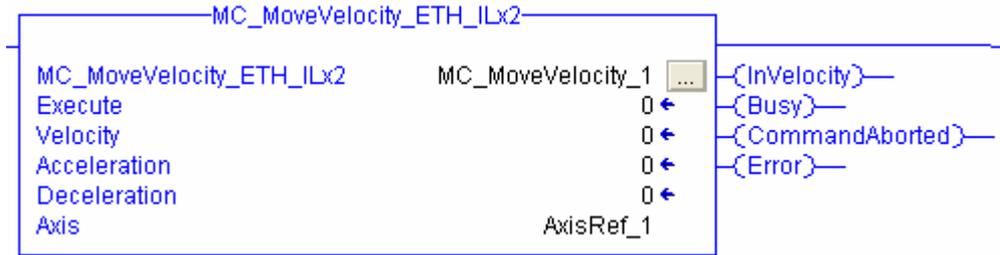
Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Positioning by distance *Distance* in addition to the original target position as speed *Velocity*.

3.5.10 MC_MoveVelocity_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Velocity	Type: INT (value range: -13200...13200) initial value: 0 Value for the reference speed of the movement [min-1].
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600 Value for steepness of acceleration ramp [1min-1/s]
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
InVelocity	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE FALSE: Target speed not (yet) reached. TRUE: Target speed reached.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Start operating mode Profile Velocity with speed *Velocity*. When the target speed is reached, *InVelocity* is set.

Note

Changing *Velocity* to "0"

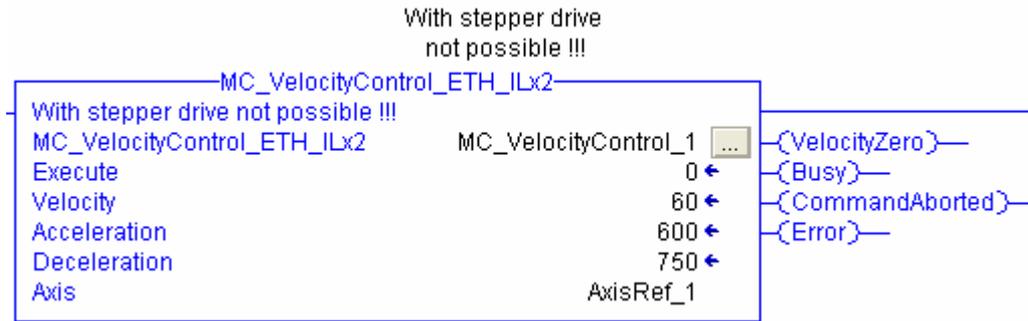
This allows you to stop the drive with the normal deceleration at any time. If the speed is set to "0", the drive is only stopped temporarily! This means that as soon as the speed is set to a value not equal to "0", the drive will immediately resume movement.

Position range overrun

In the Profile Velocity operating mode, the drive may exceed the position range. This is not an error for the drive, the operating mode continues to run. The *Referenced* output of the function block MC_ReadStatus_ETH_ILx2 allows you to read whether or not the absolute position reference is still available.

3.5.11 MC_VelocityControl_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Velocity	Type: INT (value range: -13200...13200) initial value: 0 Value for the reference speed of the movement [min-1].
Acceleration	Type: DINT (value range: 1..3000000) initial value: 600

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	Value for steepness of acceleration ramp [1min-1/s]
Deceleration	Type: DINT (value range: 200..3000000) initial value: 750 Value for steepness of deceleration ramp [1min-1/s]
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
VelocityZero	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE FALSE: Actual velocity is not equal to 0 TRUE: Actual velocity is equal to 0. Drive is at a standstill.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
CommandAborted	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.
Error	Type: BOOL (value range: TRUE, FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

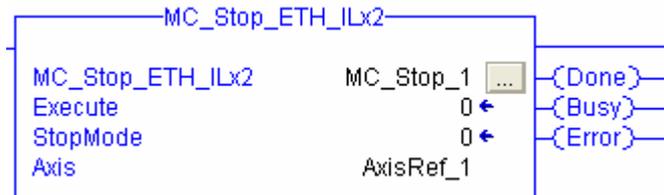
Start operating mode Velocity Control with speed *Velocity*. When the drive is at a standstill, *VelocityZero* is set.

Note

This operating mode is only supported by the drives ILA2 und ILE2, it is not implemented for the stepper motor drive ILS2.

3.5.12 MC_Stop_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
StopMode	Type: DINT (value range: 0..1) initial value: 0 0 = HALT 1 = Quick Stop HALT / Quick Stop configuration: see product manual
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

PLCopen AddOnInstructions ILx2 EthernetIP

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE, FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Stops the drive with a torque ramp.

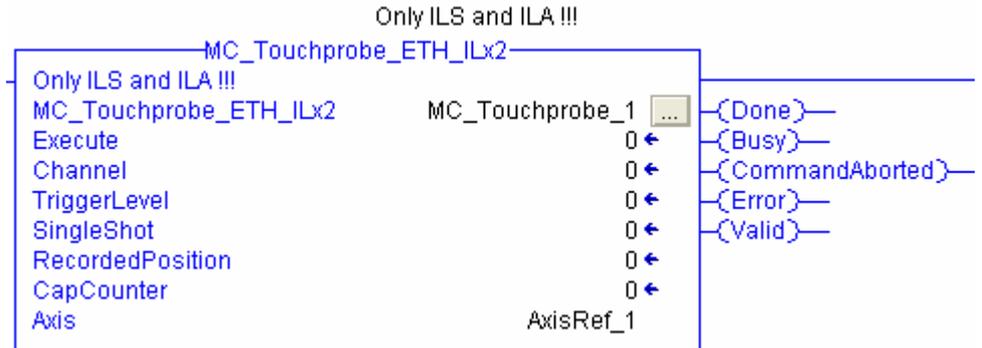
Note

This function cannot be aborted by other function blocks. As long as Execute = TRUE, no other function block can be started. Even after standstill, the drive remains blocked.

The function block decelerates the motor with a torque ramp. The parameter LIM_I_maxHalt (see product manual) specifies the current for the torque ramp. After the drive has come to a standstill, the position is determined internally, position control is activated and the motor is held with the power stage enabled.

3.5.13 MC_TouchProbe_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Channel	Type: UINT (value range: 1..2) initial value 1 Channel number: Selection of the channel to which the other parameters relate.
TriggerLevel	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Triggering signal edge: FALSE: Falling edge TRUE: Rising edge
SingleShot	Type: BOOL (value range: FALSE, TRUE) initial value: TRUE

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	<p>FALSE: If the triggering event occurs again, the captured position is overwritten with the most new position.</p> <p>TRUE: Position capture is switched off after the triggering event so that the captured position cannot be overwritten..</p>
Axis	<p>Type: AXIS_REF_EIP_ILx2</p> <p>(value range: <name of axis>) initial value: empty</p> <p>Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.</p>

Output variables

Variable	Possible values, meaning
Done	<p>Type: BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>TRUE: Execution terminated. FALSE: Execution not (yet) terminated.</p>
Busy	<p>Type: BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.</p>
CommandAborted	<p>Type: BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>TRUE: Execution was canceled by another function block. FALSE: Execution not (yet) canceled.</p>
Error	<p>Type: BOOL</p> <p>(value range: TRUE; FALSE) initial value: FALSE</p> <p>TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.</p>
Valid	<p>Type: BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>TRUE: Execution terminated. A read value at the parameter output RecordedPosition is valid. FALSE: Execution not (yet) terminated. A value at the parameter output RecordedPosition is not (yet) valid.</p>

Variable	Possible values, meaning
RecordedPosition	Type DINT (value range: -2147483648..2147483647) initial value: 0 Captured motor position when the triggering event occurs

Task of the function block

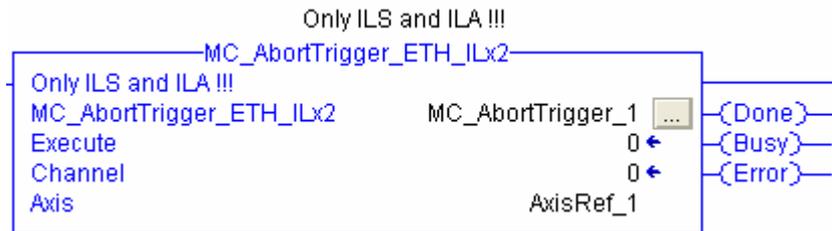
Adjust and start position capture.

Note

Only the drives ILA2K and ILS2K feature the fast position capture function.

3.5.14 MC_AbortTrigger_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call. TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Channel	Type: UINT (value range: 1..2)

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	initial value 1 1: Abort position capture via channel 1 (CAP1). 2: Abort position capture via channel 2 (CAP2).
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

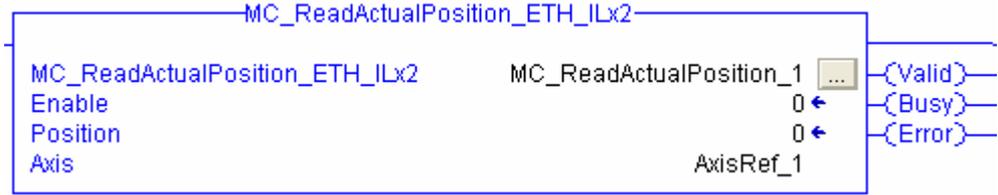
Abort position capture

Note

Only the drives ILA2K and ILS2K feature the fast position capture function.

3.5.15 MC_ReadActualPosition_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE

PLCopen AddOnInstructions ILx2 EthernetIP

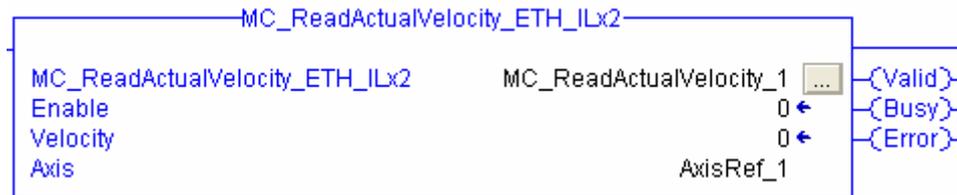
Variable	Possible values, meaning
	TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.
Position	Type: DINT (Value range:) Actual motor position in user-defined units

Task of the function block

Read the actual position of the motor in user-defined units

3.5.16 MC_ReadActualVelocity_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE

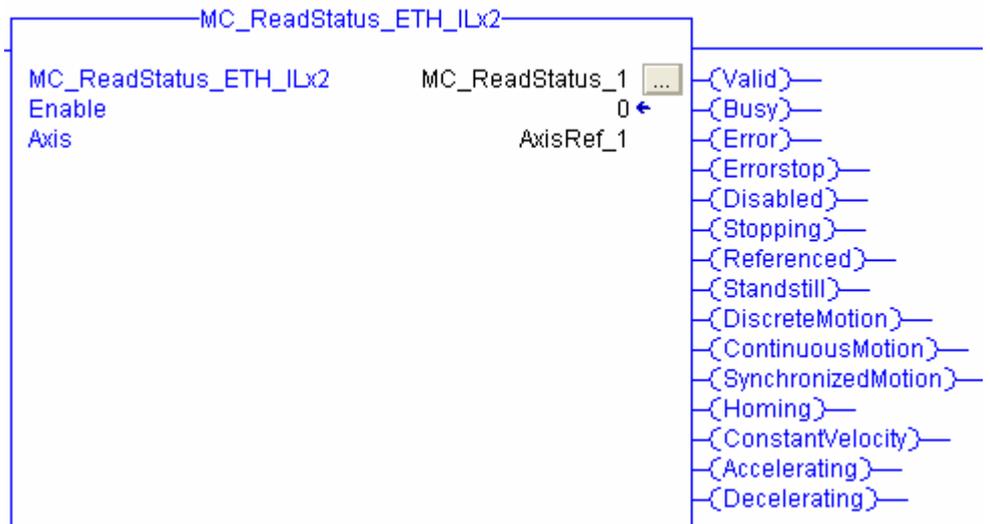
Variable	Possible values, meaning
	TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.
Velocity	Type: INT (value range: -13200..+13200) Current speed of rotation of motor [min-1]

Task of the function block

Read the current speed of rotation of the motor [min-1]

3.5.17 MC_ReadStatus_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	(value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.
ErrorStop	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Movement was stopped due to an error
Disabled	Type: BOOL (value range: FALSE, TRUE) Initial value: FALSE TRUE: Motor current is switched off. FALSE: Motor current is switched on.

Variable	Possible values, meaning
Stopping	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Function block MC_Stop_ETH_ILx2 is executed; movement is being stopped.
Referenced	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Drive is homed; position reference with relation to the mechanical system is known.
StandStill	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Drive is at a standstill.
DiscreteMotion	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Drive is in Profile Position operating mode.
ContinuousMotion	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The drive is in an operating mode with continuous motion, the Profile Velocity operating mode.
SynchronizedMotion	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The drive is in an operating mode with synchronized motion, such as Electronic Gear operating mode.
Homing	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE) TRUE: The drive is in operating mode Homing.
ConstantVelocity	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The motor rotates at constant speed.
Accelerating	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The drive accelerates.
Decelerating	Type: BOOL (value range: FALSE, TRUE)

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	initial value: FALSE
	TRUE: The drive decelerates.

Task of the function block

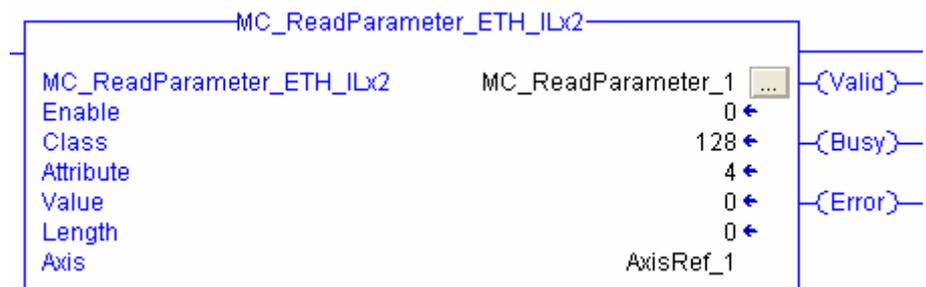
The current state of the drive is output.

Note

The drive is exactly in one of the states *StandStill*, *Homing*, *DiscreteMotion*, *ContinuousMotion*, *SynchronizedMotion*, *Stopping*, *Disabled* or *Errorstop*. The output with the corresponding name of the function block is then TRUE.

3.5.18 MC_ReadParameter_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Class	Type: UINT (value range: 0..65535) Class of the object to be read; the objects are listed in the product manual with class, instance and attribute. Since the drive only works with Attribute = 1, this value is permanently set to value 1.
Attribute	Type: UINT;

Variable	Possible values, meaning
	(value range: 0..255) Attribute of the object to be read.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

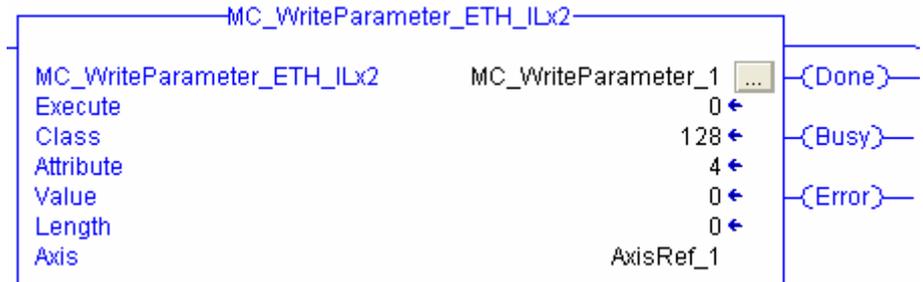
Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.
Value	Type: DINT (value range: -2147483648..2147483647) initial value: 0 Value of device parameter
Length	Type: UINT (value range: 0..65535) initial value: 0 Length in bytes of read device parameter

Task of the function block

Read an object from the device parameter list.

3.5.19 MC_WriteParameter_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	<p>Type: BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE .</p> <p>After termination of the execution, Execute determines the behavior of the outputs:</p> <p>FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call.</p> <p>TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.</p>
ParameterNumber	<p>Type: INT</p> <p>(value range: 0..65535)</p> <p>0: Parameter is selected with index and subindex.</p> <p>>0: Number of the parameter to be written:</p> <p>2: Position of positive software limit switch [usr]</p> <p>3: Position of negative software limit switch [usr]</p> <p>4: Enable (bit0=1) or disable (bit0=0) positive software limit switch</p> <p>5: Enable (bit0=1) or disable (bit0=0) negative software limit switch</p> <p>Other numbers are not supported.</p>
Value	<p>Type: DINT</p> <p>(value range: -2147483648..2147483647) initial value: 0</p>

Variable	Possible values, meaning
	Value of device parameter
Index	Type: UINT (value range: 0..65535) Index of the object to be written; the objects are listed in the manual with their indexes and subindexes. Only valid if ParameterNumber = 0.
Subindex	Type: UINT (value range: 0..255) Subindex of the object to be written; the objects are listed in the manual with their indexes and subindexes. Only valid if ParameterNumber = 0.
Length	Type: UINT (value range: 0..65535) initial value: 0 Length in bytes of device parameter to be written
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

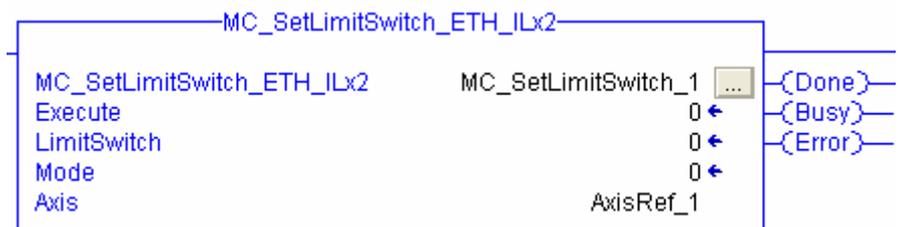
PLCopen AddOnInstructions ILx2 EthernetIP

Task of the function block

Write an object in the parameter list.

3.5.20 MC_SetLimitSwitch_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	<p>Type BOOL</p> <p>(value range: FALSE, TRUE) initial value: FALSE</p> <p>Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE .</p> <p>After termination of the execution, Execute determines the behavior of the outputs:</p> <p>FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call.</p> <p>TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.</p>
LimitSwitch	<p>Type INT</p> <p>(value range: 1..2) initial value: 1</p> <p>1 = LimP 2= LimN</p>
Mode	<p>Type INT</p> <p>(value range: 0..2) initial value: 0</p> <p>0 = Inactive 1 = Normally closed 2 = Normally open</p>

Variable	Possible values, meaning
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

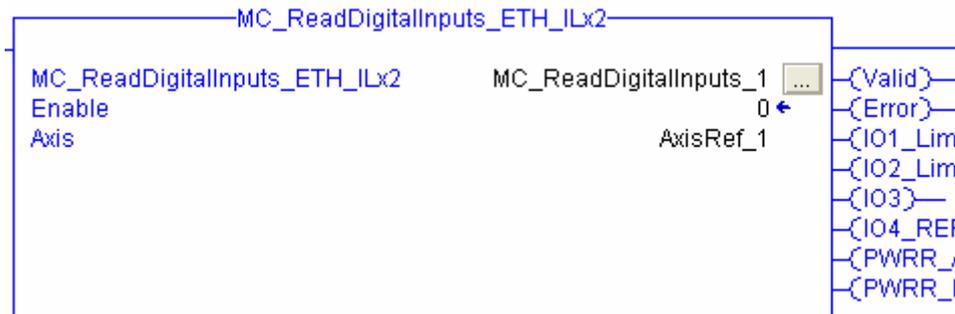
Limit switch configuration.
 The positive limit switch (LimP) and the negative limit witch (LimN) are parameterized with this function block.
 Mode = 0 : Limit switch is disabled.
 Mode = 1 : Limit switch is configured as a normally closed contact
 Mode = 2 : Limit switch is configured as a normally open contact

Note

If the configuration is to remain active even after a Power Off, the command SaveEEPROM must be transmitted to the drive with the function"MC_WriteParameter".

3.5.21 MC_ReadDigitalInputs_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. The read value at the parameter output <i>Value</i> is valid. FALSE: Execution not (yet) terminated. The read value at the parameter output <i>Value</i> is not (yet) valid.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed.

Variable	Possible values, meaning
	FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range:TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

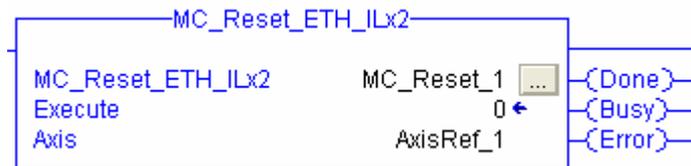
Reads the current input assignment of the drive.

Note

The meaning of the signals is described in the product manual.
 The input I0 (/REF) is only used by the drive for homing to the reference signal (see MC_Home_ETH_ILx2). If this function is not used, the input can be used for other purposes as required.
 The limit switch function of the inputs I1 (/LIMN) and I2 (/LIMP) can be disabled. If the limit switch function is not used, the inputs can be used for other purposes as required.

3.5.22 MC_Reset_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Execute	Type BOOL (value range: FALSE, TRUE) initial value: FALSE Rising edge: FALSE->TRUE starts the execution. A new rising edge continues the execution with the input parameters that are then active. Execution is terminated once the Busy output is FALSE . After termination of the execution, Execute determines the behavior of the outputs: FALSE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE for exactly one call.

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
	TRUE: At the same time as Busy = FALSE, either Done, Error or CommandAborted becomes TRUE and remains TRUE until the function block is called with Execute = FALSE.
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

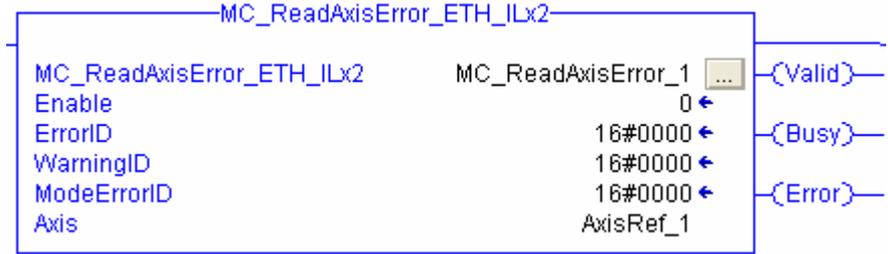
Variable	Possible values, meaning
Done	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. FALSE: Execution not (yet) terminated.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed. FALSE: Execution terminated; the function block is not active.
Error	Type: BOOL (value range: TRUE; FALSE) initial value: FALSE TRUE: Execution was terminated with an error. FALSE: No error has (yet) occurred during execution.

Task of the function block

Error acknowledgement. The error cell is cleared so that it is available for future error messages. If the motor has been stopped by the automatic error response, it will be enabled again, provided that the cause of the error has been rectified when the error message is acknowledged.

3.5.23 MC_ReadAxisError_ETH_ILx2

Graphical representation



Input variables

Variable	Possible values, meaning
Enable	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE Level-sensitive; starts or stops execution of the function block. TRUE: Function block is executed repeatedly. FALSE: Execution is terminated immediately; the control outputs Valid, Busy and Error become FALSE immediately
Axis	Type: AXIS_REF_EIP_ILx2 (value range: <name of axis>) initial value: empty Name of the drive for which the function block is to be executed. A global data structure must be created for each drive. This data structure is passed here as a parameter.

Output variables

Variable	Possible values, meaning
Valid	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: Execution terminated. The read value at the parameter output ErrorID is valid. FALSE: Execution not (yet) terminated. The read value at the parameter output ErrorID is not (yet) valid.
Busy	Type: BOOL (value range: FALSE, TRUE) initial value: FALSE TRUE: The function block is being executed.

PLCopen AddOnInstructions ILx2 EthernetIP

Variable	Possible values, meaning
Error	<p>FALSE: Execution terminated; the function block is not active.</p> <p>Type: BOOL</p> <p>(value range: TRUE; FALSE) initial value: FALSE</p> <p>TRUE: Execution was terminated with an error.</p> <p>FALSE: No error has (yet) occurred during execution.</p>
ErrorID	<p>Type: WORD</p> <p>(value range: 0000_h ... FFFF_h) initial value: 0000_h</p> <p>0: No error message in the error cell</p> <p>> 0: Error number (see list of error numbers in the appendix).</p>

Task of the function block

Reads device error.

4 Communication monitoring

This chapter provides a suggestion for monitoring the communication with the drive in the application.

The project "SE_Motion_ETH_ILx2.ACD" contains the sample routine "CheckIOState_SingleAxis", which monitors the communication with the drive.



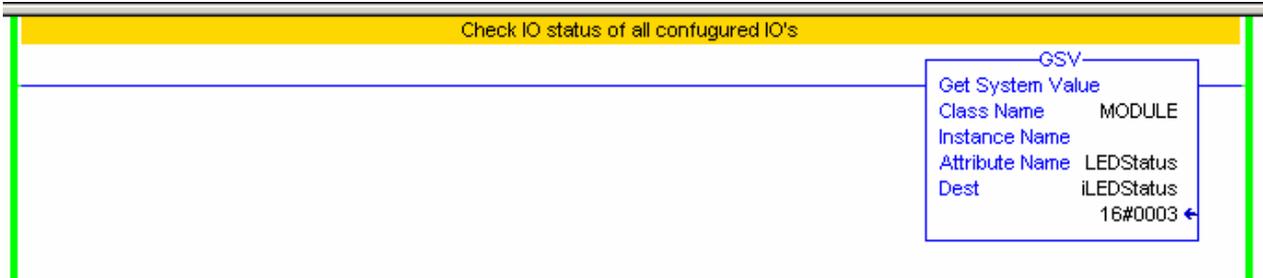
4.1 Reading the I/O LED state

This routine lets you read the status of the I/O LED. It allows you to monitor the entire I/O configuration.

- Define a variable of type INT.

+ iLEDStatus	16#0003	Hex	INT
--------------	---------	-----	-----

- Call the routine GSV (Get System Value) in the following way.



- After the routine has been called, the status of the I/O LED can be evaluated:
iLEDStatus:
0 I/O LED is OFF
 No I/O connections configured.
- 1 I/O LED flashes (red)
 None of the configured I/O connections work properly.
- 2 I/O LED flashes (green)
 Some of the configured I/O connections work properly. At least one of the configured I/O connections does not work properly.
- 3 I/O LED steady (green)
 All I/O connection work properly.

PLCopen AddOnInstructions ILx2 EthernetIP

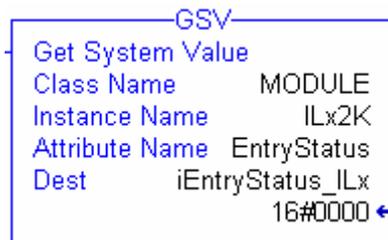
4.2 Monitoring individual I/O connections

This routine lets you read the status of individual I/O connections. It allows you to monitor the EtherNet/IP communication of each individual drive.

- Define a variable of type INT.

⊕ iEntryStatus_ILx		INT	Hex
--------------------	--	-----	-----

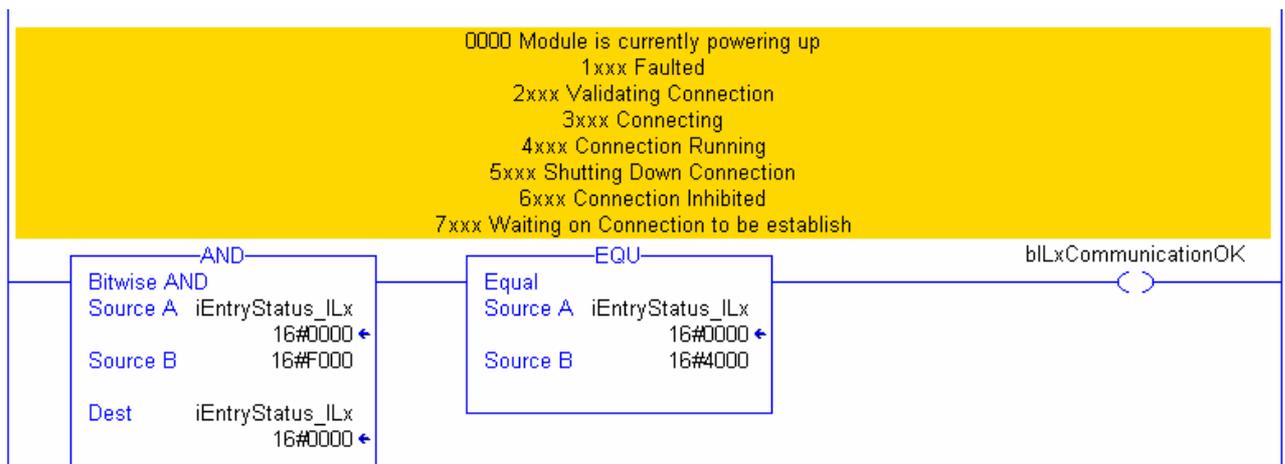
- Call the routine GSV (Get System Value) in the following way. The input "Instance Name" must correspond to the drive name in the I/O configuration.



- Determining the device status:

- 0000 Module is currently powering up
- 1xxx Fehler
- 2xxx Validating Connection
- 3xxx Connecting
- 4xxx Connection Running
- 5xxx Shutting Down Connection
- 6xxx Connection Inhibited
- 7xxx Waiting on Connection to be establish

- The following illustration shows an example "Communication OK".



5 Appendix

5.1 Error numbers

The error numbers are the return values of the function block MC_ReadAxisError_ETH_ILx2.

ErrorID hex.	ErrorID dec.	Error class	Description
Up to 00FF _h	Up to 255		See CiA405 error messages
Drive error messages			
1100 _h	4352	0	Parameter out of permissible range
1101 _h	4353	0	Parameter does not exist (index)
1102 _h	4354	0	Parameter does not exist (subindex)
1103 _h	4355	0	Writing of parameter not permissible (read only)
1104 _h	4356	0	Write access denied (no access authorization)
1106 _h	4358	0	Command not allowed while power stage is active
1107 _h	4359	0	Access via other interface blocked
1108 _h	4360	0	Parameter cannot be read (Block Upload)
1109 _h	4360	0	Power fail data invalid
110A _h	4362	0	No bootloader present
110B _h	4363	3	Initialization error
1300 _h	4864	3	Safety function Safe Torque OFF triggered (STO_A and STO_B)
1301 _h	4865	4	Inputs of the STO_A and STO_B safety function have different levels
1310 _h	4880	3	Reference signal frequency too high
1603 _h	5635	0	Capture memory used by other function
1606 _h	5638	0	Capture still active
1607 _h	5639	0	No trigger parameter defined for capture
1608 _h	5640	0	Trigger option not permissible for trigger parameter
1609 _h	5641	0	No capture channel defined
160A _h	5642	0	No capture data available
160B _h	5643	0	Parameter cannot be logged
160C _h	5644	1	Autotuning: Moment of inertia outside of permissible range
160E _h	5646	1	Autotuning: Test movement could not be started
160F _h	5647	1	Autotuning: Power stage cannot be enabled
1610 _h	6548	1	Autotuning: Processing aborted
1611 _h	5649	1	System error: Autotuning internal write access
1612 _h	5650	1	System error: Autotuning internal read access
1613 _h	5651	1	Autotuning: Maximum permissible positioning range exceeded
1614 _h	5652	0	Autotuning: Already active
1617 _h	5655	1	Autotuning: Friction torque or load torque too great
1618 _h	5656	1	Autotuning: Optimization unsuccessful
1A00 _h	6656	0	System error: FIFO memory overflow
1A01 _h	6657	3	Motor has been changed
1A02 _h	6658	3	Motor has been changed
1B00 _h	6912	4	System error: Incorrect parameters for motor and power stage
1B01 _h	6913	3	User parameter max. speed of rotation too high
1B02 _h	6914	3	User parameter maximum current, holding current or Quick Stop current too high
2300 _h	8960	3	Power stage overcurrent
2301 _h	8961	3	Braking resistor overcurrent
3100 _h	12544	par.	Phase error mains supply
3200 _h	12800	3	DC bus overvoltage

PLCopen AddOnInstructions ILx2 EthernetIP

ErrorID hex.	ErrorID dec.	Error class	Description
3201 _h	12801	3	DC bus undervoltage (switch-off threshold)
3202 _h	12802	2	DC bus undervoltage (Quick Stop threshold)
3203 _h	12803	4	Motor encoder supply voltage
3206 _h	12806	0	DC bus undervoltage (warning)
4100 _h	16640	3	Power stage overtemperature
4101 _h	16641	0	Warning power stage overtemperature
4102 _h	16642	0	Warning power stage overload (I ² t)
4200 _h	16896	3	Device overtemperature
4300 _h	17152	3	Motor overtemperature
4301 _h	17153	0	Warning motor overtemperature
4302 _h	17154	0	Warning motor overload (I ² t)
4402 _h	17410	0	Warning braking resistor overload (I ² t)
5200 _h	20992	3	No connection to motor encoder
5201 _h	20993	4	Error in motor encoder communication
5202 _h	20994	4	Motor encoder is not supported
5203 _h	20995	4	No connection to motor encoder
5204 _h	20996	3	Connection to motor encoder lost
5430 _h	21552	0	System error: EEPROM read error
5431 _h	21553	0	System error: EEPROM write error
5435 _h	21557	0	System error: EEPROM not formatted
5437 _h	21559	0	System error: EEPROM checksum error manufacturer data
5438 _h	21560	0	System error: EEPROM checksum error user parameters
5439 _h	21561	0	System error: EEPROM checksum error CAN parameters
543A _h	21562	0	System error: EEPROM HardwareInfo invalid
543B _h	21563	0	System error: EEPROM manufacturer data invalid
543C _h	21564	0	System error: EEPROM CAN data invalid
543D _h	21565	0	System error: EEPROM user parameters invalid
5600 _h	22016	3	Motor connection phase error
5601 _h	22017	4	Interruption or incorrect motor encoder signals
5602 _h	22018	4	Interruption or incorrect motor encoder signals
5603 _h	22019	4	Commutation error
6107 _h	24839	0	Parameter outside of value range (calculation error)
6108 _h	24840	0	Function not available
610D _h	24845	0	Error in selection parameter
610F _h	24847	4	Internal time base failed (timer 0)
7120 _h	28960	4	Invalid motor data
7121 _h	28961	2	System error: Error in motor encoder communication
7122 _h	28962	4	Impermissible motor data
7123 _h	28963	4	Motor current offset outside of permissible range
7124 _h	28964	4	System error: motor encoder defective
7200 _h	29184	4	System error: calibration analog/digital converter
7201 _h	29185	4	System error: Motor encoder initialization (quadrant evaluation)
7327 _h	29479	4	System error: Position sensor not ready
7328 _h	29480	4	Motor encoder signals: Incorrect position capture
7329 _h	29481	0	Motor encoder signals: Warning
7330 _h	29482	4	System error: Motor encoder (Hiperface)
7331 _h	29483	4	System error: Motor encoder initialization
7333 _h	29485	4	System error: Deviation in calibration of analog/digital converter
7334 _h	29486	3	System error: Analog/digital converter offset too great
7335 _h	29487	0	Communication with motor encoder busy
7336 _h	29488	3	Offset during SinCos drift compensation too high
7337 _h	29489	1	Writing of offset not be successful
7400 _h	29696	0	System error: Invalid interrupt (XINT2)

ErrorID hex.	ErrorID dec.	Error class	Description
7500 _h	29952	0	Modbus: Overrun error
7501 _h	29953	0	Modbus: Framing error
7502 _h	29954	0	Modbus: Parity error
7503 _h	29955	0	Modbus:Rreceive error
8110 _h	33040	0	CANopen over EtherCAT: CAN overflow (message lost)
8130 _h	33072	2	CANopen over EtherCAT: Heartbeat or Life Guard error
8201 _h	33281	0	CANopen over EtherCAT: RxPDO1 could not be processed
8202 _h	33282	0	CANopen over EtherCAT: RxPDO2 could not be processed
8203 _h	33283	0	CANopen over EtherCAT: RxPDO3 could not be processed
8204 _h	33284	0	CANopen over EtherCAT: RxPDO4 could not be processed
8205 _h	33285	0	CANopen over EtherCAT: TxPDO could not be processed
8206 _h	33286	0	CANopen over EtherCAT: Overflow internal queue message lost
A060 _h	41056	2	Calculation error electronic gear
A061 _h	41057	2	Change in reference value for Electronic Gear too great
A300 _h	41728	0	Torque ramp with HALT current active
A301 _h	41729	0	Drive in state 'QuickStopActive'
A302 _h	41730	1	Interruption by LIMP
A303 _h	41731	1	Interruption by LIMN
A304 _h	41732	1	Interruption by REF
A305 _h	41733	0	Power stage cannot be enabled in current operating state of state machine
A306 _h	41734	1	Interruption by user-initiated software stop
A307 _h	41735	0	Interruption by internal software stop
A308 _h	41736	0	Drive in state 'Fault'
A309 _h	41737	0	Drive not in state 'OperationEnable'
A310 _h	41744	0	Power stage not enabled
A312 _h	41746	0	Profile generation interrupted
A313 _h	41747	0	Position overrun (pos_over=1), therefore, reference point is no longer defined (ref_ok=0)
A314 _h	41748	0	No reference position
A315 _h	41749	0	Homing active
A316 _h	41750	0	Overrun during calculation of acceleration
A317 _h	41751	0	Drive is not at a standstill
A318 _h	41752	0	Operating mode active (x_end=0)
A319 _h	41753	1	Manual tuning/autotuning: Distance range exceeded
A31A _h	41754	0	Manual tuning/Autotuning: Amplitude/offset too high
A31B _h	41755	0	HALT requested
A31C _h	41756	0	Invalid position setting with software limit switch
A31D _h	41757	0	Speed range exceeded (CTRL_n_max)
A31E _h	41758	1	Interruption by positive software limit switch
A31F _h	41759	1	Interruption by negative software limit switch
A320 _h	41760	par.	Position tracking error
A321 _h	41761	0	RS422 position interface is not defined as input
A324 _h	41764	1	Error during homing (additional info = detailed error number)
A325 _h	41765	1	Limit switch to be approached not enabled
A326 _h	41766	1	REF switch not found between LIMP and LIMN
A327 _h	41767	1	Reference movement to REF without reversal of direction, impermissible activation of limit switch LIM
A328 _h	41768	1	Reference movement to REF without reversal of direction, overtraveling of of LIM or REF not permissible
A329 _h	41769	1	More than one signal LIMP/LIMN/REF active
A32A _h	41770	1	External monitoring signal LIMP with negative direction
A32B _h	41771	1	External monitoring signal LIMN with positive direction
A32C _h	41772	1	Reference movement error at REF (e.g. by impact)

PLCopen AddOnInstructions ILx2 EthernetIP

ErrorID hex.	ErrorID dec.	Error class	Description
A32D _h	41773	1	Reference movement error at LIMP (e.g. by impact)
A32E _h	41774	1	Reference movement error at LIMN (e.g. by impact)
A32F _h	41775	1	Index pulse not found
A330 _h	41776	0	Reference movement to index pulse cannot be reproduced. Index pulse is too close to the switch.
A331 _h	41777	3	No start-up operating mode selected for local control mode
A332 _h	41778	1	Jog error (additional info = detailed error number)
A334 _h	41780	2	Timeout during standstill window monitoring
A335 _h	41781	1	Processing only possible in fieldbus mode
B100 _h	45312	0	Modbus: Unknown service
B200 _h	45568	0	Modbus: Protocol error
B201 _h	45569	2	Modbus: Nodeguard error
B202 _h	45570	0	Modbus: Nodeguard warning
B203 _h	45571	0	Modbus: Incorrect number of monitor objects
B204 _h	45572	0	Modbus: Service too long
B600 _h	46592		Ethernet: Initialization error
B601 _h	46593		Ethernet: Realtime data error
B602 _h	46594		Ethernet: Realtime data warning
B603 _h	46595		Ethernet: Protocol-specific error
B604 _h	46596		Ethernet: Protocol-specific warning
B605 _h	46597		Ethernet: Unknown error
B606 _h	46598		Ethernet: Delayed parameter access to module
B607 _h	46599		Ethernet: Is currently processing another request
B608 _h	46600		Ethernet: Realtime Hot Reset
B609 _h	46601		Ethernet: Realtime Hot Stop
B60A _h	46602		Ethernet: Timeout in internal communication
B60B _h	46603		Ethernet: Error in internal communication
Library error messages			
FF10 _h	65296	0	Data type and parameter length do not match
FF22 _h	65314	0	An attempt was made to interrupt a non-interruptible function block (MC Power, MC Stop, MC Home, MC SetPosition)
FF23 _h	65315	0	Trigger function already active
FF24 _h	65316	0	Timeout Toggle Bit
FF27 _h	65319	0	Drive is not in state StandStill
FF2A _h	65322	0	Trigger event lost
FF34 _h	65332	0	Power stage does not switch to state Enabled
FF36 _h	65334	0	Operating mode is not supported by the drive
FF37 _h	65335	0	Power stage is not in state Enabled
FF3B _h	65339	0	Power stage is not in state Disabled