SoMachine
Motion Control Library Guide

05/2017

EIO0000002221.02

www.schneider-electric.com

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

You agree not to reproduce, other than for your own personal, noncommercial use, all or part of this document on any medium whatsoever without permission of Schneider Electric, given in writing. You also agree not to establish any hypertext links to this document or its content. Schneider Electric does not grant any right or license for the personal and noncommercial use of the document or its content, except for a non-exclusive license to consult it on an "as is" basis, at your own risk. All other rights are reserved.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2017 Schneider Electric. All Rights Reserved.

Table of Contents

Safety Information 7
About the Book 11
Chapter 1 General Description of Motion Control Libraries 17
1.1 General Description 18
Libraries and Related Function Blocks 19
PLCopen State Diagram 23
Transitions Between Function Blocks 24
Library Diagnostic Codes 27
1.2 Common Inputs and Outputs 33
Behavior of Function Blocks with the Input `Enable` 34
Behavior of Function Blocks with the Input `Execute` 37
Chapter 2 Function Blocks - Single Axis 39
2.1 Initialization 40
`MC_Power` 40
2.2 Operating Mode Jog 43
`MC_Jog` 44
`Jog_LXM32` 47
`Jog_ILX` 51
2.3 Operating Mode Profile Torque 55
`MC_TorqueControl` 56
`SetTorqueRamp_LXM32` 59
`TorqueControl_LXM32` 61
2.4 Operating Mode Profile Velocity 64
`MC_MoveVelocity` 65
`MoveVelocity_LXM32` 69
`VelocityControlAnalogInput_ATV` 72
`VelocityControlSelectAI_ATV` 75
`Control_ATV` 78
2.5 Operating Mode Profile Position 84
`MC_MoveAbsolute` 85
`MC_MoveAdditive` 88
`MC_MoveRelative` 91

2.6 Operating Mode Homing 94
MC_Home 95
Home_LXM32 98
Home_ILX 102
MC_SetPosition 106
2.7 Stopping 108
MC_Stop 109
SetStopRamp_LXM32 112
SetStopRamp_ILX 114
Stop_LXM32 116
MC_Halt 119
Halt_LXM32 122
2.8 Position Capture via Signal Input 124
MC_TouchProbe 125
TouchProbe_LXM32 128
TouchProbe_ILX 131
MC_AbortTrigger 134
Chapter 3 Function Blocks - Multi Axis 137
3.1 Operating Mode Electronic Gear 138
GearInPos_LXM32 139
GearIn_LXM32 142
Chapter 4 Function Blocks - Administrative 145
4.1 Reading a Parameter 146
MC_ReadActualTorque 147
MC_ReadActualVelocity 149
MC_ReadActualPosition 151
MC_ReadAxisInfo 153
MC_ReadMotionState 156
MC_ReadStatus 159
MC_ReadParameter 162
4.2 Writing a Parameter 165
MC_WriteParameter 166
SetDriveRamp_ATV 169
SetDriveRamp_LXM32 171
SetDriveRamp_ILX 173
SetLimitSwitch_LXM32 175
SetLimitSwitch_ILX 177
SetFrequencyRange_ATV 179

ResetParameters_ATV 182
ResetParameters_LXM32 184
ResetParameters_ILX 186
StoreParameters_ATV 188
StoreParameters_LXM32 190
StoreParameters_ILX 192
4.3 Inputs and Outputs 194
ReadAnalogInput_ATV 195
MC_ReadDigitalInput 198
MC_ReadDigitalOutput 201
MC_WriteDigitalOutput 204
4.4 Error Handling 207
ReadAxisWarning_LXM32 208
ReadAxisWarning_ILX 210
MC_ReadAxisError 212
MC_Reset 214
Glossary 217
Index 219

Safety Information

Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, service, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result** in minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

BEFORE YOU BEGIN

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

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

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

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

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

About the Book

At a Glance

Document Scope

The function blocks of the three libraries described here are used under the SoMachine software environment to control:

- ATV320, ATV340, ATV6•• and ATV9•• drives over CANopen, Ethernet/IP and Modbus/TCP,
- ATV32, ATV71, LXM32M drives and Lexium ILA, ILE and ILS integrated drives over Ethernet/IP and Modbus/TCP.

The function blocks are compliant with the IEC 61131-3 standard.

Validity Note

This document has been updated for the release of SoMachine V4.3.

The former title of this guide is SoMachine Industrial Ethernet - Motion Control Library Guide.

Related Documents

Title of documentation	Reference number
SoMachine - Programming Guide	EIO0000000067 (eng) EIO0000000069 (fre) EIO0000000068 (ger) EIO0000000071 (spa) EIO0000000070 (ita) EIO0000000072 (chi)
SoMachine - Functions and Libraries User Guide	EIO0000000735 (eng) EIO0000000792 (fre) EIO0000000793 (ger) EIO0000000795 (spa) EIO0000000794 (ita) EIO0000000796 (chi)
SoMachine Industrial Ethernet - User Guide	EIO0000002215 (eng) EIO0000002216 (fre) EIO0000002217 (ger) EIO0000002218 (spa) EIO0000002219 (ita) EIO0000002220 (chi)

Title of documentation	Reference number
Altivar 32 Variable speed drives for synchronous and asynchronous motors - Programming Manual	S1A28692 (eng) S1A28693 (fre) S1A28694 (ger) S1A28695 (spa) S1A28696 (ita) S1A28697 (chi)
ATV32 - Communication Parameters	S1A44568 (eng)
Altivar 32 Variable speed drives for synchronous and asynchronous motors - Modbus TCP - EtherNet/IP Communication Manual	S1A28701 (eng)
Altivar Machine ATV320 Variable Speed Drives for Asynchronous and Synchronous Motors - Programming Manual	NVE41295 (eng) NVE41296 (fre) NVE41297 (ger) NVE41298 (spa) NVE41299 (ita) NVE41300 (chi)
ATV320 - Communication Parameters	NVE41316 (eng)
Altivar Machine 320 Variable Speed Drives for Asynchronous and Synchronous Motors - CANopen Manual - VW3A3608, 618, 628	NVE41309 (eng)
Altivar Machine ATV320 Variable Speed Drives For Asynchronous and Synchronous Motors - Modbus TCP - EtherNet/IP Manual - VW3A3616	NVE41313 (eng)
Altivar Machine ATV340 Variable Speed Drives for Asynchronous and Synchronous Motors - Programming Manual	NVE61643 (eng) NVE61644 (fre) NVE61645 (ger) NVE61647 (spa) NVE61648 (ita) NVE61649 (chi)
ATV340 - Communication Parameters	NVE61728 (eng)
Altivar Machine ATV340 Variable Speed Drives for Asynchronous and Synchronous Motors - CANopen Manual - VW3A3608, 618, 628	NVE61655 (eng)
Altivar Machine ATV340 Variable Speed Drives for Asynchronous and Synchronous Motors - Ethernet Embedded Manual	NVE61653 (eng)
Altivar 71 Variable speed drives for synchronous and asynchronous motors - Programming manual	1755855 (eng) 1755854 (fre) 1755856 (ger) 1755857 (spa) 1755858 (ita) 1755859 (chi)
Altivar 71 Communication parameters - User manual	1755861 (eng)
Altivar 61/71 EtherNet/IP™ card - User manual VW3 A3 316	AAV68822 (eng)
Altivar 61/71 Modbus TCP/IP Ethernet card - User manual VW3 A3 310	1755879 (eng)
Altivar 61/71 Modbus TCP/IP Daisy Chain Ethernet card - User manual VW3 A3 310d	AAV69931 (eng)

Title of documentation	Reference number
Altivar 61/71 Modbus TCP/IP card - User manual	HRB10064 (eng)
Altivar 61/71 EtherNet/IP™ card - User manual VW3A3320	HRB10065 (eng)
Altivar Process - Variable Speed Drives ATV630, ATV650, ATV660, ATV680 - Programming Manual	EAV64318 (eng) EAV64320 (fre) EAV64321 (ger) EAV64322 (spa) EAV64323 (ita) EAV64324 (chi) EAV64318RU (rus)
ATV600 - Communication parameters	EAV64332 (eng)
Altivar Process ATV600 Variable Speed Drive for Asynchronous and Synchronous Motors - CANopen Manual - VW3A3608, 618, 628	EVA64333 (eng)
Altivar Process ATV600 - Variable Speed Drives for Asynchronous and Synchronous Motors - Embedded Ethernet Manual	EAV64327 (eng)
Altivar Process ATV600 - Variable Speed Drives for Asynchronous and Synchronous Motors - EthernetIP Modbus TCP Manual: VW3A3720, VW3A3721	EAV64328 (eng)
Altivar Process - Variable Speed Drives ATV930, ATV950, ATV960, ATV980, ATV993 - Programming Manual	NHA80757 (eng) NHA80758 (fre) NHA80759 (ger) NHA80760 (spa) NHA80761 (ita) NHA80762 (chi)
Altivar Process 900 - CANopen Manual - VW3A3608, 618, 628	NHA80945 (eng)
Altivar Process 900 - Ethernet manual - Embedded	NHA80940 (eng)
LXM32M AC servo drive - Product manual	0198441113767 (eng) 0198441113768 (fre) 0198441113766 (ger) 0198441113770 (spa) 0198441113769 (ita) 0198441113771 (chi)
LXM32M EtherNet/IP module - Fieldbus manual	0198441113802 (eng) 0198441113803 (fre) 0198441113801 (ger)
LXM32M Modbus/TCP module - Fieldbus manual	0198441113843 (eng) 0198441113844 (fre) 0198441113842 (ger)
ILA2K EtherNet/IP Lexium Integrated Drive - Product manual	0198441113670 (eng) 0198441113669 (ger)
ILE2K EtherNet/IP Lexium Integrated Drive - Product manual	0198441113676 (eng) 0198441113675 (ger)

Title of documentation	Reference number
ILS2K EtherNet/IP Lexium Integrated Drive - Product manual	0198441113682 (eng) 0198441113681 (ger)
ILA2T Modbus/TCP Lexium Integrated Drive - Product manual	0198441113652 (eng) 0198441113651 (ger)
ILE2T Modbus/TCP Lexium Integrated Drive - Product manual	0198441113658 (eng) 0198441113657 (ger)
ILS2T Modbus/TCP Lexium Integrated Drive - Product manual	0198441113664 (eng) 0198441113663 (ger)

You can download these technical publications and other technical information from our website at *http://www.schneider-electric.com/en/download*.

Product Related Information

⚠ WARNING

LOSS OF CONTROL

- The designer of any control scheme must consider the potential failure modes of control paths and, for certain critical control functions, provide a means to achieve a safe state during and after a path failure. Examples of critical control functions are emergency stop and overtravel stop, power outage and restart.
- Separate or redundant control paths must be provided for critical control functions.
- System control paths may include communication links. Consideration must be given to the implications of unanticipated transmission delays or failures of the link.
- Observe all accident prevention regulations and local safety guidelines.[1]
- Each implementation of this equipment must be individually and thoroughly tested for proper operation before being placed into service.

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

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), "Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control" and to NEMA ICS 7.1 (latest edition), "Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems" or their equivalent governing your particular location.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

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

Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in this manual, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives and general automation, this may include, but is not limited to, terms such as *safety*, *safety function*, *safe state*, *fault*, *fault reset*, *malfunction*, *failure*, *error*, *error message*, *dangerous*, etc.

Among others, these standards include:

Standard	Description
EN 61131-2:2007	Programmable controllers, part 2: Equipment requirements and tests.
ISO 13849-1:2008	Safety of machinery: Safety related parts of control systems. General principles for design.
EN 61496-1:2013	Safety of machinery: Electro-sensitive protective equipment. Part 1: General requirements and tests.
ISO 12100:2010	Safety of machinery - General principles for design - Risk assessment and risk reduction
EN 60204-1:2006	Safety of machinery - Electrical equipment of machines - Part 1: General requirements
EN 1088:2008 ISO 14119:2013	Safety of machinery - Interlocking devices associated with guards - Principles for design and selection
ISO 13850:2006	Safety of machinery - Emergency stop - Principles for design
EN/IEC 62061:2005	Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems
IEC 61508-1:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements.
IEC 61508-2:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems.
IEC 61508-3:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements.
IEC 61784-3:2008	Digital data communication for measurement and control: Functional safety field buses.
2006/42/EC	Machinery Directive
2014/30/EU	Electromagnetic Compatibility Directive
2014/35/EU	Low Voltage Directive

In addition, terms used in the present document may tangentially be used as they are derived from other standards such as:

Standard	Description
IEC 60034 series	Rotating electrical machines
IEC 61800 series	Adjustable speed electrical power drive systems
IEC 61158 series	Digital data communications for measurement and control – Fieldbus for use in industrial control systems

Finally, the term *zone of operation* may be used in conjunction with the description of specific hazards, and is defined as it is for a *hazard zone* or *danger zone* in the *Machinery Directive* (*2006/42/EC*) and *ISO 12100:2010*.

NOTE: The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

Chapter 1
General Description of Motion Control Libraries

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
1.1	General Description	18
1.2	Common Inputs and Outputs	33

Section 1.1
General Description

What Is in This Section?

This section contains the following topics:

Topic	Page
Libraries and Related Function Blocks	19
PLCopen State Diagram	23
Transitions Between Function Blocks	24
Library Diagnostic Codes	27

Libraries and Related Function Blocks

Library Overview

Libraries:

- The GMC Independent PLCopen MC library
  Library name: GMC Independent PLCopen MC *(see page 20)*
  Namespace: GIPLC
- The GMC Independent Altivar library
  Library name: GMC Independent Altivar *(see page 21)*
  Namespace: GIATV
- The GMC Independent Lexium library
  Library name: GMC Independent Lexium *(see page 21)*
  Namespace: GILXM

Naming conventions:

- Function blocks with the prefix MC_ (**Motion Control**) are compliant with the PLCopen specification V2.00. They conform to a global standard for programming motion control applications.
- Function blocks without a prefix MC_ are vendor-specific (Schneider Electric); however, they comply with the general PLCopen rules.

Supported Drives and Fieldbuses

Drive	CANopen	EtherNet/IP	Modbus/TCP
ATV320, ATV340, ATV6••, ATV9••	✓	✓	✓
ATV32, ATV71, LXM32M, Lexium ILA, ILE and ILS integrated drives	-	✓	✓
✓ Supported - Not supported			

Function Blocks of GMC Independent PLCopen MC (GIPLC)

The **GMC Independent PLCopen MC** library contains these function blocks that can be used with Altivar, Lexium 32, and Lexium ILA, ILE and ILS drives:

Category	Function block	Description
Single axis	MC_Power *(see page 40)*	Initialization
	MC_Jog *(see page 44)*	Operating mode: Jog
	MC_TorqueControl *(see page 56)*	Operating mode: Profile Torque
	MC_MoveVelocity *(see page 65)*	Operating mode: Profile Velocity
	MC_MoveAbsolute *(see page 85)*	Operating mode: Profile Position
	MC_MoveAdditive *(see page 88)*	
	MC_MoveRelative *(see page 91)*	
	MC_Home *(see page 95)*	Operating mode: Homing
	MC_SetPosition *(see page 106)*	
	MC_Stop *(see page 109)*	Stopping
	MC_Halt *(see page 119)*	
	MC_TouchProbe *(see page 125)*	Position capture through signal input
	MC_AbortTrigger *(see page 134)*	
Administrative	MC_ReadActualTorque *(see page 147)*	Reading a parameter
	MC_ReadActualVelocity *(see page 149)*	
	MC_ReadActualPosition *(see page 151)*	
	MC_ReadAxisInfo *(see page 153)*	
	MC_ReadMotionState *(see page 156)*	
	MC_ReadStatus *(see page 159)*	
	MC_ReadParameter *(see page 162)*	
	MC_WriteParameter *(see page 166)*	Writing a parameter
	MC_ReadDigitalInput *(see page 198)*	Inputs and outputs
	MC_ReadDigitalOutput *(see page 201)*	
	MC_WriteDigitalOutput *(see page 204)*	
	MC_ReadAxisError *(see page 212)*	Error handling
	MC_Reset *(see page 214)*	

Function Blocks of GMC Independent Altivar (GIATV)

The **GMC Independent Altivar** library contains these function blocks that can be used with Altivar drives:

Category	Function block	Description
Single axis	VelocityControlAnalogInput_ATV *(see page 72)*	Operating mode: Profile Velocity
	VelocityControlSelectAI_ATV *(see page 75)*	
	Control_ATV *(see page 78)*	
Administrative	SetDriveRamp_ATV *(see page 169)*	Writing a parameter
	SetFrequencyRange_ATV *(see page 179)*	
	ResetParameters_ATV *(see page 182)*	
	StoreParameters_ATV *(see page 188)*	
	ReadAnalogInput_ATV *(see page 195)*	Inputs and outputs

Function Blocks of GMC Independent Lexium (GILXM)

The **GMC Independent Lexium** library contains function blocks that can be used with Lexium drives.

These function blocks can be used with Lexium 32 drives:

Category	Function block	Description
Single axis	Jog_LXM32 *(see page 47)*	Operating mode: Jog
	SetTorqueRamp_LXM32 *(see page 59)*	Operating mode: Profile Torque
	TorqueControl_LXM32 *(see page 61)*	
	MoveVelocity_LXM32 *(see page 69)*	Operating mode: Profile Velocity
	Home_LXM32 *(see page 98)*	Operating mode: Homing
	SetStopRamp_LXM32 *(see page 112)*	Stopping
	Stop_LXM32 *(see page 116)*	
	Halt_LXM32 *(see page 122)*	
	TouchProbe_LXM32 *(see page 128)*	Position capture through signal input
Multi axis	GearInPos_LXM32 *(see page 139)*	Operating mode: Electronic Gear
	GearIn_LXM32 *(see page 142)*	
Administrative	SetDriveRamp_LXM32 *(see page 171)*	Writing a parameter
	SetLimitSwitch_LXM32 *(see page 175)*	
	ResetParameters_LXM32 *(see page 184)*	
	StoreParameters_LXM32 *(see page 190)*	
	ReadAxisWarning_LXM32 *(see page 208)*	Error handling

These function blocks can be used with Lexium ILA, ILE and ILS integrated drives:

Category	Function block	Description
Single axis	Jog_ILX *(see page 51)*	Operating mode: Jog
	Home_ILX *(see page 102)*	Operating mode: Homing
	SetStopRamp_ILX *(see page 114)*	Stopping
	TouchProbe_ILX *(see page 131)*	Position capture through signal input
Administrative	SetDriveRamp_ILX *(see page 173)*	Writing a parameter
	SetLimitSwitch_ILX *(see page 177)*	
	ResetParameters_ILX *(see page 186)*	
	StoreParameters_ILX *(see page 192)*	
	ReadAxisWarning_ILX *(see page 210)*	Error handling

PLCopen State Diagram

The state diagram represents the axis in terms of PLCopen. At any given point in time, the axis is in exactly one state. If a function block is executed or an error is detected, this may cause a state transition. The function block `MC_ReadStatus` delivers the current status of the axis.

Note 1 An error has been detected. (Transition from any state).

Note 2 The input `Enable` of the function block `MC_Power` is set to FALSE and no error has been detected (transition from any state).

Note 3 `MC_Reset` and `MC_Power.Status` = FALSE.

Note 4 `MC_Reset` and `MC_Power.Status` = TRUE and `MC_Power.Enable` = TRUE.

Note 5 `MC_Power.Enable` = TRUE and `MC_Power.Status` = TRUE.

Note 6 `MC_Stop.Done` = TRUE and `MC_Stop.Execute` = FALSE.

Transitions Between Function Blocks

This table presents how the execution of a function block (function block 1) can be terminated by another function block (function block 2).

Function block 1	Function block 2				
	MC_Jog	MC_Home	MC_MoveAbsolute	MC_MoveAdditive	MC_MoveRelative
MC_Jog	Immediately	Not permitted	Immediately	Immediately	Immediately
MC_Home	Not permitted	Not permitted	Not permitted	Not permitted	Not permitted
MC_MoveAbsolute	Motor standstill	Not permitted	Immediately	Immediately	Immediately
MC_MoveAdditive	Motor standstill	Not permitted	Immediately	Immediately	Immediately
MC_MoveRelative	Motor standstill	Not permitted	Immediately	Immediately	Immediately
MC_MoveVelocity	Motor standstill	Not permitted	Immediately	Immediately	Immediately
MC_TorqueControl	Motor standstill	Not permitted	Immediately	Immediately	Immediately
MC_Stop	Not permitted	Not permitted	Not permitted	Not permitted	Not permitted
MC_Halt	Motor standstill	Not permitted	Not permitted	Not permitted	Not permitted

Function block 1	Function block 2			
	MC_MoveVelocity	MC_TorqueControl	MC_Stop	MC_Halt
MC_Jog	Immediately	Immediately	Immediately	Immediately
MC_Home	Not permitted	Not permitted	Immediately	Not permitted
MC_MoveAbsolute	Immediately	Immediately	Immediately	Immediately
MC_MoveAdditive	Immediately	Immediately	Immediately	Immediately
MC_MoveRelative	Immediately	Immediately	Immediately	Immediately
MC_MoveVelocity	Immediately	Immediately	Immediately	Immediately
MC_TorqueControl	Immediately	Immediately	Immediately	Immediately
MC_Stop	Not permitted	Not permitted	Immediately	Not permitted
MC_Halt	Not permitted	Not permitted	Immediately	Immediately

Immediately

The execution of function block 2 is started on the fly, that is, without delay. The execution of function block 1 is aborted.

Function block 1 (`MC_MoveAbsolute`) starts at position 0	• `Position` = 100 • `Velocity` = 1000
Function block 2 (`MC_MoveVelocity`) starts at position 60	`Velocity` = 2000

Motor Standstill

The execution of function block 2 first decelerates the motor to a standstill with the adjusted deceleration ramp. The execution of function block 1 is aborted thereafter. The movement as per function block 2 starts as soon as the motor has come to a standstill.

Function block 1 (`MC_MoveVelocity`) starts at position 0	`Velocity` = 1000
Function block 2 (`MC_MoveAbsolute`) starts at position 100	• `Position` = 400 • `Velocity` = 2000

Not Permitted

Function block 1 cannot be aborted by the new function block. Function block 2 is not executed.

Library Diagnostic Codes

This table presents the diagnostic codes of the library.

Value (hex)	Description
00	No error detected.
01	Interface not implemented. The function block is not supported by the device.
02	Axis blocked. The axis is blocked by a different function block which cannot be interrupted.
03	Power disabled. The power stage must be enabled before the function block can be executed.
04	No re-execution. The function block cannot be repeated as long as the output `Busy` = TRUE.
0B	Parameter address invalid. The specified parameter address is not supported by the device.
0C	Input number out of range. The number entered for the signal input is outside of the permissible value range[1].
0D	Output number out of range. The number specified for the signal output is outside of the permissible value range[1].
0E	An internal error has been detected. Contact your local customer service office.
65	Done. Execution of the command terminated successfully.
66	Working. The command is still being executed.
67	Initialization unsuccessful. Initialization of the library not successful.
68	Configuring. The command was not executed, the device is not ready.
69	Indeterminable device state. Internal error detected. Contact your local customer service office.
6A	ID out of range. Internal error detected. Contact your local customer service office.
6B	Order not present. Internal error detected. Contact your local customer service office.
6C	Communication error detected. The connection to the device has been interrupted.
6D	Device state invalid. The function block cannot be executed in the current operating state of the device.

[1] See the respective function blocks documentation for permissible value range.

Value (hex)	Description
6E	Timeout. The command was not executed within the time window.
6F	Value out of range. The value is outside the permissible value range[1].
70	Parameter error detected. Error detected while reading or writing a parameter.
71	Indeterminable response. Indeterminable response while reading or writing a parameter.
72	Command error detected. Internal error detected. Contact your local customer service office.
73	Buffer full. Internal error detected. Contact your local customer service office.
74	The command was aborted.
75	Parameter not supported by device.
76	Touch probe number invalid. The specified value for the number of the `Touchprobe` input is invalid.
77	The specified edge of the `Touchprobe` input is invalid.
78	Touch probe limit error detected. The specified value for the number of the `Touchprobe` inputs is not supported by the library. Up to three `Touchprobe` inputs are supported.
79	Touch probe inactive. An attempt has been made to cancel an inactive Touchprobe.
7A	Touch probe active. An attempt has been made to execute an active `Touchprobe`.
7B	Attempt to reset an unrecoverable error. The detected error cannot be reset with `MC_Reset`. Restart the device after having remedied the cause of the detected error.
7C	Acceleration out of range. The value for the acceleration is outside of the permissible value range[1].
7D	Deceleration out of range. The value for the deceleration is outside of the permissible value range[1].
7E	Position out of range. The value for the target position is outside of the permissible value range[1].
7F	Velocity out of range. The value for the target velocity is outside of the permissible value range[1].
80	Torque out of range. The value for the target torque is outside of the permissible value range[1].
[1] See the respective function blocks documentation for permissible value range.	

Value (hex)	Description
81	Numerator out of range. The value for the numerator is outside of the permissible value range[1].
82	Denominator out of range. The value for the denominator is outside of the permissible value range[1].
83	External transition to NST. The command is not executed due to an external blocking event.
84	Halt active. The `Halt` function is active and the command is not executed.
85	Function block Control_ATV *(see page 78)* active. The function block cannot be executed as long as the function block Control_ATV *(see page 78)* is enabled.
86	Not ready for power-on. The power stage cannot be enabled in the current operating state of the drive.
87	Incorrect drive type. The function block does not support the linked Axis_Ref type.
88	Setpoint source invalid. Invalid value at the input `SetpointSource` of the function block TorqueControl_LXM32 *(see page 61)* or MoveVelocity_LXM32 *(see page 69)*.
[1] See the respective function blocks documentation for permissible value range.	

This table presents the communication error and diagnostic codes.

Value (hex)	Description
100	The exchange is valid.
101	The exchange stopped when the timeout expired.
102	The exchange was stopped by a user request (the `Abort` command).
103	The address format is incorrect.
104	The address of the remote device is incorrect.
105	The management table format is incorrect.
106	Specific parameters are incorrect.
107	An error occurred while sending the request to the destination.
109	The reception buffer size is insufficient.
10A	The transmission buffer size is insufficient.
10B	A system resource is unavailable.
10C	The transaction number is incorrect.
10E	The length is incorrect.
1FE	The operation error code contains a protocol-specific code.

Value (hex)	Description
1FF	The message was refused.
201	The target system resource is incommunicative.
205	The length is incorrect.
206	The communication channel is associated with an error detected.
207	The address is incorrect.
20B	A system resource is unavailable.
20C	A target communication function is not active.
20D	The target is incommunicative.
20F	The channel is not configured.
300	The exchange is valid.
301	The request has not been processed.
302	The received response is incorrect.
400	Service was successfully performed by the object specified.
401	A connection-related service is unsuccessful along the connection path.
402	Resources needed for the object to perform the requested service are unavailable.
403	Refer to Status Code 20 hex.
404	The path segment identifier or the segment syntax was not understood by the processing node. Path processing stops when this error is encountered.
405	The path is referencing an object class, instance, or structure element that is incorrect or is not contained in the processing node. Path processing stops when this error is encountered.
406	Only part of the expected data was transferred.
407	The messaging connection was lost.
408	The requested service was not implemented or was not defined for this Object Class/Instance.
409	Invalid attribute data.
40A	An attribute in the `Get_Attribute_List` or `Set_Attribute_List` response has a non-zero status.
40B	The object is already in the mode/state being requested by the service.
40C	The object cannot perform the requested service in its present mode/state.
40D	The requested instance of object to be created already exists.
40E	A request to modify a non-modifiable attribute was received.
40F	A permission/privilege check is unsuccessful.
410	The mode/state of the device does not allow the execution of the requested service.
411	The data to be transmitted in the response buffer is larger than the allocated response buffer.

Value (hex)	Description
412	The service specified an operation that is going to fragment a primitive data value, that is, half a REAL data type.
413	The service did not supply enough data to perform the specified operation.
414	The attribute specified in the request is not supported.
415	The service supplied more data than was expected.
416	The object specified does not exist in the device.
417	The fragmentation sequence for this service is not currently active for this data.
418	The attribute data of this object was not saved prior to the requested service.
419	The attribute data of this object was not saved.
41A	The service request packet was too large for transmission on a network in the path to the destination. The routing device was forced to abort the service.
41B	The service response packet was too large for transmission on a network in the path from the destination. The routing device was forced to abort the service.
41C	The service did not supply an attribute in a list of attributes that was needed by the service to perform the requested behavior.
41D	The service is returning the list of attributes supplied with status information for those attributes that were invalid.
41E	An embedded service resulted in an error.
41F	A vendor-specific error has been encountered. The **Additional Code** field of the error response defines the particular error encountered. Use of this general diagnostic code should only be performed when none of the diagnostic codes presented in this table or within an Object Class definition accurately reflect the error.
420	A parameter associated with the request was invalid. This code is used when a parameter does not meet the requirements of this specification and/or the requirements defined in an Application Object specification.
421	An attempt was made to write to a write-once medium (for example, WORM drive, PROM) that has already been written, or to modify a value that cannot be changed once established.
422	An invalid reply is received (for example, reply service code does not match the request service code, or reply message is shorter than the minimum expected reply size). This status code can serve for other causes of invalid replies.
423	The message received is larger than the receiving buffer can handle. The entire message is discarded.
424	The format of the received message is not supported by the server.
425	The key segment that is included as the first segment in the path does not match the destination module. The object-specific status indicates in which segment of the key.
426	The size of the path which was sent with the Service Request is either not large enough to allow the Request to be routed to an object or too much routing data was included.
427	An attempt was made to set an attribute that is not able to be set at this time.

Value (hex)	Description
428	The member ID specified in the request does not exist in the specified Class/Instance/Attribute.
429	A request to modify a non-modifiable member was received.
42A	This diagnostic code may only be reported by Group 2. Only servers with 4 K or less code space and only in place of service not supported, attribute not supported and attribute not settable.
42B	A CIP to Modbus translator received an invalid Modbus exception code.
42C	A request to read a non-readable attribute was received.
42D	The requested object instance cannot be deleted.
42E	The object supports the service, but not for the designated application path (for example, attribute). NOTE: Not to be used for any set service (use General Status code 0E hex or 29 hex instead).
4FF	No response from the target.

Section 1.2
Common Inputs and Outputs

What Is in This Section?

This section contains the following topics:

Topic	Page
Behavior of Function Blocks with the Input `Enable`	34
Behavior of Function Blocks with the Input `Execute`	37

Behavior of Function Blocks with the Input `Enable`

Example 1

Single execution without an error detected (execution requires more than one call).

Example 2

Single execution with an error detected (execution requires more than one call).

Example 3

Single execution without an error detected (execution requires only one call).

Example 4

Single execution with an error detected (execution requires only one call).

Example 5

Repeated execution without an error detected (execution requires more than one call).

Example 6

Repeated execution with an error detected (execution requires more than one call).

Example 7

Repeated execution without an error detected (execution requires only one cycle).

Example 8

Repeated execution with an error detected (execution requires only one call).

Behavior of Function Blocks with the Input `Execute`

Example 1

Execution terminated without an error detected.

Example 2

Execution terminated with an error detected.

Example 3

Execution aborted because another motion function block has been started.

Example 4

If the input `Execute` is set to FALSE during a cycle, the function block execution is not terminated; the output `Done` is set to TRUE only for one cycle.

Chapter 2
Function Blocks - Single Axis

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
2.1	Initialization	40
2.2	Operating Mode Jog	43
2.3	Operating Mode Profile Torque	55
2.4	Operating Mode Profile Velocity	64
2.5	Operating Mode Profile Position	84
2.6	Operating Mode Homing	94
2.7	Stopping	108
2.8	Position Capture via Signal Input	124

Section 2.1
Initialization

MC_Power

Functional Description

This function block enables or disables the drive power stage.

TRUE at the input `Enable` enables the power stage. Once the power stage is enabled, the output `Status` is set.

FALSE at the input `Enable` disables the power stage. Once the power stage is disabled, the output `Status` is reset.

If errors are detected during execution, the output `Error` is set.

The function block must not be used like a general `Enable` function block. Every time the function block is called the input `Enable` is compared with the output status. If these values are different a new command is executed, either to switch on the power stage (`Enable` = TRUE and `Status` = FALSE) or to switch off the power stage (`Enable` = FALSE and `Status` = TRUE). The function has to be called as long as the commanded state of the power stage is achieved or until an error occurs. If a function block error (for example, time-out) occurred, the `Error` output is set and will be reset again with the next call of the function block.

The function block should not be called cyclically. Call this function block only if it is required to switch off or switch on the power stage.

Library Name and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Status`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Power stage is disabled. ● TRUE: Power stage is enabled.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

If a Node Guarding or Heartbeat error is detected, the error memory must be reset by using the function block `MC_Reset` before the power stage can be enabled again.

Additional Information

PLCopen State Diagram *(see page 23)*

Initialization *(see page 40)*

Section 2.2
Operating Mode Jog

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_Jog	44
Jog_LXM32	47
Jog_ILX	51

MC_Jog

Functional Description

This function block starts the operating mode Jog.

In the operating mode Jog, a movement is started via the inputs `Forward` and `Backward`.

If the inputs `Forward` and `Backward` are set to FALSE, the operating mode is terminated and the output `Done` is set to TRUE.

If the inputs `Forward` and `Backward` are set to TRUE, the operating mode remains active, the jog movement is stopped and the output `Busy` remains set to TRUE.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Forward`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● `Forward` = FALSE and `Backward` = FALSE: Movement is terminated. ● `Forward` = TRUE and `Backward` = FALSE: Movement in positive direction is started. ● `Forward` = FALSE and `Backward` = TRUE: Movement in negative direction is started. ● `Forward` = TRUE and `Backward` = TRUE: The operating mode remains active, the jog movement is stopped, and the output `Busy` remains set to TRUE.
`Backward`	BOOL	

Input	Data type	Description
`Velocity`	DINT	Value range: 1...2147483647 Default value: 0. Velocity in user-defined units. **NOTE:** For Altivar, the values for `LowFrequency` and `HighFrequency` are set in the function block `SetFrequencyRange_ATV`. If the value for the target velocity `Velocity` is less than the value for `LowFrequency`, the movement is made with the velocity value for `LowFrequency`. No error is detected. If the value for the target velocity `Velocity` is greater than the value for `HighFrequency`, the movement is made with the velocity value for `HighFrequency`. No error is detected.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

Transitions Between Function Blocks *(see page 24)*

Operating Mode Jog *(see page 43)*

Jog_LXM32

Functional Description

This function block starts the operating mode Jog.

In the operating mode Jog, a movement is started via the inputs `Forward` and `Backward`.

If the inputs `Forward` and `Backward` are set to FALSE, the operating mode is terminated and the output `Done` is set to TRUE.

If the inputs `Forward` and `Backward` are set to TRUE, the operating mode remains active, the jog movement is stopped and the output `Busy` remains set to TRUE.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Forward`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● `Forward` = FALSE and `Backward` = FALSE: Movement is terminated. ● `Forward` = TRUE and `Backward` = FALSE: Movement in positive direction is started. ● `Forward` = FALSE and `Backward` = TRUE: Movement in negative direction is started. ● `Forward` = TRUE and `Backward` = TRUE: The operating mode remains active, the jog movement is stopped, and the output Busy remains set to TRUE.
`Backward`	BOOL	
`Fast`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The velocity can be modified during the movement. ● FALSE: Movement at the velocity set in `VeloSlow`. ● TRUE: Movement at the velocity set in `VeloFast`.
`TipPos`	DINT	Value range: 0...2147483647 Default value: 20 ● 0: Continuous movement is started immediately. ● >0: Value is used for distance to be moved in user-defined units. The movement is stopped, the waiting time `WaitTime` starts. After the waiting time `WaitTime` has elapsed, a continuous movement is started.
`WaitTime`	UINT	Value range: 0...65535 Default value: 500 Waiting time in ms. If `TipPos` is >0, the waiting time `WaitTime` starts as soon as the adjusted distance has been covered. After the waiting time `WaitTime` has elapsed, a continuous movement is started.
`VeloSlow`	DINT	Value range: 1...2147483647 Default value: 60 Velocity in user-defined units. If `Fast` = FALSE, the movement is made at this velocity.
`VeloFast`	DINT	Value range: 1...2147483647 Default value: 180 Velocity in user-defined units. If `Fast` = TRUE, the movement is made at this velocity.
`Acceleration`	DINT	Value range: 1...2147483647 Default value: 600 Acceleration ramp in user-defined units.

Input	Data type	Description
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 600 Deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Operating Mode Jog *(see page 43)*

Jog_ILX

Functional Description

This function block starts the operating mode Jog.

In the operating mode Jog, a movement is started via the inputs `Forward` and `Backward`.

If the inputs `Forward` and `Backward` are set to FALSE, the operating mode is terminated and the output `Done` is set to TRUE.

If the inputs `Forward` and `Backward` are set to TRUE, the operating mode remains active, the jog movement is stopped and the output `Busy` remains set to TRUE.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
Forward	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● Forward = FALSE and Backward = FALSE: Movement is terminated. ● Forward = TRUE and Backward = FALSE: Movement in positive direction is started. ● Forward = FALSE and Backward = TRUE: Movement in negative direction is started. ● Forward = TRUE and Backward = TRUE: The operating mode remains active, the jog movement is stopped, and the output Busy remains set to TRUE.
Backward	BOOL	
Fast	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The velocity can be modified during the movement. ● FALSE: Movement at the velocity set in VeloSlow. ● TRUE: Movement at the velocity set in VeloFast.
TipPos	DINT	Value range: 0...2147483647 Default value: 20 ● 0: Continuous movement is started immediately. ● >0: Value is used for distance to be moved in user-defined units. The movement is stopped, the waiting time WaitTime starts. After the waiting time WaitTime has elapsed, a continuous movement is started.
WaitTime	UINT	Value range: 0...65535 Default value: 500 Waiting time in the unit ms. If TipPos is >0, the waiting time WaitTime starts as soon as the adjusted distance has been covered. After the waiting time WaitTime has elapsed, a continuous movement is started.
VelSlow	DINT	Value range: 1...13200 Default value: 60 Velocity in user-defined units. If Fast = FALSE, the movement is made at this velocity.
VelFast	DINT	Value range: 1...13200 Default value: 180 Velocity in user-defined units. If Fast = TRUE, the movement is made at this velocity.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific acceleration and deceleration values for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default value for deceleration is 750 user-defined units.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function block `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

Operating Mode Jog *(see page 43)*

Section 2.3
Operating Mode Profile Torque

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_TorqueControl	56
SetTorqueRamp_LXM32	59
TorqueControl_LXM32	61

MC_TorqueControl

Functional Description

This function block starts the operating mode Profile Torque. In the operating mode Profile Torque, a movement is made with a target torque. The reference value for the target torque is supplied via the input `Torque`. When the target torque is reached, the output `InTorque` is set to TRUE.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Torque`	INT	Value range: -30000...30000 Default value: 0 Target torque in user-defined units. The value corresponds to 0.1% of the nominal torque of the motor. Example: `Torque` = 300 corresponds to 30% of the nominal torque of the motor. Refer to the drive documentation for an overview of the parameters.

Outputs

Output	Data type	Description
`InTorque`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Target torque not reached. ● TRUE: Target torque reached.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed. **NOTE:** The output `Busy` remains TRUE even when the target velocity has been reached or `Execute` becomes FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` is executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Operating Mode Profile Torque *(see page 55)*

SetTorqueRamp_LXM32

Functional Description

This function block sets the torque ramp for `MC_TorqueControl` *(see page 56)*.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Ramp`	DINT	Value range: 1...30000000 Default value: 100000 The input `Ramp` is used in the operating mode Profile Torque. The value corresponds to 0.1% per second of the nominal torque of the motor. Example: If `Ramp` = 1000, then 100% of the nominal torque of the motor is reached in one second.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Operating Mode Profile Torque *(see page 55)*

TorqueControl_LXM32

Functional Description

This function block starts the operating mode Profile Torque. In the operating mode Profile Torque, a movement is made with a target torque. The source for the target torque is set via the input `SetpointSource`. When the target torque is reached, the output `InTorque` is set to TRUE.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`SetpointSource`	UINT	Value range: 0...2 Default value: 0 Source of the target torque. ● 0: Target torque via input `Torque` ● 1: Target torque via analog input (I/O module) ● 2: Target torque via PTI interface

Input	Data type	Description
`Torque`	INT	Value range: -30000...30000 Default value: 0 Target torque in user-defined units. The value corresponds to 0.1% of the nominal torque of the motor. Example: `Torque` = 300 corresponds to 30% of the nominal torque of the motor. Refer to the drive documentation for an overview of the parameters.
`TorqueRamp`	DINT	Value range: 1...30000000 Default value: 100000 The value corresponds to 0.1% per second of the nominal torque of the motor. Example: If `Ramp` = 1000, then 100% of the nominal torque of the motor is reached in one second.

Outputs

Output	Data type	Description
`InTorque`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Target torque not reached. ● TRUE: Target torque reached.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed. **NOTE:** The output `Busy` remains TRUE even when the target velocity has been reached or `Execute` becomes FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` is executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.

Output	Data type	Description
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Operating Mode Profile Torque *(see page 55)*

Section 2.4
Operating Mode Profile Velocity

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_MoveVelocity	65
MoveVelocity_LXM32	69
VelocityControlAnalogInput_ATV	72
VelocityControlSelectAI_ATV	75
Control_ATV	78

MC_MoveVelocity

Functional Description

This function block starts the operating mode Profile Velocity. In the operating mode Profile Velocity, a movement is made with a target velocity. The reference value for the target velocity is supplied via the input `Velocity`. When the target velocity is reached, the output `InVelocity` is set to TRUE.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Input	Data type	Description
`Velocity`	DINT	Value range: -2147483648...2147483647 Default value: 0 Target velocity in user-defined units. **NOTE:** For Altivar, the values for `LowFrequency` and `HighFrequency` are set in the function block `SetFrequencyRange_ATV`. If the value for the target velocity `Velocity` is less than the value for `LowFrequency`, the movement is made with the velocity value for `LowFrequency`. No error is detected. If the value for the target velocity `Velocity` is greater than the value for `HighFrequency`, the movement is made with the velocity value for `HighFrequency`. No error is detected.

Outputs

Output	Data type	Description
`InVelocity`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Target value not reached. ● TRUE: Target value reached.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed. **NOTE:** The output `Busy` remains TRUE even when the target velocity has been reached or `Execute` becomes FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` is executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.

Output	Data type	Description
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

The output `Busy` remains TRUE even if the target velocity has been reached or the input `Execute` is set to FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` *(see page 109)* is executed.

In the operating mode Profile Velocity, a movement beyond the movement range is possible. In the case of a movement beyond the movement range, the reference point established by homing becomes invalid.

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Operating Mode Profile Velocity *(see page 64)*

MoveVelocity_LXM32

Functional Description

This function block starts the operating mode Profile Velocity. In the operating mode Profile Velocity, a movement is made with a target velocity. The source for the target velocity is set via the input `SetpointSource`. When the target velocity is reached, the output `InVelocity` is set to TRUE.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`SetpointSource`	UINT	Value range: 0...1 Default value: 0 Source of the target velocity. ● 0: Target velocity via input `Velocity` ● 1: Target velocity via analog input (I/O module)

Input	Data type	Description
`Velocity`	DINT	Value range: -2147483648...2147483647 Default value: 0 Target velocity in user-defined units.
`Acceleration`	DINT	Value range: 1...2147483647 Default value: 600 Acceleration ramp in user-defined units.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 600 Deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`InVelocity`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Target value not reached. ● TRUE: Target value reached.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed. **NOTE:** The output `Busy` remains TRUE even when the target velocity has been reached or `Execute` becomes FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` is executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The output `Busy` remains TRUE even if the target velocity has been reached or the input `Execute` is set to FALSE. The output `Busy` is set to FALSE as soon as another function block such as `MC_Stop` *(see page 109)* is executed.

Additional Information

Operating Mode Profile Velocity *(see page 64)*

VelocityControlAnalogInput_ATV

Functional Description

This function block uses the reference values supplied by the analog input selected with the function block `VelocityControlSelectAI_ATV` *(see page 75)*.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`NegativeDir`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Clockwise rotation. ● TRUE: Counter-clockwise rotation.

Outputs

Output	Data type	Description
`InVelocity`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: The velocity does not correspond to the reference value. ● TRUE: The velocity corresponds to the reference value.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

More information is provided in the following function blocks description: `SetFrequencyRange_ATV` *(see page 179)* and `VelocityControlSelectAI_ATV` *(see page 75)*. If analog voltage levels –10 V...+10 V are used, the direction of movement (rotation) is inversed when the sign changes. If the voltage is 0 V, this may result in jumps in the direction of movement, in the minimum frequency and in jumps at standstill.

Additional Information

Operating Mode Profile Velocity *(see page 64)*

VelocityControlSelectAI_ATV

Functional Description

This function block selects the analog input for supplying the reference value. See also `VelocityControlAnalogInput_ATV` *(see page 72)*.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Input	Data type	Description
`InputNumber`	UINT	Value range: 1...4 Default value: 1 ATV32/ATV320: ● 1: AI1 ● 2: AI2 ● 3: AI3 ATV340: ● 1: AI1 ● 2: AI2 ATV71: ● 1: AI1 ● 2: AI2 ● 3: AI3 (with expansion card) ● 4: AI4 (with expansion card)

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

This function block can only be executed if the drive is in the operating state 3 Switch On Disabled. To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Additional Information

Operating Mode Profile Velocity *(see page 64)*

Control_ATV

Functional Description

This function block manages the Controlword, Statusword, reference velocity and the direction of movement for the drive.

This function block requires an Adaptation of the I/O Mapping *(see page 81)*.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data Type	Description
i_xEn	BOOL	Value range: FALSE, TRUE. Default value: FALSE. Command for activating or deactivating the function block. ● FALSE: Deactivate the function block ● TRUE: Activate the function block
i_xKeepOpEn	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Power stage is disabled if no command is active. ● TRUE: Power stage remains enabled if no command is active.

Input	Data Type	Description
`i_xFwd`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Stops a movement in positive direction. ● TRUE: If the drive is in the operating state "Switched On" and if there is no local forcing active, a movement is started in negative direction (Reverse) with the velocity reference value `i_wSpdRef`. The command "Reverse" is triggered with a rising edge. The movement stops when the level is FALSE.
`i_xRev`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Stops a movement in negative direction. ● TRUE: If the drive is in the operating state "Switched On" and if there is no local forcing active, a movement is started in positive direction (Forward) with the velocity reference value `i_wSpdRef`. The command "Forward" is triggered with a rising edge. The movement stops when the level is FALSE.
`i_xQckStop`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: If there is a motor movement, the drive triggers a "Quick Stop". ● TRUE: No triggering of a "Quick Stop". After a Quick Stop, the drive automatically switches to the operating state "Switched On" when the actual velocity and the actual current values have reached a value of zero and if the commands Forward and Reverse are both FALSE. The Quick Stop must be deactivated (set `i_xQckStop` to TRUE) to restart the movement.
`i_xFreeWhl`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: If there is a motor movement, the drive triggers a "Free Wheel Stop". ● TRUE: No triggering of a "Free Wheel Stop".
`i_xFltRst`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: No triggering of a "Fault Reset". ● TRUE: The drive triggers a "Fault Reset".
`i_wSpdRef`	WORD	Value range: Default value: 0 Reference velocity for the drive.

Outputs

Output	Data Type	Description
`q_xEn`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. Function block activated/deactivated. Direct copy from `i_xEn`.
`q_xAlrm`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. Is set to FALSE when the function block is deactivated and when the drive transitions to operating state "Switch On Disabled". Is set to TRUE when the drive detects an error (bit 3 of the status word).

Inputs/Outputs

Input/Output	Data Type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of other function blocks of the library leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only activate this function block when all of the following function blocks are inactive:
  - MC_Power
  - MC_Jog
  - MC_MoveVelocity
  - VelocityControlAnalogInput_ATV
  - VelocityControlSelectAI_ATV
  - MC_Stop
  - MC_Reset
- Deactivate this function block before activating any of the function blocks listed above.

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

Note the following:

After a "Quick Stop", the operating state "Quick Stop Active" is automatically left when the actual velocity and the actual current values have reached a value of zero and if the commands “Forward” and “Reverse” are both FALSE. To restart the movement, deactivate the Quick Stop (set `i_xQckStop` to TRUE).

A "Quick Stop" has a higher priority than a regular stop ("Forward" and "Reverse" set to FALSE).

A "Free Wheel Stop" has a higher priority than a "Quick Stop".

If the drive displays the flashing message `COF` on the 7-segment display after a download of an application to the drive, a rising edge and then a falling edge are required at the “Fault Reset” input (`i_xFltRst`) to restart proper communication with the drive.

Adaptation of the I/O Mapping (EtherNet/IP and Modbus/TCP)

If you are using EtherNet/IP or Modbus/TCP, you need to adjust the I/O mapping of the drive in order to use this function block.

The I/O mapping of the drive can only be adjusted with the DTM commissioning software. See Device Type Manager - User Guide for additional information on the DTM.

For EtherNet/IP, the library uses the assemblies 100 and 101 and requires the following mapping:

- Assembly 100 (controller to drive):
  - First word: CMD, logic address 8501 (factory setting)
  - Second word: LFRD, logic address 8602 (factory setting)
- Assembly 101 (drive to controller):
  - First word: ETA, logic address 3201 (factory setting)
  - Second word: RFRD, logic address 8604 (factory setting)
  - Third word: LCR, logic address 3204 (needs to be added to the mapping)

For Modbus/TCP, the library uses the I/O scanning service and requires the following mapping:

- I/O scanner output setting (controller to drive):
  - Output 1: CMD, logic address 8501 (factory setting)
  - Output 2: LFRD, logic address 8602 (factory setting)
- I/O scanner input setting (drive to controller):
  - Input 1: ETA, logic address 3201 (factory setting)
  - Input 2: RFRD, logic address 8604 (factory setting)
  - Input 3: LCR, logic address 3204 (needs to be added to the mapping)

Using the Function Block

Starting the function block with the default settings:

Step	Action
1	Deactivate "Free Wheel": Set `i_xFreeWhl` to TRUE.
2	Deactivate "Quick Stop": Set `i_xQckStop` to TRUE.
3	Activate the function block: Set `i_xEn` to TRUE.
4	Set a reference velocity: Set `i_wSpdRef` to a value not equal to zero.
5	Start a movement in positive ("Forward") or negative ("Reverse") direction: Set `i_xFwd` or `i_xRev` to TRUE.

Visualization

Visualization of function block Control_ATV:

See **Programming with SoMachine → Visualization** for additional information on the visualization of a function block.

With the above minimum configuration, the visualization of this function block can be used to control the drive. After the I/O mapping of the 5 data specified above, the drive can be started with the following sequence of steps:

Step	Action
1	Click the button **Enable** to activate the function block.
2	Click the button **Quick Stop** to deactivate "Quick Stop".
3	Click the button **Free Wheel** to deactivate "Free Wheel".
4	Enter a velocity value not equal to zero in revolutions per minute (in the field next to the **Force Speed** button).
5	Click the button **Force Speed**.
6	Click the button **Forward** or **Reverse**: The motor performs a movement in positive or negative direction.

Additional Information

Operating Mode Profile Velocity *(see page 64)*

Section 2.5
Operating Mode Profile Position

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_MoveAbsolute	85
MC_MoveAdditive	88
MC_MoveRelative	91

MC_MoveAbsolute

Functional Description

This function block starts a movement to the absolute target position `Position`.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0. Target position absolute in user-defined units.
`Velocity`	DINT	Value range: -2147483648...2147483647 Default value: 0 Target velocity in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

Absolute positioning requires a valid zero point. You can use the function block `MC_ReadMotionState` *(see page 156)* to check for a valid zero point.

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Operating Mode Profile Position *(see page 84)*

MC_MoveAdditive

Functional Description

This function block starts a movement to the original target position including distance `Distance`.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Distance`	DINT	Value range: -2147483648...2147483647 Default value: 0. Target position relative to the previous target position in user-defined units.
`Velocity`	DINT	Value range: -2147483648...2147483647 Default value: 0 Target velocity in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Operating Mode Profile Position *(see page 84)*

MC_MoveRelative

Functional Description

This function block starts a movement by distance `Distance` to the actual position.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Distance`	DINT	Value range: -2147483648...2147483647 Default value: 0. Target position relative with reference to actual position.
`Velocity`	DINT	Value range: -2147483648...2147483647 Default value: 0 Target velocity in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Operating Mode Profile Position *(see page 84)*

Section 2.6
Operating Mode Homing

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_Home	95
Home_LXM32	98
Home_ILX	102
MC_SetPosition	106

MC_Home

Functional Description

This function block configures and starts a reference movement.

The MC_Home function block commands the homing procedure. Drive-specific homing parameters like homing mode are provided by the device implementation. Re-executing this function block commands a Quick-Stop in the drive and leads to a the function block error NoReExecution. The homing procedure can be aborted only by executing MC_Stop or disabling the power stage via MC_Power.

Executing another motion function block while MC_Home is busy does not influence the homing procedure. This means that the function block MC_Home stays busy and the executed function block ends in a function block error.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
Execute	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input Execute starts the function block. The function block continues execution and the output Busy is set to TRUE. A rising edge at the input Execute is not permitted while the function block is being executed.

Input	Data type	Description
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0. Position at reference point in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific acceleration and deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transitions Between Function Blocks *(see page 24)*

Operating Mode Homing *(see page 94)*

Home_LXM32

Functional Description

This function block configures and starts a reference movement.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is not permitted while the function block is being executed.
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0 Position in user-defined units. `HomingMode` 1...34: Position at reference point `HomingMode` 35: Position for position setting

Input	Data type	Description
`HomingMode`	UINT	Value range: 1...35 Default value: 17 Specifies the homing method ● 1: LIMN with index pulse ● 2: LIMP with index pulse ● 7: REF+ with index pulse, inverted, outside ● 8: REF+ with index pulse, inverted, inside ● 9: REF+ with index pulse, not inverted, inside ● 10: REF+ with index pulse, not inverted, outside ● 11: REF- with index pulse, inverted, outside ● 12: REF- with index pulse, inverted, inside ● 13: REF- with index pulse, not inverted, inside ● 14: REF- with index pulse, not inverted, outside ● 17: LIMN ● 18: LIMP ● 23: REF+, inverted, outside ● 24: REF+, inverted, inside ● 25: REF+, not inverted, inside ● 26: REF+, not inverted, outside ● 27: REF-, inverted, outside ● 28: REF-, inverted, inside ● 29: REF-, not inverted, inside ● 30: REF-, not inverted, outside ● 33: Index pulse in negative direction ● 34: Index pulse in positive direction ● 35: Position setting Refer to drive documentation *(see page 11)* for a description of the homing method.
`VHome`	DINT	Value range: 1...2147483647 Default value: 60 Target velocity for searching the switch in user-defined units.
`VOutHome`	DINT	Value range: 1...2147483647 Default value: 6 Target velocity for moving away from switch in user-defined units.

Input	Data type	Description
POutHome	DINT	Value range: 0...2147483647 Default value: 0 Maximum distance for searching the switching point in user-defined units. ● 0: Search distance monitoring disabled ● >0: Maximum distance After detection of the switch, the drive starts to search for the defined switching point. If the defined switching point is not found within the distance defined here, an error is detected and the reference movement is canceled.
PDisHome	DINT	Value range: 0...2147483647 Default value: 200 Maximum search distance after overtravel of switch in user-defined units. ● 0: Search distance monitoring disabled ● >0: Search distance The switch must be activated again within this search distance; otherwise the reference movement is canceled.
Acceleration	DINT	Value range: 1...2147483647 Default value: 600 Acceleration ramp in user-defined units.
Deceleration	DINT	Value range: 1...2147483647 Default value: 600 Deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
Done	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
Busy	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
CommandAborted	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.

Output	Data type	Description
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Operating Mode Homing *(see page 94)*

Home_ILX

Functional Description

This function block commands the homing procedure using the Lexium ILA, ILE and ILS integrated drives specific homing method set via the input `HomingMode`. Drive-specific homing parameters such as homing mode are provided by the device implementation. Re-executing this function block commands a Quick-Stop in the drive and leads to a function block error `NoReExecution` .

The homing procedure can be aborted only by executing `MC_Stop` or disabling the power stage through `MC_Power`.

Executing another motion function block while `MC_Home` is `busy` does not influence the homing procedure. This means that the function block `MC_Home` stays `busy` and the executed function block ends in a function block error.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is not permitted while the function block is being executed.
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0 Position in user-defined units: ● 1...34: Position at reference point ● 35: Position for position setting
`HomingMode`	UINT	Value range: 1...35 Default value: 18 Specifies the homing method ● 1: LIMN with index pulse ● 2: LIMP with index pulse ● 7: REF+ with index pulse, inverted, outside ● 8: REF+ with index pulse, inverted, inside ● 9: REF+ with index pulse, not inverted, inside ● 10: REF+ with index pulse, not inverted, outside ● 11: REF- with index pulse, inverted, outside ● 12: REF- with index pulse, inverted, inside ● 13: REF- with index pulse, not inverted, inside ● 14: REF- with index pulse, not inverted, outside ● 17: LIMN ● 18: LIMP ● 23: REF+, inverted, outside ● 24: REF+, inverted, inside ● 25: REF+, not inverted, inside ● 26: REF+, not inverted, outside ● 27: REF-, inverted, outside ● 28: REF-, inverted, inside ● 29: REF-, not inverted, inside ● 30: REF-, not inverted, outside ● 33: Index pulse in negative direction ● 34: Index pulse in positive direction ● 35: Position setting Refer to drive documentation *(see page 11)* for a description of the homing method.

Input	Data type	Description
VHome	DINT	Value range: 1...13200 Default value: 60 Target velocity for searching the switch in user-defined units.
VOutHome	DINT	Value range: 1...13200 Default value: 6 Target velocity for searching the switch in user-defined units.
POutHome	DINT	Value range: 0...2147483647 Default value: 0 Maximum distance for search for switching point in user-defined units. ● 0: Search distance monitoring disabled ● >0: Maximum distance After detection of the switch, the drive starts to search for the defined switching point. If the defined switching point is not found within the distance defined here, an error is detected and the reference movement is canceled.
PDisHome	DINT	Value range: 0...2147483647 Default value: 200 Maximum search distance after overtravel of switch in user-defined units. ● 0: Search distance monitoring disabled ● >0: Search distance The switch must be activated again within this search distance; otherwise the reference movement is canceled.

Outputs

Output	Data type	Description
Done	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
Busy	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
CommandAborted	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.

Output	Data type	Description
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific acceleration and deceleration values for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default acceleration and deceleration values written by this function block are as follows:

- The default value for acceleration is 600 user-defined units.
- The default value for deceleration is 750 user-defined units.

If you require other acceleration and/or deceleration values, you must use the vendor-specific function blocks to do so. Use the function block `SetDriveRamp_ILX` *(see page 173)* to define the acceleration and deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

Operating Mode Homing *(see page 94)*

MC_SetPosition

Functional Description

This function block sets a position value at the actual position of the motor. The zero point is defined by the position value. The function block can only be used when the motor is at a standstill.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0 Position in user-defined units. Value for position setting.
`Relative`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: The actual position is set to the value of the input `Position`. ● TRUE: The value of the `Position` is added to the actual position.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Operating Mode Homing *(see page 94)*

Section 2.7
Stopping

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_Stop	109
SetStopRamp_LXM32	112
SetStopRamp_ILX	114
Stop_LXM32	116
MC_Halt	119
Halt_LXM32	122

MC_Stop

Functional Description

This function block stops the ongoing movement. The operating mode is aborted by this function block.

The function block `MC_Stop` commands a motion stop to the drive. The drive-specific stop parameters like deceleration are provided by the device implementation. Re-executing this function block does not influence the ongoing deceleration.

The stop procedure can be aborted only by disabling the power stage via `MC_Power`.

Executing another motion function block while `MC_Stop` is busy does not influence the stop procedure. This means that the function block `MC_Stop` stays busy and the executed function block ends in a function block error.

As long as the input `Execute` is `TRUE` the execution of a motion command is not possible. In this case, executed motion function blocks end in a function block error.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

The function block can only be interrupted by disabling the power stage via the function block `MC_Power` *(see page 40)*.

As long as the input `Execute` is TRUE, no other function block except for `MC_Power` *(see page 40)* can be started.

For ATV the stop method is according to the stop configuration (see product manual).

To setup the deceleration values for LXM32M, Lexium ILA, ILE and ILS integrated drives, you must use the vendor-specific function blocks. Use the function blocks `SetStopRamp_LXM32` *(see page 112)* and `SetStopRamp_ILX` *(see page 114)* to define the deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Stopping *(see page 108)*

SetStopRamp_LXM32

Functional Description

This function block sets the deceleration ramp for the function block `MC_Stop` *(see page 109)*.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 6000 Value for the deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Stopping *(see page 108)*

SetStopRamp_ILX

Functional Description

This function block writes the value for the quick stop deceleration of the drive. This value is used if the function block `MC_Stop` is executed.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Deceleration`	DINT	Value range: 200...3000000 Default value: 6000 Value for the deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Stopping *(see page 108)*

Stop_LXM32

Functional Description

This function block stops the ongoing movement. The operating mode is aborted by this function block.

The execution of the function block cannot be aborted by another function block call.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 6000 Value for the deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The function block can only be interrupted by disabling the power stage through the function block `MC_Power` *(see page 40)*.

As long as the input `Execute` is TRUE, no other function block except for `MC_Power` *(see page 40)* can be started.

Additional Information

Stopping *(see page 108)*

MC_Halt

Functional Description

This function block stops the ongoing movement. The operating mode is aborted by this function block.

The execution of the function block can be aborted by another function block. If a Halt is triggered, there is a transition of the PLCopen state to Discrete Motion until the motor has reached a standstill. Once the motor has reached a standstill, the output `Done` is set and the state transitions to StandStill.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This function block uses library-specific deceleration values for LXM32M (EtherNet/IP and Modbus/TCP) and for Lexium ILA, ILE and ILS integrated drives (EtherNet/IP only). This means that pre-configured values for these parameters (for example, via the commissioning tool) are overwritten when this function block is executed.

The default deceleration values written by this function block are as follows:

- The default values for deceleration are 600 user-defined units for LXM32M and 750 user-defined units for Lexium ILA, ILE and ILS integrated drives.

If you require other deceleration values, you must use the vendor-specific function blocks to do so. Use the function blocks `SetDriveRamp_LXM32` *(see page 171)* and `SetDriveRamp_ILX` *(see page 173)* to define the deceleration. The function block has to be executed only once if a change of the ramp values is required.

Additional Information

PLCopen State Diagram *(see page 23)*

Transition Between Function Blocks *(see page 24)*

Stopping *(see page 108)*

Halt_LXM32

Functional Description

This function block stops the ongoing movement. The operating mode is aborted by this function block.

The execution of the function block can be aborted by another function block. If a Halt is triggered, there is a transition of the PLCopen state to Discrete Motion until the motor has reached a standstill. Once the motor has reached a standstill, the output `Done` is set and the state transitions to StandStill.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 600 Value for the deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Stopping *(see page 108)*

Section 2.8
Position Capture via Signal Input

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_TouchProbe	125
TouchProbe_LXM32	128
TouchProbe_ILX	131
MC_AbortTrigger	134

MC_TouchProbe

Functional Description

This function block configures and starts position capture.

The function block returns the axis position at the occurrence of a trigger event. The drive trigger parameters are provided by the device implementation.

Executing the function block `MC_AbortTrigger` while `MC_TouchProbe` is busy aborts the function for the referenced trigger input.

A new rising edge at the input `Execute` overwrites and restarts the active trigger function.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`RecordedPosition`	DINT	Returns the value of the axis position at the occurrence of a trigger event in user-defined units. Value range: -2147483648...2147483647 Default value: 0 Captured motor position.

Inputs/Outputs

Input/Output	Data type	Elements	Data type	Description
`Axis`	Axis_Ref	–		Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.
`TriggerInput`	MC_Trigger_Ref	`TouchProbeNumber`	UINT	Selects the capture unit of the drive.
		`TriggerEdge`	ET_TriggerEdge	Edge to trigger position capture. ● 1 = Falling edge. ● 2 = Rising edge. ● 3 = Both falling edge and rising edge.

Notes

Use `MC_AbortTrigger` function block to abort `MC_TouchProbe` function block execution.

Additional Information

Position Capture via Signal Input *(see page 124)*

TouchProbe_LXM32

Functional Description

This function block configures and starts position capture.

The function block returns the axis position at the occurrence of a trigger event. The drive trigger parameters are provided by the device implementation.

Executing the function block `MC_AbortTrigger` while `MC_TouchProbe` is busy aborts the function for the referenced trigger input.

A new rising edge at the input `Execute` overwrites and restarts the active trigger function.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Input	Data type	Description
`SingleShot`	BOOL	Value range: FALSE, TRUE. Default value: TRUE. ● FALSE: Captures continuously. ● TRUE: Captures once.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Output	Data type	Description
`RecordedPosition`	DINT	Returns the value the axis position at the occurrence of a trigger event. Value range: -2147483648...2147483647 Default value: 0 Captured motor position in user-defined units.

Inputs/Outputs

Input/Output	Data type	Elements	Data type	Description
`Axis`	Axis_Ref	–		Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.
`TriggerInput`	MC_Trigger_Ref	`TouchProbeNumber`	UINT	Selects the capture unit of the drive.
		`TriggerEdge`	ET_TriggerEdge	Edge to trigger position capture. ● 1 = Falling edge. ● 2 = Rising edge. ● 3 = Falling edge or rising edge.

Notes

Use `MC_AbortTrigger` function block to abort `TouchProbe_LXM32` function block execution.

Additional Information

Position Capture via Signal Input *(see page 124)*

TouchProbe_ILX

Functional Description

This function block configures and starts position capture.

The function block returns the axis position at the occurrence of a trigger event. The drive trigger parameters are provided by the device implementation.

Executing the function block `MC_AbortTrigger` while `MC_TouchProbe` is busy aborts the function for the referenced trigger input.

A new rising edge at the input `Execute` overwrites and restarts the active trigger function.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.

Input	Data type	Description
`SingleShot`	BOOL	Value range: FALSE, TRUE. Default value: TRUE. Selects the trigger occurrence. ● FALSE = Captures continuously ● TRUE = Captures once

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Output	Data type	Description
RecordedPosition	DINT	Returns the value the axis position at the occurrence of a trigger event. Value range: -2147483648...2147483647 Default value: 0 Captured motor position in user-defined units.

Inputs/Outputs

Input/Output	Data type	Elements	Data type	Description
Axis	Axis_Ref	–		Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.
TriggerInput	MC_TriggerRef	TouchProbeNumber	UINT	Selects the capture unit of the drive.
		TriggerEdge	ET_TriggerEdge	Selects the trigger edge for capturing the position. ● 1 = Rising edge ● 2 = Falling edge ● 3 = Falling edge or rising edge.

Notes

Use MC_AbortTrigger function block to abort TouchProbe_ILX function block execution.

Additional Information

Position Capture via Signal Input *(see page 124)*

MC_AbortTrigger

Functional Description

This function block terminates a position capture.

Library and Namespace

Library name:**GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Elements	Data type	Description
`Axis`	Axis_Ref	–		Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.
`TriggerInput`	MC_Trigger_Ref	`TouchProbeNumber`	UINT	Selects the capture unit of the drive.
		`TriggerEdge`	ET_TriggerEdge	Specification of the edge to trigger position capture. Is used in the input `TriggerInput`. ● 1 = Falling edge. ● 2 = Rising edge. ● 3 = Falling edge or rising edge.

Additional Information

Position Capture via Signal Input *(see page 124)*

Chapter 3
Function Blocks - Multi Axis

Section 3.1
Operating Mode Electronic Gear

What Is in This Section?

This section contains the following topics:

Topic	Page
GearInPos_LXM32	139
GearIn_LXM32	142

GearInPos_LXM32

Functional Description

This function block starts the operating mode Electronic Gear with the method position synchronization.

In the operating mode Electronic Gear, movements are carried out synchronously according to externally supplied reference value signals. A position reference value is calculated on the basis of these external reference values plus an adjustable gear ratio. The reference value signals can be A/B signals, P/D signals or CW/CCW signals.

A movement can be made using one of two methods:

- Position synchronization without compensation movement
  Reference value signals supplied during an interruption caused by a halt command or by a detected error of error class 1 are not taken into account.
- Position synchronization with compensation movement
  Reference value signals supplied during an interruption caused by a halt command or by a detected error of error class 1 are taken into account and compensated for.

Refer to the drive documentation *(see page 11)* for additional information on compensation movements and error classes.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`SyncMode`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. Compensation movement: ● FALSE: With compensation movement ● TRUE: Without compensation movement
`RatioNumerator`	DINT	Value range: 1...2147489647 Default value: 1 Gear ratio: Numerator of gear ratio.
`RatioDenominator`	DINT	Value range: 1...2147489647 Default value: 1 Gear ratio: Denominator of gear ratio.

Outputs

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.

Output	Data type	Description
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`InGear`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: If the adjusted gear ratio is reached for the first time in the operating mode Electronic Gear.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This requires the drive parameter **GEARratio** to be 0 (Refer to the drive documentation *(see page 11)*). This way, `RatioNumerator` and `RatioDenominator` are used to calculate the gear ratio.

The velocity of the compensation movement is limited by the maximum current (drive parameter **CTRL_I_max**) and the maximum velocity of the motor.

The enabled direction of movement in the operating mode Electronic Gear is set via the drive parameter **GEARdir_enabl**.

Once the operating mode is active, the compensation movement must not exceed the maximum permissible position deviation. If the required compensation movement exceeds the maximum permissible position deviation, an error is detected.

The operating mode Electronic Gear with the method velocity synchronization is started with the function block GearIn_LXM32 *(see page 142)*.

Additional Information

Operating Mode Electronic Gear *(see page 138)*

GearIn_LXM32

Functional Description

This function block starts the operating mode Electronic Gear with the method velocity synchronization.

In the operating mode Electronic Gear, movements are carried out according to externally supplied reference value signals. A velocity reference value is calculated on the basis of these external reference values plus an adjustable gear ratio. The reference value signals can be A/B signals, P/D signals or CW/CCW signals.

The movement is made synchronously (velocity synchronicity) with the supplied reference value signals.

Refer to the drive documentation *(see page 11)* for additional information on error classes.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. This function block can be restarted while it is executed. The target values are overwritten by the new values at the point in time the rising edge occurs.
`RatioNumerator`	DINT	Value range: 1...2147489647 Default value: 1 **Gear ratio**: Numerator of gear ratio.
`RatioDenominator`	DINT	Value range: 1...2147489647 Default value: 1 **Gear ratio**: Denominator of gear ratio.
`Acceleration`	DINT	Value range: 1...2147483647 Default value: 600 Acceleration ramp in user-defined units.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 600 Deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`CommandAborted`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been aborted. ● TRUE: Execution has been aborted by another function block.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.

Output	Data type	Description
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`InGear`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: If the adjusted gear ratio is reached for the first time.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

This requires the parameter **GEARratio** to be 0 (Refer to the drive documentation *(see page 11)*). This way, `RatioNumerator` and `RatioDenominator` are used to calculate the gear ratio.

The enabled direction of movement in the operating mode Electronic Gear is set via the drive parameter **GEARdir_enabl**.

The operating mode Electronic Gear with the method Position Synchronization is started with the function block `GearInPos_LXM32` *(see page 139)* .

In the operating mode Electronic Gear with the method Velocity Synchronization, a position overtravel does not trigger a detected error. A position overtravel results in a loss of the zero point.

Additional Information

Operating Mode Electronic Gear *(see page 138)*

Chapter 4
Function Blocks - Administrative

What Is in This Chapter?

This chapter contains the following sections:

Section	Topic	Page
4.1	Reading a Parameter	146
4.2	Writing a Parameter	165
4.3	Inputs and Outputs	194
4.4	Error Handling	207

Section 4.1
Reading a Parameter

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_ReadActualTorque	147
MC_ReadActualVelocity	149
MC_ReadActualPosition	151
MC_ReadAxisInfo	153
MC_ReadMotionState	156
MC_ReadStatus	159
MC_ReadParameter	162

MC_ReadActualTorque

Functional Description

This function block reads the actual torque of the motor.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Torque`	INT	Value range: -32768...32767 Actual torque in user-defined units. 100.0 % corresponds to the continuous stall torque. The read value is indicated in increments of 0.1 %. Example: The value `Torque` = 300 is read. This means that the effective torque amounts to 30 % of the nominal torque of the motor.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadActualVelocity

Functional Description

This function block reads the actual velocity of the motor.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Velocity`	DINT	Value range: -2147483648...2147483647 Actual velocity in user-defined units.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadActualPosition

Functional Description

This function block reads the actual position.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Position`	DINT	Value range: -2147483648...2147483647 Default value: 0. Actual position in user-defined units.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadAxisInfo

Functional Description

This function block gets status information on the axis.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`LimitSwitchPos`	BOOL	Value range: FALSE, TRUE. ● TRUE: Positive limit switch is triggered.
`LimitSwitchNeg`	BOOL	Value range: FALSE, TRUE. ● TRUE: Negative limit switch is triggered.
`CommunicationReady`	BOOL	Value range: FALSE, TRUE. ● TRUE: Network has been initialized and is ready for communication.
`ReadyForPowerOn`	BOOL	Value range: FALSE, TRUE. ● TRUE: Drive is ready to enable the power stage.
`PowerOn`	BOOL	Value range: FALSE, TRUE. ● TRUE: Power stage is enabled.
`IsHomed`	BOOL	Value range: FALSE, TRUE. ● TRUE: Reference point is valid (axis homed).
`AxisWarning`	BOOL	Value range: FALSE, TRUE. ● TRUE: An alert is active.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadMotionState

Functional Description

This function block outputs status information on the current movement.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`ConstantVelocity`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: A movement at a constant velocity is performed.
`Accelerating`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: The motor accelerates.
`Decelerating`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: The motor decelerates.
`DirectionPositive`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: The motor shaft rotates in positive direction.
`DirectionNegative`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: The motor shaft rotates in negative direction.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadStatus

Functional Description

This function block reads the current status of the drive.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Errorstop`	DINT	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Errorstop.
`Disabled`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Power stage is disabled.
`Stopping`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Stopping.
`Homing`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Homing.
`StandStill`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state StandStill.

Output	Data type	Description
`DiscreteMotion`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Discrete Motion.
`ContinuousMotion`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Continuous Motion.
`SynchronizedMotion`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● TRUE: Drive in operating state Synchronized Motion.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

MC_ReadParameter

Functional Description

This function block reads a value from a specific parameter.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Input	Data type	Description
`ParameterNumber`	UINT	Value range: 0...65535 Default value: 1000 ATV: ● 10: Actual velocity **RFRD**. ● 11: Target velocity **LFRD**. ● 1000: Selection via input `Address`. LXM32M: ● 1: Reference position (from profile generator). **_RAMP_p_target** ● 2: Positive position limit of software limit switch. **MON_swLimP** ● 3: Negative position limit of software limit switch. **MON_swLimN** ● 4: Monitoring of the positive software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **MON_SW_Limits** ● 5: Monitoring of the negative software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **MON_SW_Limits** ● 8: Maximum user-defined value for velocities in user-defined units **_ScaleVELmax** ● 10: Actual velocity in user-defined units. **_v_act** ● 11: Target velocity in user-defined units. **_RAMP_v_target** ● 12/14: Maximum user-defined value for acceleration and deceleration in user-defined units. **_ScaleRAMPmax** ● 1000: Selection via input `Address`. Lexium ILA, ILE and ILS integrated drives: ● 1: Reference position (from profile generator). **_p_tarRAMPusr** ● 2: Positive position limit of software limit switch. **SPVswLimPusr** ● 3: Negative position limit of software limit switch. **SPVswLimNusr** ● 4: Monitoring of the positive software limit switch. (Activated: Bit 0= 0. Deactivated: Bit 0 = 1). **SPV_SW_Limits** ● 5: Monitoring of the negative software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **SPV_SW_Limits** ● 8: Maximum user-defined value for velocities in user-defined units. **RAMPn_max** ● 10: Actual velocity in user-defined units. **_n_act** ● 11: Target velocity in user-defined units. **_n_targetRAMP** ● 1000: Selection via input `Address`.
`Address`	DWORD	Address of the parameter to be read. Can only be used if the input `ParameterNumber` = 1000. Refer to the product manual *(see page 11)* for a list of the parameters with the corresponding address of the parameters. Example of the address for CANopen: Index: 2038_h, Subindex: 05_h -> **00203805_h** Example of the address for Ethernet/IP: Class: $8C_h$, Instance: 01_h, Attribute: 05h -> **$008C0105_h$** Example of the address for ModbusTCP: Logic/Modbus address: $219C_h$ -> **$0000219C_h$**

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Value`	DINT	Value range: -2147483648...2147483647 Default value: 0 Value of the parameter.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Reading a Parameter *(see page 146)*

Section 4.2
Writing a Parameter

What Is in This Section?

This section contains the following topics:

Topic	Page
MC_WriteParameter	166
SetDriveRamp_ATV	169
SetDriveRamp_LXM32	171
SetDriveRamp_ILX	173
SetLimitSwitch_LXM32	175
SetLimitSwitch_ILX	177
SetFrequencyRange_ATV	179
ResetParameters_ATV	182
ResetParameters_LXM32	184
ResetParameters_ILX	186
StoreParameters_ATV	188
StoreParameters_LXM32	190
StoreParameters_ILX	192

MC_WriteParameter

Functional Description

This function block writes a value to a specific parameter.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Input	Data type	Description
`ParameterNumber`	INT	Value range: 0...65535 ATV: ● 1000: Selection via input `Address`. LXM32M: ● 2: Positive position limit of software limit switch. **MON_swLimP** ● 3: Negative position limit of software limit switch. **MON_swLimN** ● 4: Monitoring of the positive software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **MON_SW_Limits** ● 5: Monitoring of the negative software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **MON_SW_Limits** ● 1000: Selection via input `Address`. Lexium ILA, ILE and ILS integrated drives: ● 2: Positive position limit of software limit switch. **SPVswLimPusr** ● 3: Negative position limit of software limit switch. **SPVswLimNusr** ● 4: Monitoring of the positive software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **SPV_SW_Limits** ● 5: Monitoring of the negative software limit switch. (Activated: Bit 0 = 0. Deactivated: Bit 0 = 1). **SPV_SW_Limits** ● 1000: Selection via input `Address`.
`Address`	DWORD	Address of the parameter to be written. Fieldbus address (examples): CANopen: Index: 2038_h, Subindex: 05_h -> 00203805_h Ethernet/IP: Class: $8C_h$, Instance: 01_h, Attribute: 05h -> $008C0105_h$ ModbusTCP: Logic/Modbus address: $219C_h$ -> $0000219C_h$ Refer to the product manual *(see page 11)* for a list of the parameters with the corresponding address of the parameters. Can only be used if the input `ParameterNumber` = 1000.
`Length`	UINT	Value range: 0...4 Default value: 0 Length of the parameter to be written in bytes. Refer to the product manual *(see page 11)* for a list of the parameters with the corresponding length of the parameters.
`Value`	DINT	Value range: -2147483648...2147483647 Default value: 0 Value to be written to the parameter. The units of the values depend on the parameter.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If the inputs `ParameterNumber`, `Address`, `Length` or `Value` are modified while `Busy` is TRUE, the new values are not used until the function block is executed again.

Additional Information

Writing a Parameter *(see page 165)*

SetDriveRamp_ATV

Functional Description

This function block configures the acceleration ramp and the deceleration ramp of the axis.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Acceleration`	DINT	Value range: 0...9999 Default value: 30 Unit: 0.1 s. **Example:** With a value of 30, 3 seconds are required to accelerate from 0 to the nominal frequency of the motor. It must be possible to reach the value with the available torque of the motor.

Input	Data type	Description
`Deceleration`	DINT	Value range: 0...9999 Default value: 30 Unit: 0.1 s. **Example:** With a value of 30, 3 seconds are required to decelerate from the nominal frequency of the motor to 0. It must be possible to reach the value with the available torque of the motor.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

SetDriveRamp_LXM32

Functional Description

This function block configures the acceleration ramp and the deceleration ramp of the axis.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Acceleration`	DINT	Value range: 1...2147483647 Default value: 600 Acceleration ramp in user-defined units.
`Deceleration`	DINT	Value range: 1...2147483647 Default value: 600 Deceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

SetDriveRamp_ILX

Functional Description

This function block configures the acceleration ramp and the deceleration ramp of the axis.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`Acceleration`	DINT	Value range: 30...3000000 Default value: 600 Acceleration ramp in user-defined units.
`Deceleration`	DINT	Value range: 30...3000000 Default value: 750 Acceleration ramp in user-defined units.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

SetLimitSwitch_LXM32

Functional Description

This function block configures the positive limit switch (**LIMP**) and the negative limit switch (**LIMN**).

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`LimitSwitch`	UINT	Value range: 1...2 Default value: 1 Selects the address of limit switch. ● 1: Limit switch in positive direction of movement **LIMP** ● 2: Limit switch in negative direction of movement **LIMN**
`Mode`	UINT	Value range: 0...2 Default value: 0 ● 0: Deactivate limit switch ● 1: Activate limit switch as normally closed contact ● 2: Activate limit switch as normally open contact

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The function block can only be executed in the PLCopen state Disabled (operating state 4 Ready To Switch On of drive). To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Additional Information

Writing a Parameter *(see page 165)*

SetLimitSwitch_ILX

Functional Description

This function block enables and disables the limit switches and sets the activation edge.

The input `LimitSwitch` selects the address for the limit switch and the input `Mode` sets the value for the activation edge.

NOTE: The function block can be executed only in state Disabled.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`LimitSwitch`	UINT	Value range: 1...2 Default value: 1. Selects the address of limit switch. ● 1 = Limit switch in positive direction of movement. **IOsigLIMP** ● 2 = Limit switch in negative direction of movement. **IOsigLIMN**

Input	Data type	Description
`Mode`	UINT	Value range: 1...2 Default value: 0. Value for the activation edge. ● 0 = Deactivate limit switch ● 1 = Activate limit switch as normally closed contact ● 2 = Activate limit switch as normally open contact

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

SetFrequencyRange_ATV

Functional Description

This function block configures the frequency ranges of the drive for the function blocks `MC_MoveVelocity` and `MC_Jog`. If the frequency (speed of rotation) becomes inferior to the value in `LowFrequency`, the drive uses the frequency specified in `LowFrequency` without triggering an error message. If the frequency (speed of rotation) exceeds the value in `HighFrequency`, the drive uses the frequency specified in `HighFrequency` without triggering an error message.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.
`LowFrequency`	DINT	Value range: 0...`HighFrequency` Default value: 0 Unit: 0.1 Hz Motor frequency at minimum reference value. **NOTE:** If the value of `LowFrequency` exceeds the value of `HighFrequency`, the value of `HighFrequency` is used.

Input	Data type	Description
`HighFrequency`	DINT	Value range: `LowFrequency`...`MaxFrequency` Default value: 500 Unit: 0.1 Hz Motor frequency at maximum reference value. **NOTE:** If the value of `HighFrequency` exceeds the value of `MaxFrequency`, the value of `MaxFrequency` is used.
`MaxFrequency`	DINT	Value range: 100...5000/10000 (refer to the product manual) Default value: 600 Unit: 0.1 Hz Maximum permissible motor frequency. Adapt the value to the motor and the mechanical situation.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The function block can only be executed in the PLCopen state Disabled (operating state 3 Switch On Disabled of drive). To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Additional Information

Writing a Parameter *(see page 165)*

ResetParameters_ATV

Functional Description

This function block restores parameters to the factory settings.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The function block can only be executed in the PLCopen state Disabled (operating state 4 Ready To Switch On of drive). To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Additional Information

Writing a Parameter *(see page 165)*

ResetParameters_LXM32

Functional Description

This function block restores parameters to the factory settings.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

The function block can only be executed in the PLCopen state Disabled (operating state 4 Ready To Switch On of drive). To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Parameters are restored to the factory settings with the exception of:
- Communication parameters
- Inversion of direction of movement
- Type of reference value signal for PTI interface
- Settings of encoder simulation
- Functions of digital inputs and outputs

The new settings are not saved to the non-volatile memory. Use `StoreParameters_LXM32` *(see page 190)* to save the new settings to the non-volatile memory.

Additional Information

Writing a Parameter *(see page 165)*

ResetParameters_ILX

Functional Description

This function block restores parameters to the factory settings.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Notes

The function block can only be executed in the PLCopen state Disabled (operating state 4 Ready To Switch On of drive). To transition to this state, disable the power stage with the function block `MC_Power` *(see page 40)*.

Parameters are restored to the factory settings with the exception of:
- Communication parameters
- Inversion of direction of movement
- Functions of digital inputs and outputs

The new settings are not saved to the non-volatile memory. Use `StoreParameters_ILX` *(see page 192)* to save the new settings to the non-volatile memory.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

StoreParameters_ATV

Functional Description

This function block saves the parameter values to the non-volatile memory.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

StoreParameters_LXM32

Functional Description

This function block saves the parameter values to the non-volatile memory.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

StoreParameters_ILX

Functional Description

This function block saves the parameter values to the non-volatile memory.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Writing a Parameter *(see page 165)*

Section 4.3
Inputs and Outputs

What Is in This Section?

This section contains the following topics:

Topic	Page
ReadAnalogInput_ATV	195
MC_ReadDigitalInput	198
MC_ReadDigitalOutput	201
MC_WriteDigitalOutput	204

ReadAnalogInput_ATV

Functional Description

This function block reads the value of an analog input.

Library and Namespace

Library name: **GMC Independent Altivar**

Namespace: **GIATV**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Input	Data type	Description
`InputNumber`	UINT	Default value: 1 ATV32/ATV320: Value range: 1...3 ● 1: AI1 ● 2: AI2 ● 3: AI3 ATV340/ATV6••/ATV9••: Value range: 1...5 ● 1: AI1 ● 2: AI2 ● 3: AI3 ● 4: AI4 (with expansion card) ● 5: AI5 (with expansion card) ATV71: Value range: 1...4 ● 1: AI1 ● 2: AI2 ● 3: AI3 (with expansion card) ● 4: AI4 (with expansion card)

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.

Output	Data type	Description
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Value`	INT	Default value: 0 Corresponds to the input voltage in mV or the input current in 0.001 mA increments at the selected analog input.

Inputs/Outputs

Input/Output	Data type	Description
`Input`	Input_Ref	`Input` is a special data type for digital and analog inputs (if available). The data type corresponds to the axis reference from the device configuration (instance) to which the inputs belong (similar to `Axis`). In the case of function blocks provided for reading analog and digital inputs, `Input` replaces the input `Axis`.

Additional Information

Inputs and Outputs *(see page 194)*

MC_ReadDigitalInput

Functional Description

This function block reads the state of a digital input.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Input	Data type	Description
`InputNumber`	UINT	Default value: 1 ATV32/ATV320: Value range: 1...6 ● 1: LI1 ● ... ● 6: LI6 ATV340/ATV6••/ATV9••: Value range: 1...14 ● 1: DI1 ● ... ● 8: DI8 ● 9: DI11 (with expansion card) ● ... ● 14: DI16 (with expansion card) ATV71: Value range: 1...14 ● 1: LI1 ● ... ● 6: LI6 ● 7: LI7 (with expansion card) ● ... ● 14: LI14 (with expansion card) LXM32M: Value range: 1...5 ● 1: DI1 ● ... ● 5: DI5 Lexium ILA, ILE and ILS integrated drives: Value range: 1...4 ● 1: LIO1 ● ... ● 4: LIO4

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.

Output	Data type	Description
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Value`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Level at selected input is 0 V. ● TRUE: Level at selected input is 24 V.

Inputs/Outputs

Input/Output	Data type	Description
`Input`	MC_Input_Ref	`Input` is a special data type for digital and analog inputs (if available). The data type corresponds to the axis reference from the device configuration (instance) to which the inputs belong (similar to `Axis`). In the case of function blocks provided for reading analog and digital inputs, `Input` replaces the input `Axis`.

Additional Information

Inputs and Outputs *(see page 194)*

MC_ReadDigitalOutput

Functional Description

This function block reads the state of a digital output.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Input	Data type	Description
`OutputNumber`	UINT	ATV32/ATV320: Value range: 1...3 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: LO1 ATV340/ATV6••/ATV9••: Value range: 1...10 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: Relay3 ● 4: Relay4 (with expansion card) ● 5: Relay5 (with expansion card) ● 6: Relay6 (with expansion card) ● 7: DQ1 (only ATV340 and ATV9••) ● 8: DQ2 (only ATV340) ● 9: DQ11 (with expansion card) ● 10: DQ12 (with expansion card) ATV71: Value range: 1...8 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: Relay3 (with expansion card) ● 4: Relay4 (with expansion card) ● 5: LO1 (with expansion card) ● ... ● 8: LO4 (with expansion card) LXM32M: Value range: 0...2 Default value: 0 ● 0: DQ0 ● ... ● 2: DQ2 Lexium ILA, ILE and ILS integrated drives: Value range: 1...4 Default value: 1 ● 1: LIO1 ● ... ● 4: LIO4

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`Value`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Level at selected input is 0 V. ● TRUE: Level at selected input is 24 V.

Inputs/Outputs

Input/Output	Data type	Description
`Output`	MC_Output_Ref	`Output` is a special data type for digital and analog outputs (if available). The data type corresponds to the axis reference from the device configuration (instance) to which the outputs belong (similar to `Axis`). In the case of function blocks provided for writing and reading digital inputs, `Output` replaces the output `Axis`.

Additional Information

Inputs and Outputs *(see page 194)*

MC_WriteDigitalOutput

Functional Description

This function block sets the state of a digital output.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed.

Input	Data type	Description
`OutputNumber`	UINT	ATV32/ATV320: Value range: 1...3 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: LO1 ATV340/ATV6••/ATV9••: Value range: 1...10 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: Relay3 ● 4: Relay4 (with expansion card) ● 5: Relay5 (with expansion card) ● 6: Relay6 (with expansion card) ● 7: DQ1 (only ATV340 and ATV9••) ● 8: DQ2 (only ATV340) ● 9: DQ11 (with expansion card) ● 10: DQ12 (with expansion card) ATV71: Value range: 1...8 Default value: 1 ● 1: Relay1 ● 2: Relay2 ● 3: Relay3 (with expansion card) ● 4: Relay4 (with expansion card) ● 5: LO1 (with expansion card) ● ... ● 8: LO4 (with expansion card) LXM32M: Value range: 0...2 Default value: 0 ● 0: DQ0 ● ... ● 2: DQ2 Lexium ILA, ILE and ILS integrated drives: Value range: 1...4 Default value: 1 ● 1: LIO1 ● ... ● 4: LIO4

Input	Data type	Description
`Value`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: 0 V is written to the selected signal output. ● TRUE: 24 V is written to the selected signal output.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Output`	MC_Output_Ref	`Output` is a special data type for digital and analog outputs (if available). The data type corresponds to the axis reference from the device configuration (instance) to which the outputs belong (similar to `Axis`). In the case of function blocks provided for writing and reading digital inputs, `Output` replaces the output `Axis`.

Additional Information

Inputs and Outputs *(see page 194)*

Section 4.4
Error Handling

What Is in This Section?

This section contains the following topics:

Topic	Page
ReadAxisWarning_LXM32	208
ReadAxisWarning_ILX	210
MC_ReadAxisError	212
MC_Reset	214

ReadAxisWarning_LXM32

Functional Description

This function block reads the error code of the most recent detected error of error class 0.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`WarningID`	WORD	Default value: 0 Error code of the most recent detected error of error class 0. Refer to the drive documentation for a description of the error codes and classes.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Error Handling *(see page 207)*

ReadAxisWarning_ILX

Functional Description

This function block reads the error code of the most recent detected error of error class 0.

Library and Namespace

Library name: **GMC Independent Lexium**

Namespace: **GILXM**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`WarningID`	WORD	Default value: 0. Error code of the most recent detected error of error class 0. Refer to the drive documentation for a description of the error codes and classes.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Error Handling *(see page 207)*

MC_ReadAxisError

Functional Description

This function block reads the error information relative to the most recent detected error.

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Enable`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. The input `Enable` starts or terminates execution of a function block. ● FALSE: Execution of the function block is terminated. The outputs `Valid`, `Busy`, and `Error` are set to FALSE. ● TRUE: The function block is being executed. The function block continues executing as long as the input `Enable` is set to TRUE.

Outputs

Output	Data type	Description
`Valid`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started or an error has been detected. The values at the outputs are not valid. ● TRUE: Execution has been completed without an error detected. The values at the outputs are valid and can be further processed.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.
`AxisErrorID`	WORD	Initial value: 0 This output shows library-specific and drive-specific error codes. Refer to the chapter Library Diagnostic Codes *(see page 27)* for a list of the library-specific error codes. Refer to the drive documentation for an overview of the drive-specific error codes.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Additional Information

Error Handling *(see page 207)*

MC_Reset

Functional Description

This function block acknowledges an error message. The error memory is cleared so that it is available for new error messages. If the power stage is disabled by the automatic error response, it can be enabled again if the cause of the detected error has been rectified when the error message is acknowledged.

NOTE: If there is no drive error active, the function block itself will set `ErrorId` = 109 (Device state invalid).

Library and Namespace

Library name: **GMC Independent PLCopen MC**

Namespace: **GIPLC**

Graphical Representation

Inputs

Input	Data type	Description
`Execute`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. A rising edge of the input `Execute` starts the function block. The function block continues execution and the output `Busy` is set to TRUE. A rising edge at the input Execute is ignored while the function block is being executed. ● FALSE: If `Enable` is set to FALSE, the outputs `Done`, `Error`, or `CommandAborted` are set to TRUE for one cycle. ● TRUE: If `Enable` is set to FALSE, the outputs `Done`, `Error`, or `CommandAborted` remain set to TRUE.

Outputs

Output	Data type	Description
`Done`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution has not been started, or an error has been detected. ● TRUE: Execution terminated without an error detected.
`Busy`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Function block is not being executed. ● TRUE: Function block is being executed.
`Error`	BOOL	Value range: FALSE, TRUE. Default value: FALSE. ● FALSE: Execution of the function block is running, no error has been detected. ● TRUE: An error has been detected in the execution of the function block.
`ErrorID`	WORD	Returns the value of a diagnostic code. Refer to Library Diagnostic Codes *(see page 27)*. If the value is 0 and if the output `Error` of this function block is set to TRUE, then the diagnostic code can be read with the output `AxisErrorID` of the function block MC_ReadAxisError *(see page 212)*.

Inputs/Outputs

Input/Output	Data type	Description
`Axis`	Axis_Ref	Reference to the axis (instance) for which the function block is to be executed (corresponds to the name of the axis). The name of the axis must be defined in the SoMachine **Devices tree**.

Notes

If you have activated this function block, simultaneous use of the Control_ATV function block leads to unintended behavior.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not activate the Control_ATV function block when this function block is active.
- Deactivate this function block or allow the function block to terminate before activating the Control_ATV function block.

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

Additional Information

Error Handling *(see page 207)*

Glossary

A

application
A program including configuration data, symbols, and documentation.

B

byte
A type that is encoded in an 8-bit format, ranging from 00 hex to FF hex.

C

configuration
The arrangement and interconnection of hardware components within a system and the hardware and software parameters that determine the operating characteristics of the system.

controller
Automates industrial processes (also known as programmable logic controller or programmable controller).

E

expansion bus
An electronic communication bus between expansion I/O modules and a controller.

H

homing
The method used to establish the reference point for absolute movement.

I

I/O
(*input/output*)

P

PLCopen

For more information, refer to *http://www.plcopen.org/*.

program

The component of an application that consists of compiled source code capable of being installed in the memory of a logic controller.

Index

C

common inputs and outputs
- behavior of function blocks with the input Enable, *34*
- behavior of function blocks with the input Execute, *37*

Control_ATV, *78*

E

error handling, *207*

G

GearIn_LXM32, *142*
GearInPos_LXM32, *139*
general description
- libraries and related function blocks, *19*
- library diagnostic codes, *27*
- PLCopen state diagram, *23*
- transitions between function blocks, *24*

H

Halt_LXM32, *122*
Home_ILX, *102*
Home_LXM32, *98*

I

initialization, *40*
inputs and outputs, *194*

J

Jog_ILX, *51*
Jog_LXM32, *47*

M

MC_AbortTrigger, *134*
MC_Halt, *119*
MC_Home, *95*
MC_Jog, *44*
MC_MoveAbsolute, *85*
MC_MoveAdditive, *88*
MC_MoveRelative, *91*
MC_MoveVelocity, *65*
MC_Power, *40*
MC_ReadActualPosition, *151*
MC_ReadActualTorque, *147*
MC_ReadActualVelocity, *149*
MC_ReadAxisError, *212*
MC_ReadAxisInfo, *153*
MC_ReadDigitalInput, *198*
MC_ReadDigitalOutput, *201*
MC_ReadMotionState, *156*
MC_ReadParameter, *162*
MC_ReadStatus, *159*
MC_Reset, *214*
MC_SetPosition, *106*
MC_Stop, *109*
MC_TorqueControl, *56*
MC_TouchProbe, *125*
MC_WriteDigitalOutput, *204*
MC_WriteParameter, *166*
MoveVelocity_LXM32, *69*

O

operating mode Electronic Gear, *138*
operating mode Homing, *94*
operating mode Jog, *43*
operating mode Profile Position, *84*
operating mode Profile Torque, *55*
operating mode Profile Velocity, *64*

P

position capture via signal input, *124*

R

ReadAnalogInput_ATV, *195*
ReadAxisWarning_ILX, *210*
ReadAxisWarning_LXM32, *208*
reading a parameter, *146*
ResetParameters_ATV, *182*
ResetParameters_ILX, *186*
ResetParameters_LXM32, *184*

S

SetDriveRamp_ATV, *169*
SetDriveRamp_ILX, *173*
SetDriveRamp_LXM32, *171*
SetFrequencyRange_ATV, *179*
SetLimitSwitch_ILX, *177*
SetLimitSwitch_LXM32, *175*
SetStopRamp_ILX, *114*
SetStopRamp_LXM32, *112*
SetTorqueRamp_LXM32, *59*
Stop_LXM32, *116*
stopping, *108*
StoreParameters_ATV, *188*
StoreParameters_ILX, *192*
StoreParameters_LXM32, *190*

T

TorqueControl_LXM32, *61*
TouchProbe_ILX, *131*
TouchProbe_LXM32, *128*

V

VelocityControlAnalogInput_ATV, *72*
VelocityControlSelectAI_ATV, *75*

W

writing a parameter, *165*