



Program example

M221 with LXM32 via Modbus
TCP





Table of Contents

Introduction	3
Overview	3
Before You Begin	3
Start-Up and Test.....	5
Operations and Adjustments.....	6
M221 with LXM32 via ModbusTCP	7
System Configuration.....	8
Configuration of LMX32	9
Modbus TCP interface of Lexium 32M.....	10
Used Operation Mode in this example:.....	16
Ethernet configuration in M221	18
Explanation of Sourcecode.....	21
Animation table	21
Task configuration.....	23
Status machine	24
Operation modes:.....	28
Homing	31
Profile velocity	34
Profile torque	36
Profile position (point to point).....	38
Jog mode (manual mode)	41
Reading and writing parameter via the implicit parameter channel	44



Introduction

Overview

This chapter gives the introduction.

Contents of this chapter

This chapter contains the following topics:

Topic	Page
Before you begin	03
Start-Up and Test	05
Operations and Adjustments	06

Before You Begin

General

The products specified in this document have been tested under actual service conditions. Of course, your specific application requirements may be different from those assumed for this and any related examples described herein. In that case, you will have to adapt the information provided in this and other related documents to your particular needs. To do so, you will need to consult the specific product documentation of the hardware and/or software components that you may add or substitute for any examples specified in this documentation. Pay particular attention and conform to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

© 2014 Schneider Electric. All rights reserved

WARNING

REGULATORY INCOMPATIBILITY

Be sure that all equipment applied and systems designed comply with all applicable local, regional and national regulations and standards

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

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



The use and application of the information contained herein require expertise in the design and programming of automated control systems. Only the user or integrator can be aware of all the conditions and factors present during installation and setup, operation, and maintenance of the machine or process, and can therefore determine the automation and associated equipment and the related safeties and interlocks which can be effectively and properly used. When selecting automation and control equipment, and any other related equipment or software, for a particular application, the user or integrator must also consider any applicable local, regional or national standards and/or regulations.

Some of the major software functions and/or hardware components used in the proposed architectures and examples described in this document cannot be substituted without significantly compromising the performance of your application. Further, any such substitutions or alterations may completely invalidate any proposed architectures, descriptions, examples, instructions, wiring diagrams and/or compatibilities between the various hardware components and software functions specified herein and in related documentation. You must be aware of the consequences of any modifications, additions or substitutions.

A residual risk, as defined by EN/ISO 12100-1, Article 5, will remain if

- it is necessary to modify the recommended logic and if the added or modified components are not properly integrated in the control circuit.
- you do not follow the required standards applicable to the operation of the machine, or if the adjustments to and the maintenance of the machine are not properly made (it is essential to strictly follow the prescribed machine maintenance schedule).
- the devices connected to any safety outputs do not have mechanically-linked contacts.

CAUTION

EQUIPMENT INCOMPATIBILITY

Read and thoroughly understand all device and software documentation before attempting any component substitutions or other changes related to the application examples provided in the document

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



Start-Up and Test

Before using electrical control and automation equipment after design and installation, the application and associated functional safety system must be subjected to a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such testing be made and that enough time is allowed to perform complete and satisfactory testing.

CAUTION

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 injury, or equipment damage.

Verify that the completed system, including the functional safety system, is free from all short circuits and grounds, except those grounds installed according to local regulations. If high-potential voltage testing is necessary, follow the recommendations in equipment documentation to help prevent injury or equipment damage.



Operations and Adjustments

General

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 installed and operated.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the hands and other parts of the body are free to enter the pinch points or other hazardous areas where 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.

WARNING

UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY

- 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.

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 examples and implementations suggested herein. It is sometimes possible to adjust the equipment incorrectly and this produce unsatisfactory or unsafe operation. Always use the manufacturer instructions as a guide to functional adjustments. Personnel who have access to these adjustments must be familiar with the equipment manufacturer instructions and the machinery used with the electrical equipment. Only those operational adjustments actually required by the machine operator should be accessible to the operator. Access to other controls should be restricted to help prevent unauthorized changes in operating characteristics.



M221 with LXM32 via ModbusTCP

This document describes a small application using a M221-PLC and one LxM32M drive with EthernetIP modul connected via ModbusTCP, using IO scanning with DriveProfileLexium.

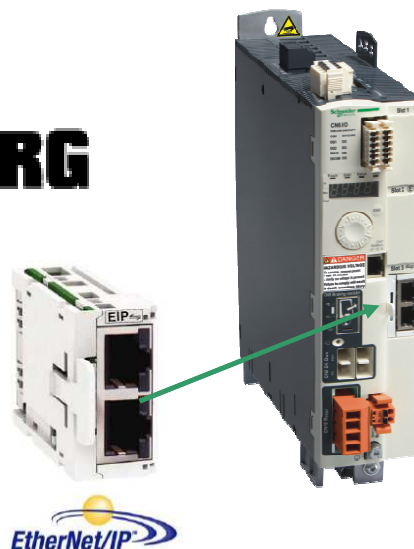
It is not intended to replace any specific product documentation. On the contrary, it offers additional information to the product documentation, for installing, configuring and starting up the system. A detailed functional description or the specification for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The program example includes how to activate the operation modes:

- Homing
- Profile Velocity mode
- Profile Torque mode
- Point to point
- Manual mode
- How to write and read parameters



MODBUS.ORG



Designed 11.11.2015 by

Please contact your local Schneider Electric partner.
www.schneider-electric.com



System Configuration



M221ME32TK
IP address 192.168.100.1



LXM32M
IP address 192.168.100.10



Configuration of LMX32

Following parameter settings are mandatory in LXM32:

DEVcmdinterf	Fieldbus Control Mode	Specification of the control mode
--------------	-----------------------	-----------------------------------

SoMove 2.3 - Untitled Project.psx*

File View Communication Device Tools Help

Data synchronized

My Device Parameters list Error memory Visualization Scope Tuning Motion Sequence Startup messages Operate

Lexium32

- Simply start
- Basic configuration
- Communication
 - Fieldbus EtherNet/IP (ETH)
 - Ethernet I/O data
 - Modbus TCP
 - Ethernet FDR

Name	Value	Description	Min. value	Max. value	Logical address
EthMode	Modbus TCP	Protocol			17410
EthIpMode	Manual	Type of obtaining IP address			17418
EthIPmodule1	192	IP address Ethernet module, byte 1	0	255	17422
EthIPmodule2	168	IP address Ethernet module, byte 2	0	255	17424
EthIPmodule3	100	IP address Ethernet module, byte 3	0	255	17426
EthIPmodule4	10	IP address Ethernet module, byte 4	0	255	17428

Exd. Operating state Power

POWER DISABLED [4] Ready To Switch On

Operating modes

Jog Step-In 20 [1usr_p] Low speed 60 [1usr_v]

Continuous Wait time 500 ms High speed 180 [1usr_v]

Step Movement Set position

Proceed Control

Set Halt CTRL 1 CTRL 2 Force

Global info

HALT = inactive
_p_act = 308 [1usr_p]
_AccessInfo = Fieldbus main ch.
_DEVcmdinterf = FB Control Mode
_DCOMopmd_act = Jog
Ref_OK = inactive

Project Loaded

SoMove 2.3 - Untitled Project.psx*

File View Communication Device Tools Help

Data synchronized

My Device Parameters list Error memory Visualization Scope Tuning Motion Sequence Startup messages Operate

Lexium32

- Simply start
- Basic configuration
- Communication
 - Fieldbus EtherNet/IP (ETH)
 - Ethernet I/O data
 - Modbus TCP
 - Ethernet FDR

Name	Value	Description	Min. value	Max. value	Logical address
EthMode	Modbus TCP	Protocol			17410
EthMbScanner	On	Modbus TCP I/O scanning			17488
EthMbScanTimeout	2.0 s	Modbus TCP I/O scanning timeout	0.0 s	60.0 s	17498
EthIPmaster1	192	IP address master, byte 1	0	255	17490
EthIPmaster2	168	IP address master, byte 2	0	255	17492
EthIPmaster3	100	IP address master, byte 3	0	255	17494
EthIPmaster4	1	IP address master, byte 4	0	255	17496
EthMbIPswap1	192	IP address of master for Modbus word swap, byte 1	0	255	17568
EthMbIPswap2	168	IP address of master for Modbus word swap, byte 2	0	255	17570
EthMbIPswap3	100	IP address of master for Modbus word swap, byte 3	0	255	17572
EthMbIPswap4	1	IP address of master for Modbus word swap, byte 4	0	255	17574

Exd. Operating state Power

POWER DISABLED [4] Ready To Switch On

Operating modes

Jog Step-In 20 [1usr_p] Low speed 60 [1usr_v]

Continuous Wait time 500 ms High speed 180 [1usr_v]

Step Movement Set position

Proceed Control

Set Halt CTRL 1 CTRL 2 Force

Global info

HALT = inactive
_p_act = 308 [1usr_p]
_AccessInfo = Fieldbus main ch.
_DEVcmdinterf = FB Control Mode
_DCOMopmd_act = Jog
Ref_OK = inactive

Project Loaded



Modbus TCP interface of Lexium 32M

There are two possibilities to control the drive Modbus TCP and to activate the different modes of operation.

Control of LXM32 according DSP402

Control of LXM32 according DriveProfileLexium (IOscanning interface)

The usage of DriveProfileLexium has some advantages. It is possible to start any mode of operation within one plc cycle. It requires less application code.

This document describes an example for ModbusTCP with DriveProfileLexium (IOscanning interface).

Drive Profile Lexium:

Output – Input: Output and input refer to the direction of data transmission from the perspective of the master.

- Output: Commands from the master to the slave
- Input: Status messages from the slave to the master

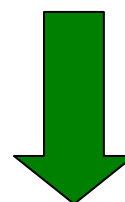


Output mapping for IO scanning

Output register size is 13. Up to 3 output parameters can be mapped optionally. Then size is 15, 17 or 19.

This example uses size 13.

Word no.	Name	Modbus address	Part
1	PCTRLms	6982	fix part
2			
3			
4			
5	PVms	6984	
6			
7			
8	dmControl	6974	
9			
10	refA32	6976	
11			
12	refB32	6978	
13			
14	ACC	1556	
15			
16	DEC	1558	
17			
18	optional data	<configured>	configurable part
19			
	optional data	<configured>	



Word 1 .. 4 (PCTRLms and PVms) is the implicit parameter channel for write or read parameters request to the drive.

Word 5 (dmControl) **is used to set the operating state and the operating mode.**

Word 6 and 7, 8 and 9 (refA32 and ref32B) **is used to set the second value for the operating mode.**

The meaning depends on the operating mode. It is described in the EthernetIP manual of Lexium 32M.

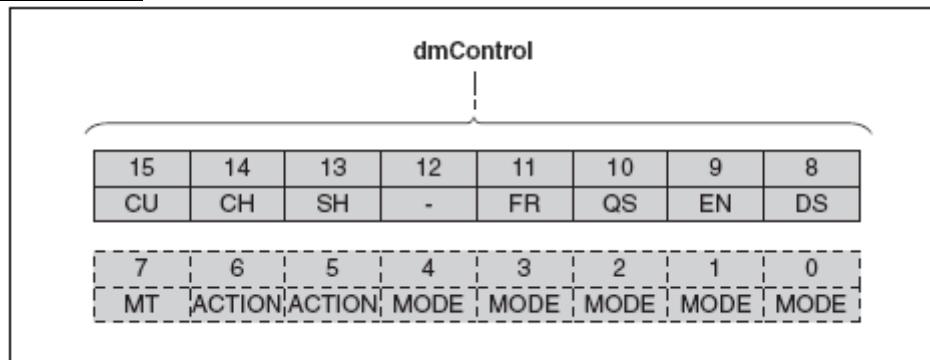
Word 10..13 (ACC and DEC): **Acceleration and deceleration ramp.**

Word 14..19 (optional data): **parameters can be mapped optionally.**



Description of the output mapping:

dmControl:



Bit	Name	Description
0 ... 4	MODE	Operating mode Value 01 _h : Profile Position Value 03 _h : Profile Velocity Value 04 _h : Profile Torque Value 06 _h : Homing Value 1F _h : Jog Value 1E _h : Electronic Gear
5 ... 6	ACTION	Operating mode-dependent
7	MT	Handshake via Mode Toggle

Bit	Name	Meaning	Operating state
8	DS	Disable power stage	6 Operation Enabled -> 4 Ready To Switch On
9	EN	Enable power stage	4 Ready To Switch On -> 6 Operation Enabled
10	QS	Executing a "Quick Stop"	6 Operation Enabled -> 7 Quick Stop Active
11	FR	Execute "Fault Reset"	7 Quick Stop Active -> 6 Operation Enabled 9 Fault -> 4 Ready To Switch On
12	-	Reserved	Reserved
13	SH	Execute "Halt"	6 Operation Enabled
14	CH	Clear "Halt"	6 Operation Enabled
15	CU	Resume operating mode interrupted by "Halt"	6 Operation Enabled

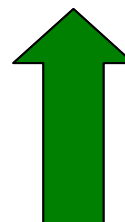


Input mapping for IO scanning

Input register size is 13. Up to 3 output parameters can be mapped optionally. Then size is 15, 17 or 19.

This example uses size 13.

Word no.	Name	Modbus address	Part
1	PCTRLsm	6994	fix part
2			
3			
4			
5	PVsm	6996	
6	driveStat	6986	
7	mfStat	6988	
8	motionStat	6990	
9	driveInput	6992	
10	p_actusr	7706	
11	v_act	7744	
12			
13	i_act	7686	
14	optional data	<configured>	configurable part
15			
16	optional data	<configured>	
17			
18	optional data	<configured>	
19			



Word 1 .. 4 (PCTRLsm and PVsm) is the implicit parameter channel for write or read parameter response from the drive.

Word 5 (driveStat) indicates the current operating state.

Word 6 (mfStat) indicates the current operating mode.

Word 7 (motionStat) provides information on the motor and the profile generator.

Word 8 (driveInput) is used to indicate the status of the digital signal inputs and signal outputs.

Word 9 and 10 (p_actusr) is used to provide information on the current motor position.

Word 11 and 12 (v_act) is used to provide information on the current velocity.

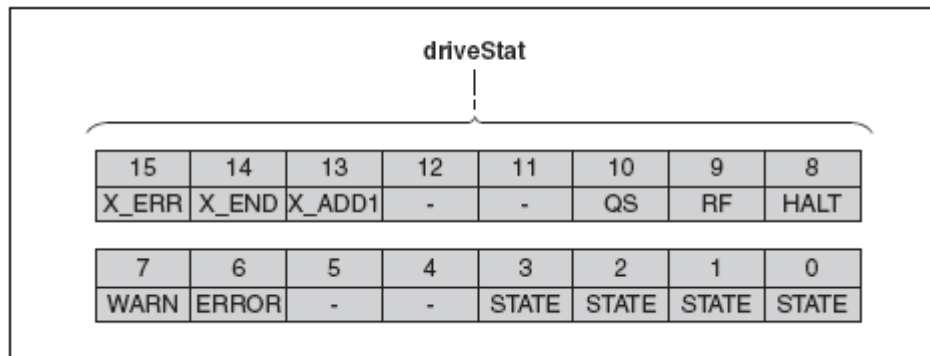
Word 13 (i_act) is used to provide information on the actual current.

Word 14..19 (optional data): parameters can be mapped optionally.



Description of the input mapping:

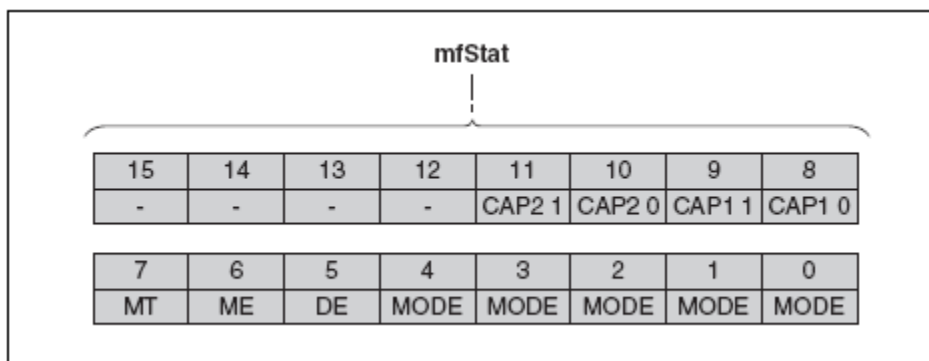
driveStat:



Bit	Name	Meaning
0 ... 3	STATE	Current operating state (binary coded)
4 ... 5	-	Reserved
6	ERROR	An error has occurred (error classes 1 ... 3)
7	WARN	A warning has occurred (error class 0)
8	HALT	"Halt" is active
9	RF	Homing valid
10	QS	"Quick Stop" is active
11 ... 12	-	Reserved
13	X_ADD1	Operating mode-dependent
14	X_END	Operating mode terminated
15	X_ERR	Operating mode terminated with error



mfStat:



Bit	Name	Description
0 ... 4	MODE	Indicates the current operating mode Value 01 _h : Profile Position Value 03 _h : Profile Velocity Value 04 _h : Profile Torque Value 06 _h : Homing Value 1F _h : Jog Value 1E _h : Electronic Gear
5	DE	The "DE" bit relates to parameters that are independent of "Mode Toggle" (MT). The "DE" bit is set if a data value in the process data channel is invalid.
6	ME	The "ME" bit relates to parameters that are dependent on "Mode Toggle" (MT). The "ME" bit is set if a request from a master (starting an operating mode) was rejected.
7	MT	Handshake via "Mode Toggle"
8 ... 9	CAP1	Bit 0 and bit 1 of parameter <code>_Cap1Count</code>
10 ... 11	CAP2	Bit 0 and bit 1 of parameter <code>_Cap2Count</code>
12 ... 15	-	Reserved



motionStat:

Bit	Meaning
0 ... 5	Reserved
6	MOTZ: Motor at a standstill
7	MOTP: Motor movement in positive direction
8	MOTN: Motor movement in negative direction
9 ... 10	Reserved
11	TAR0: Profile generator at standstill, reference velocity = 0
12	DEC: Profile generator decelerates
13	ACC: Profile generator accelerates
14	CNST: Profile generator moves at constant velocity
15	Reserved

driveInput:

Bit	Signal	Factory setting
0	DI0	Freely Available
1	DI1	Reference Switch (REF)
2	DI2	Positive Limit Switch (LIMP)
3	DI3	Negative Limit Switch (LIMN)
4	DI4	Freely Available
5	DI5	Freely Available
6	STO_A	Safety function STO ¹⁾
7	STO_B	Safety function STO ¹⁾
7	DQ0	No Fault
7	DQ1	Active
7	DQ2	Freely Available

1) Cannot be modified

Used Operation Mode in this example:

Operating mode Jog:

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
1F _h	As JOGactivate	-



Operating mode Profile Torque:

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
24 _h	As PTtq_target	As RAMP_tq_slope

Operating mode Profile Velocity:

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
23 _h	As PVv_target	-

Operating mode Profile Position:

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

Method	dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
absolute	01 _h	As PPv_target	As PPp_target
Relative with reference to the currently set target position	21 _h	As PPv_target	As PPp_target

Operating mode Homing:

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

Method	dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
Position setting	06 _h	-	As HMp_setP
Reference movement	26 _h	As HMmethod	-



Ethernet configuration in M221

Properties Configuration Programming Display

Messages

- MyController (TM221ME32TK)
 - Digital inputs
 - Digital outputs
 - Analog inputs
 - High Speed Counters
 - Pulse Generators
 - IO Bus
 - ETH1**
 - Modbus TCP
 - EtherNet/IP adapter
 - SL1 (Serial line)

Ethernet

Device name: M221

☐ IP address by DHCP

☐ IP address by BOOTP

☒ Fixed IP address

IP address: 192 - 168 - 100 - 1

Subnet mask: 255 - 255 - 255 - 0

Gateway address: 0 - 0 - 0 - 0

Transfer Rate: Auto

Properties Configuration Programming Display Commissioning

Messages

- MyController (TM221ME32TK)
 - Digital inputs
 - Digital outputs
 - Analog inputs
 - High Speed Counters
 - Pulse Generators
 - IO Bus
 - ETH1**
 - Modbus TCP
 - EtherNet/IP adapter
 - SL1 (Serial line)

Modbus TCP

Modbus mapping

☐ Enabled

Unit ID:

Output registers (%IWM): 0

Input registers (%QWM): 0

Client mode: Remote Server table (max 16)

Address: 0 - 0 - 0 - 0 Add

Unit ID: 255

Connection timeout (100 ms): 100

Index	Address	Unit ID	Connection timeout (100 ms)
1	192.168.100.10	255	100

IP address of LXM32

UnityID 255 for access to IOscanning interface



The data exchange is realized with the %WRITE_READ function block of SoMachineBasic:

Used	Address	Symbol	Link	Id	Timeout	ObjType	FirstWriteObj	WriteQuantity	IndexDataOut	FirstReadObj	ReadQuantity	IndexDataIn	Comment
<input checked="" type="checkbox"/>	%WRITE_READ_VAR0		3 - ETH1	1	100	0 (Mult. reg. - Mbs 2: 0	13	200	0	13	100		
<input type="checkbox"/>	%WRITE_READ_VAR1		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR2		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR3		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR4		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR5		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR6		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR7		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR8		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		
<input type="checkbox"/>	%WRITE_READ_VAR9		1 - SL1	1	100	0 (Mult. reg. - Mbs 2: 0	1	0	0	1	0		

Write Read Var properties													
Used	Address	Symbol	Link	Id	Timeout	ObjType	FirstWriteObj	WriteQuantity	IndexDataOut	FirstReadObj	ReadQuantity	IndexDataIn	Comment
<input checked="" type="checkbox"/>	%WRITE_READ_VAR0		3 - ETH1	1	100	0 (Mult. reg. - Mbs 2: 0	13	200	0	13	100		

Ethernet interface

Id1 = LXM32 with
IPaddress
192.168.100.10

Timeout setting 100ms
on Master side

Modbus start
address for Writing
Data is 0

Number of words
to be transmitted

Memory area in M221
for Write Data
%MW200 - %MW212

Modbus start
address for Reading
Data is 0

Memory area in M221
for Read Data
%MW100 - %MW112



With that configuration we can access DriveProfileLexium at the following memory areas:

Output data (from PLC to the drive):

Word no.	Name	Modbus address	PLC address
1	PCTRLms	6982	%MW200
2			%MW201
3	PVms	6984	%MW202
4			%MW203
5	dmControl	6974	%MW204
6	refA32	6976	%MW205
7			%MW206
8	refB32	6978	%MW207
9			%MW208
10	ACC	1556	%MW209
11			%MW210
12	DEC	1558	%MW211
13			%MW212
14	optional data	<configured>	<configured>
15			
16	optional data	<configured>	<configured>
17			
18	optional data	<configured>	<configured>
19			

Input data (from drive to PLC):

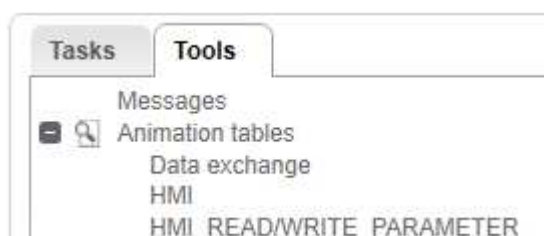
Word no.	Name	Modbus address	PLC address
1	PCTRLsm	6994	%MW100
2			%MW101
3	PVsm	6996	%MW102
4			%MW103
5	driveStat	6986	%MW104
6	mfStat	6988	%MW105
7	motionStat	6990	%MW106
8	driveInput	6992	%MW107
9	p_actusr	7706	%MW108
10			%MW109
11	v_act	7744	%MW110
12			%MW111
13	i_act	7686	%MW112
14	optional data	<configured>	<configured>
15			
16	optional data	<configured>	<configured>
17			
18	optional data	<configured>	<configured>
19			



Explanation of Sourcecode

Animation table

The program can be controlled and monitored via Animation tables.



“Data exchange” shows the data exchange between plc and drive

Data exchange							
				Add		Insert	
Used	Trace	Address	Symbol	Value	Force	Comment	
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW204	LXM32_DMCONTROL				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD205	LXM32_REAFA32				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD207	LXM32_REFB32				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD209	LXM32_ACC				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD211	LXM32_DEC				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW104	LXM32_DRIVE_STATUS				
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW105	LXM32_MFSTAT				
<input type="checkbox"/>	<input type="checkbox"/>	%MW106	LXM32_MOTIONSTAT				
<input type="checkbox"/>	<input type="checkbox"/>	%MW107	LXM32_IOSTAT				
<input type="checkbox"/>	<input type="checkbox"/>	%MD108	LXM32_P_ACTUSER				
<input type="checkbox"/>	<input type="checkbox"/>	%MD110	LXM32_V_ACT				
<input type="checkbox"/>	<input type="checkbox"/>	%MW112	LXM32_I_ACT				



With “HMI” it is possible to control LXM32.

E.g. by setting HMI_ENABLE the drive enables; HMI_FAULTRESET performs a fault reset.

HMI_OPMODE shows the current mode of operation; HMI_DRIVESTATUS shows the current drive status.

With HMI_HOMINGTYPE it is possible to choose the way homing is performed, by setting HMI_HOMING the movement will be started.

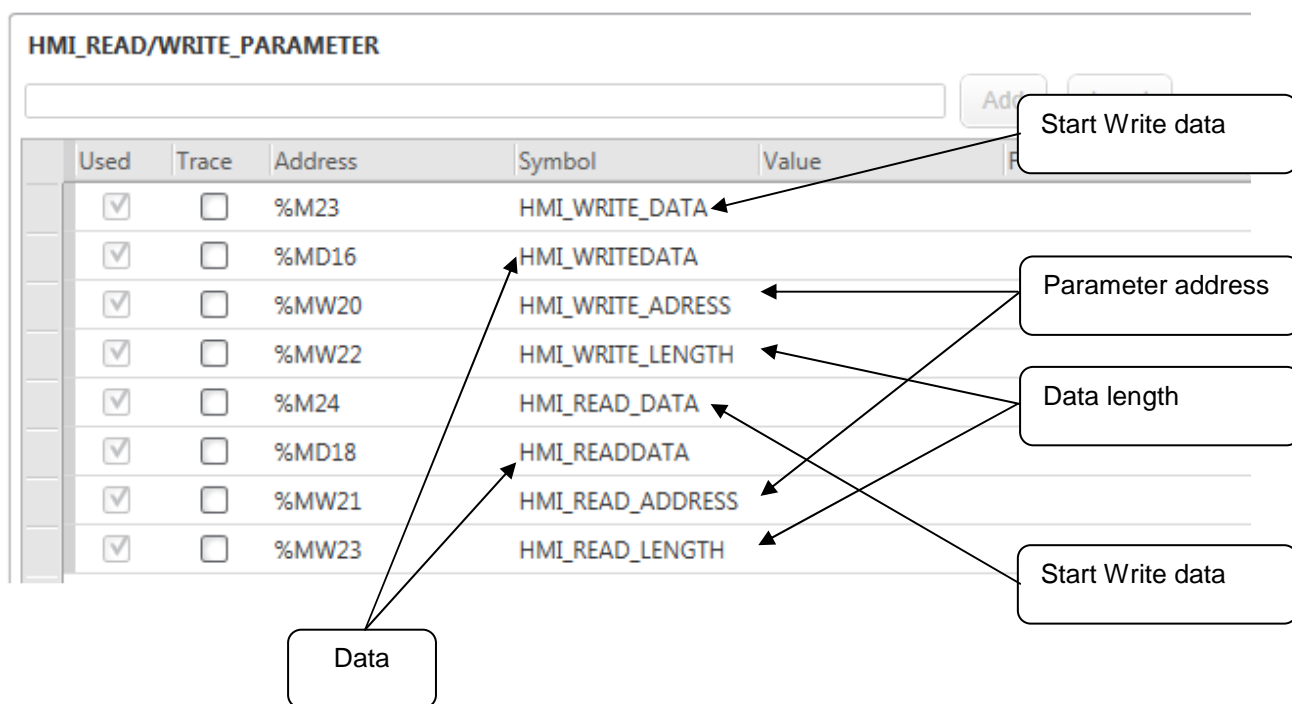
For the other modes of operation it is similar.

HMI

Used	Trace	Address	Symbol
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M10	HMI_ENABLE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M11	HMI_DISABLE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M12	HMI_QUICKSTOP
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M13	HMI_FAULTRESET
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW0	HMI_OPMODE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MW1	HMI_DRIVESTATUS
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD2	HMI_ACC
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD4	HMI_DEC
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M14	HMI_HOMING
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD6	HMI_HOMINGTYPE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M17	HMI_MOVE_VEL
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD8	HMI_VEL_SETSPEED
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M18	HMI_MOVE_TORQUE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD10	HMI_TORQUE_SETTORQUE
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M16	HMI_PTP_REL
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M15	HMI_PTP_ABS
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD12	HMI_PTP_SETSPEED
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%MD14	HMI_PTP_SETPOSITION
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M19	HMI_JOG_POS_SLOW
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M20	HMI_JOG_POS_FAST
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M21	HMI_JOG_NEG_SLOW
<input checked="" type="checkbox"/>	<input type="checkbox"/>	%M22	HMI_JOG_NEG_FAST



“HMI_READ/WRITE_PARAMETER” allows to read and write parameters.



Task configuration

The rungs are called in a cyclic task:

Master Task

Scan mode

☐ Normal

☒ Periodic (2...150 ms)

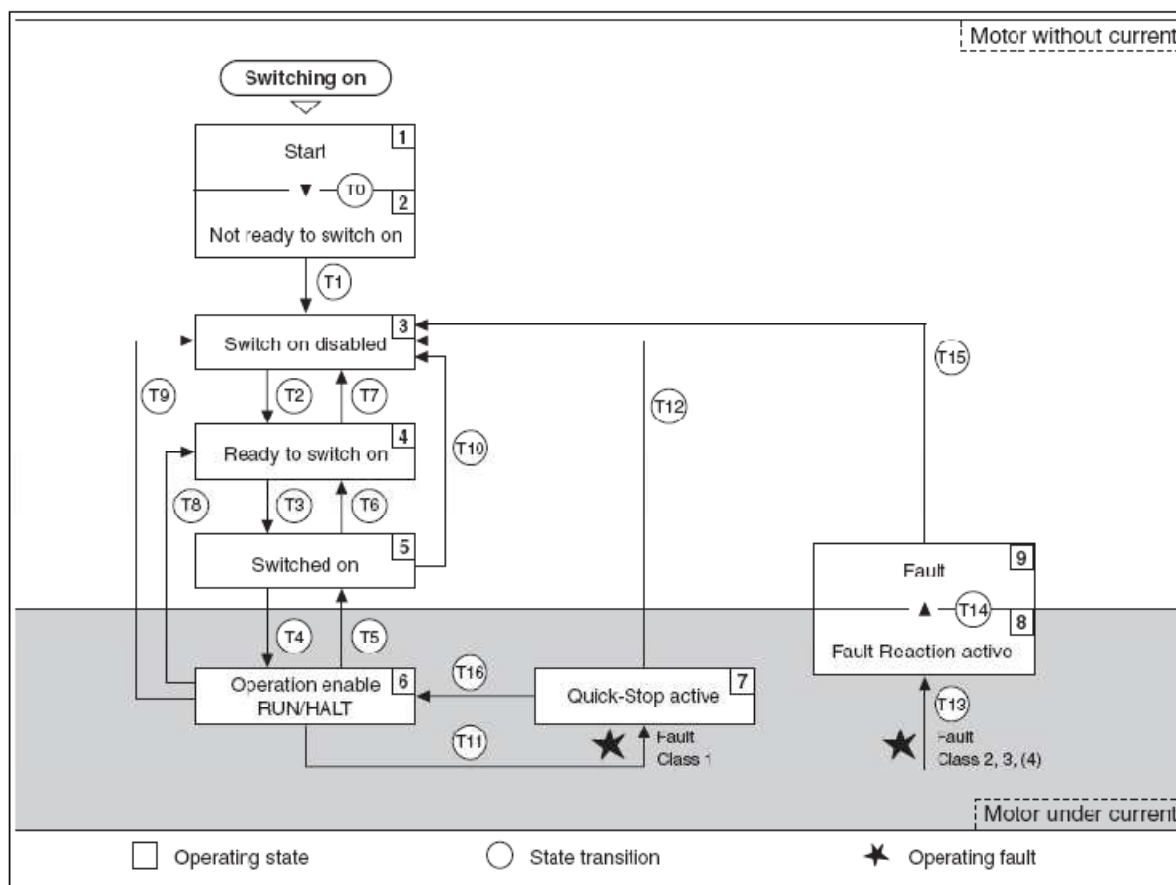
Period



Status machine

Before explaining the source code it is necessary to take a look to the state machine of the drive.

After switching on the drive a sequence of operating states is progressed through. If the drive is ok it goes automatically to state 4. When it is enabled and there is no error it changes to state 6.



Operating state	Description
1 Start	Controller supply voltage switched on Electronics are initialized
2 Not Ready To Switch On	The power stage is not ready to switch on
3 Switch On Disabled	Impossible to enable the power stage
4 Ready To Switch On	The power stage is ready to switch on.
5 Switched On	Power stage is switched on
6 Operation Enabled	Power stage is enabled Selected operating mode is active
7 Quick Stop Active	"Quick Stop" is being executed
8 Fault Reaction Active	Error response is active
9 Fault	Error response terminated Power stage is disabled



To enable the drive the bit 1 of the drive control byte must be true. We have access to it directly via the high byte of %MW204.

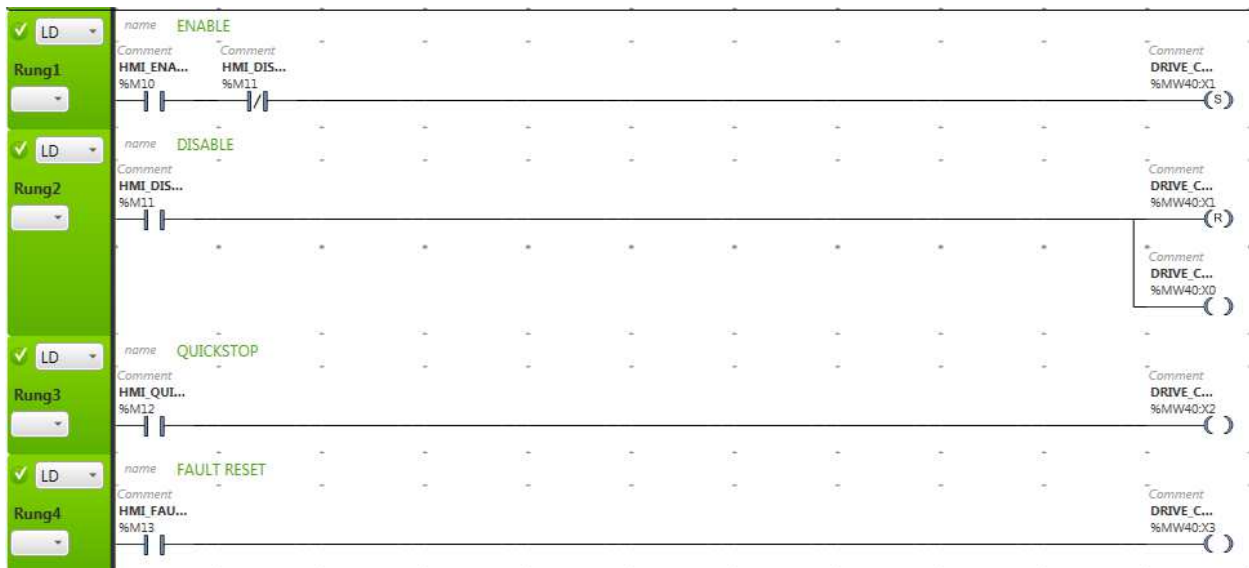
dmcontrol:

15	14	13	12	11	10	9	8
CU	CH	SH	-	FR	QS	EN	DS

8	DS	Disable power stage
9	EN	Enable power stage
10	QS	Executing a "Quick Stop"
11	FR	Execute "Fault Reset"
12	-	Reserved
13	SH	Execute "Halt"
14	CH	Clear "Halt"
15	CU	Resume operating mode interrupted by "Halt"



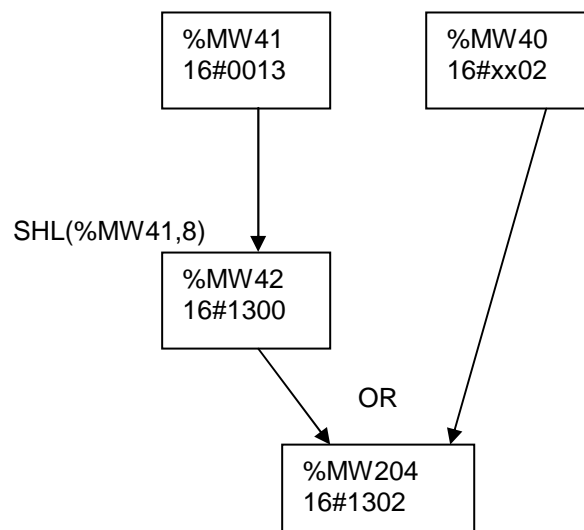
The source code corresponding to that is:



The bits %M10 (HMI_ENABLE), %M11 (HMI_DISABLE), %M12 (HMI_QUICKSTOP) and %M14 (HMI_FAULTRESET) manipulate the bits in %MW40 instead of %MW200. The reason is that drive control is only the high byte of %MW204, the low byte contains the mode control. Therefore the drive control is stored in %MW40 and the mode control in %MW41. In rung 16 both bytes are merged in %MW204:

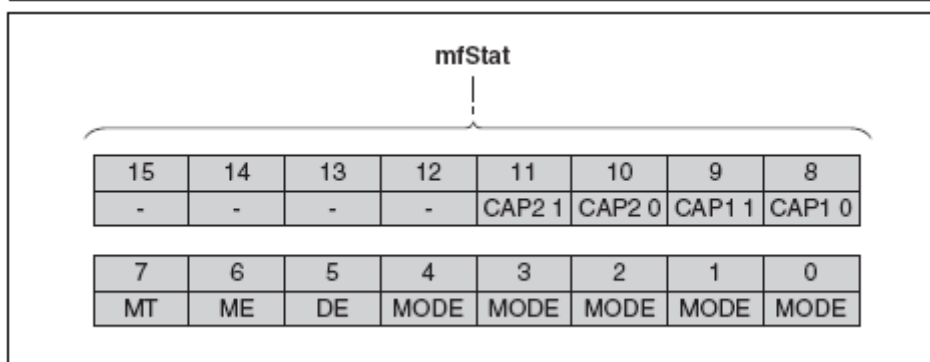
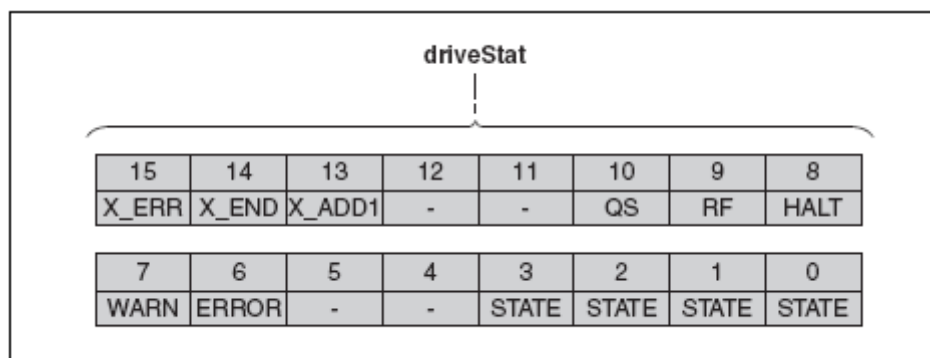


An example:





The drive status %MW104, mf status %MW105 and the motion status %MW106 contain the information of the actual state of the drive.



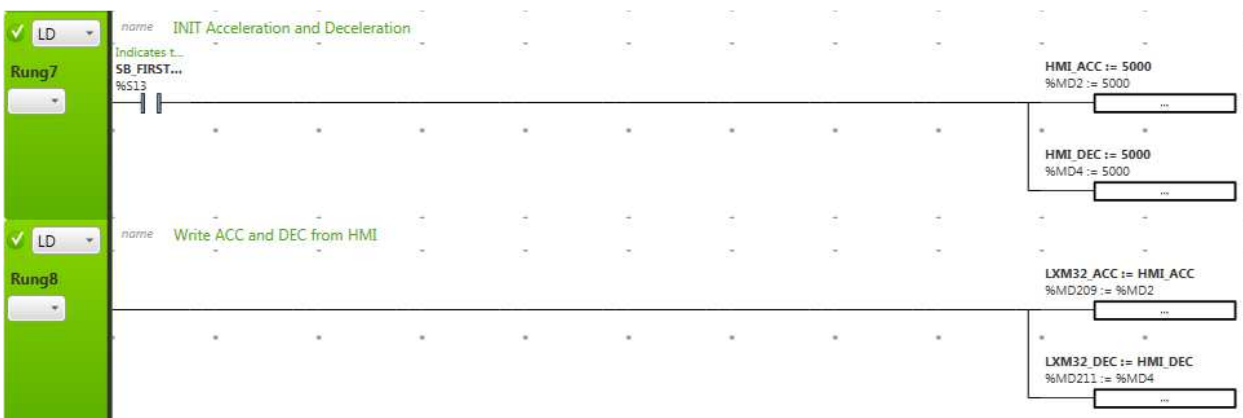
Bit	Meaning
0 ... 5	Reserved
6	MOTZ: Motor at a standstill
7	MOTP: Motor movement in positive direction
8	MOTN: Motor movement in negative direction
9 ... 10	Reserved
11	TAR0: Profile generator at standstill, reference velocity = 0
12	DEC: Profile generator decelerates
13	ACC: Profile generator accelerates
14	CNST: Profile generator moves at constant velocity
15	Reserved



The bit 0 .. 3 (state) in the drive status contains the number of the actual status, the bits 0 .. 4 (mode) in the mfstat contains the number of the actual operation mode. For the HMI only the data in those fields are needed.



In rung 7 and 8 the values for acceleration and deceleration are initialized with a default value.



Operation modes:

Bit	Name	Description
0 ... 4	MODE	Operating mode Value 01 _h : Profile Position Value 03 _h : Profile Velocity Value 04 _h : Profile Torque Value 06 _h : Homing Value 1F _h : Jog Value 1E _h : Electronic Gear
5 ... 6	ACTION	Operating mode-dependent
7	MT	Handshake via Mode Toggle



The operating modes can be changed during operation. For this purpose, the current process must be completed or explicitly canceled. The motor must be at a standstill.

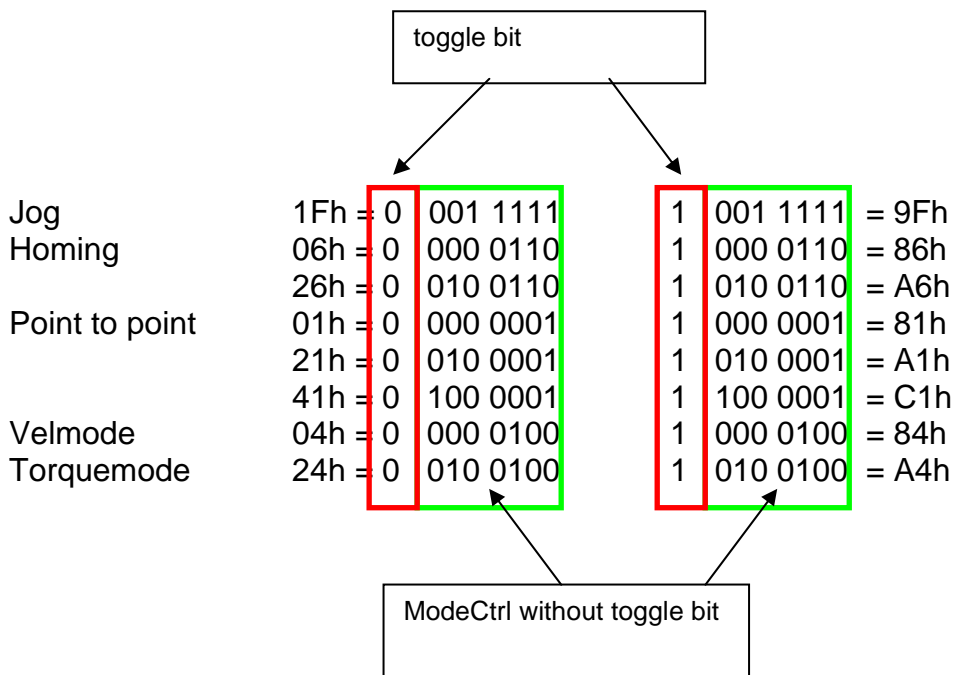
The master must enter the following values to activate an operating mode or to change reference values:

- Reference values, depending on desired operating mode
- Operating mode in "dmControl", bits 0 ... 4 (MODE).
- Action for this operating mode in bit 5 and bit 6 (ACTION)
- Toggle bit 7 (MT)

The following chapters describe the possible operating modes, functions and the corresponding reference values.



The left column shows the operation mode, the next column the entry in the mode control without the first bit, the toggle bit. This bit appears twice. Once in the *dmcontrol* and once in the *mfstat*, where it is mirrored. To start an operation mode its state has to change. When the toggle bit is false in the mode status it must be set to true in the mode control. After receiving the command it changes to true in the mode status. To activate the next operation mode it must be set to false. The hexadecimal values are:



Depending on the operation mode the meaning of the fields Ref32A and Ref32B is different. For example in velocity mode Ref32A contains the set speed, in homing it is the homing method.



Homing

The product manual contains detailed information about the operation mode.

Description In the operating mode Homing, a movement is performed to a defined position. This position is defined as the reference point.

The reference point also defines the zero point. The zero point is the point of reference for absolute movements in the operating mode Profile Position.

A movement can be made using one of 4 methods:

- Reference movement to a limit switch
In the case of a reference movement to a limit switch, a movement to the negative or positive limit switch is performed. After that, a movement is made to the switching point of the limit switch. From the switching point of the limit switch, a movement is made to the index pulse or to a parameterizable distance from the switching point. The position of the index pulse or the parameterizable distance from the switching point is the reference point.
- Reference movement to the reference switch
In the case of a reference movement to the reference switch, a movement to the reference switch is performed. After that, a movement is made to the switching point of the reference switch. From the switching point of the reference switch, a movement is made to the index pulse or to a parameterizable distance from the switching point. The position of the index pulse or the parameterizable distance from the switching point is the reference point.
- Reference movement to the index pulse
In the case of a reference movement to the index pulse, a movement is made directly to the closest index pulse. The position of the index pulse is the reference point.
- Position setting
In the case of position setting, the current motor position is set to a desired position value. The zero point is defined by the position value.

A reference movement must be completed for the new zero point to be valid. If the reference movement is interrupted, it must be started again.



Motors with multiturn encoder deliver a valid zero point after they are switched on.



Parameter name HMI menu HMI name	Description	Unit Minimum value Factory setting Maximum value	Data type R/W Persistent Expert	Parameter address via fieldbus
HmMethod	<p>Homing method</p> <p>1: LIMN with index pulse 2: LIMP with index pulse 7: REF+ with index pulse, inv., outside 8: REF+ with index pulse, inv., inside 9: REF+ with index pulse, not inv., inside 10: REF+ with index pulse, not inv., outside 11: REF- with index pulse, inv., outside 12: REF- with index pulse, inv., inside 13: REF- with index pulse, not inv., inside 14: REF- with index pulse, not inv., outside 17: LIMN 18: LIMP 23: REF+, inv., outside 24: REF+, inv., inside 25: REF+, not inv., inside 26: REF+, not inv., outside 27: REF-, inv., outside 28: REF-, inv., inside 29: REF-, not inv., inside 30: REF-, not inv., outside 33: Index pulse neg. direction 34: Index pulse pos. direction 35: Position setting</p> <p>Abbreviations: REF+: Search movement in pos. direction REF-: Search movement in pos. direction inv.: Invert direction in switch not inv.: Direction not inverted in switch outside: Index pulse / distance outside switch inside: Index pulse / distance inside switch</p> <p>Changed settings become active immediately.</p>	- 1 18 35	INT8 INT16 INT16 INT16 R/W - -	CANopen 6098:0 _h Modbus 6936 Profibus 6936 CIP 127.1.12



Starting the operating mode The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

Method	dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
Position setting	06 _h	-	As Hmp_setP
Reference movement	26 _h	As HMmethod	-

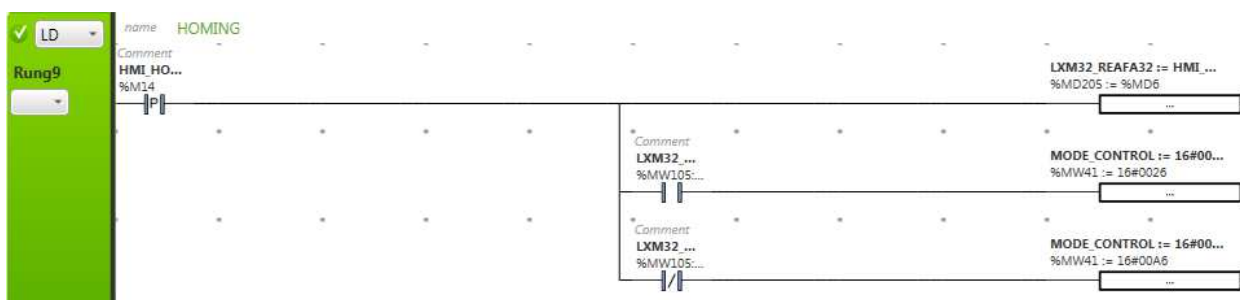
Status information The word "driveStat" provides information on the operating mode.

Bit	Name	Meaning
13	X_INFO	Reserved
14	X_END	0: Operating mode active 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Homing successful
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

The source code is very short. A rising edge on %M14 (*HMI_HOMING*) writes the homing type (*HMI_HOMINGTYPE*) into REF32A (%MD205). Depending on the toggle bit in *mfstat* (MW105:X7) the toggle bit in dmcontrol is set or reset.





Profile velocity

The product manual contains detailed information.

8.3.5 Operating mode Profile Velocity

Description In the operating mode Profile Velocity, a movement is made with a desired target velocity.

8.3.5.1 Parameterization

Overview The illustration below provides an overview of the adjustable parameters.

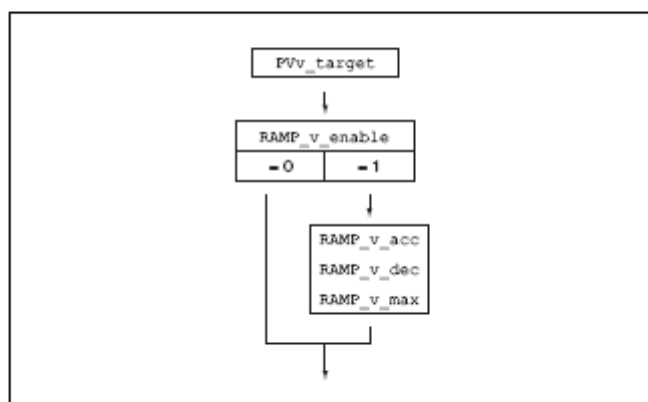


Figure 8.8 Overview of adjustable parameters

Setting the target velocity The target velocity is set by means of the parameter PVv_target.

► Set the target velocity with the parameter PVv_target.

Parameter name HMI menu HMI name	Description	Unit Minimum value Factory setting Maximum value	Data type R/W Persistent Expert	Parameter address via fieldbus
PVv_target	Target velocity for operating mode Profile Velocity The target velocity is limited to the setting in CTRL_v_max and RAMP_v_max. Changed settings become active immediately.	usr_v - 0 -	INT32 INT32 INT32 INT32 R/W - -	CANopen 60FF:0h Modbus 6938 Profibus 6938 CIP 127.1.13

Changing the motion profile for the velocity It is possible to change the settings of the motion profile for the velocity, see chapter 8.4.7.1 "Motion profile for the velocity".



Starting the operating mode

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
23 _h	As PVv_target	-

Status information

The word "driveStat" provides information on the operating mode.

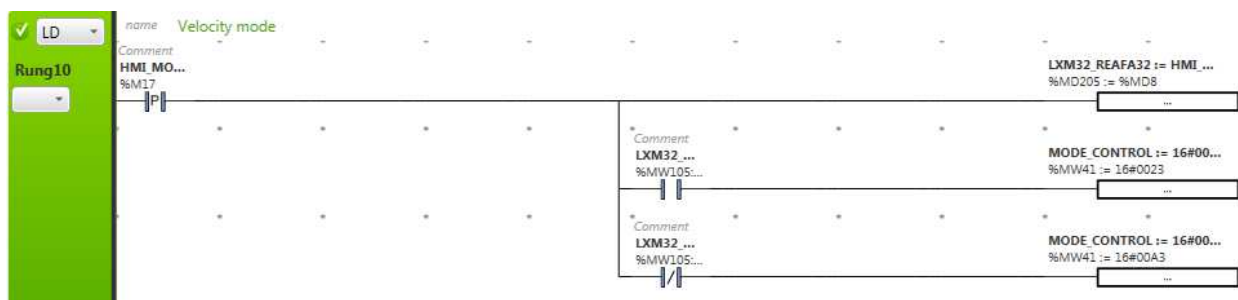
Bit	Name	Meaning
13	X_INFO	0: Target velocity not reached 1: Target velocity reached
14	X_END	0: Operating mode active 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode

The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

The velocity mode works in the same way. Now Ref32A contains the set speed.





Profile torque

Description In the operating mode Profile Torque, a movement is made with a desired target torque.

8.3.4.1 Parameterization

Overview The illustration below provides an overview of the adjustable parameters.

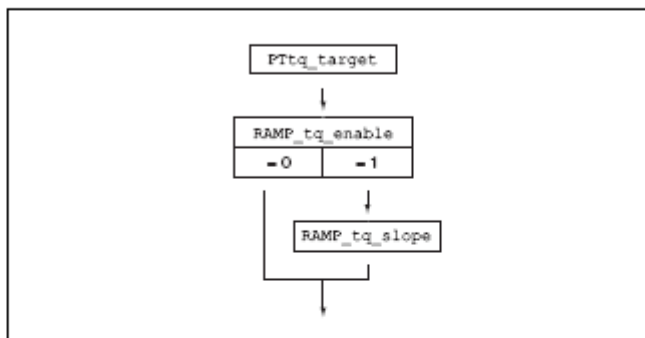


Figure 8.7 Overview of adjustable parameters

Setting the target torque The target torque is set by means of the parameter `PTtq_target`.

► Set the desired target torque with the parameter `PTtq_target`.

Parameter name HMI menu HMI name	Description	Unit Minimum value Factory setting Maximum value	Data type R/W Persistent Expert	Parameter address via fieldbus
PTtq_target	Target torque for operating mode Profile Torque 100.0 % correspond to the continuous stall torque <code>_M_M_0</code> . In increments of 0.1 %. Changed settings become active immediately.	% -3000.0 0.0 3000.0	INT16 INT16 INT16 INT16 R/W - -	CANopen 6071:0 _h Modbus 6944 Profibus 6944 CIP 127.1.16

Changing the motion profile for the torque It is possible to change the parameterization of the motion profile for the torque.

RAMP_tq_slope	Slope setting of the motion profile for torque 100.0 % correspond to the continuous stall torque <code>_M_M_0</code> . In increments of 0.1 %/s. Changed settings become active immediately.	%/s 0.1 10000.0 3000000.0	UINT32 UINT32 UINT32 UINT32 R/W per. -	CANopen 6087:0 _h Modbus 1620 Profibus 1620 CIP 106.1.42
---------------	---	------------------------------------	--	---



Starting the operating mode

The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
24 _h	As PItq_target	As RAMP_tq_slope

Status information

The word "driveStat" provides information on the operating mode.

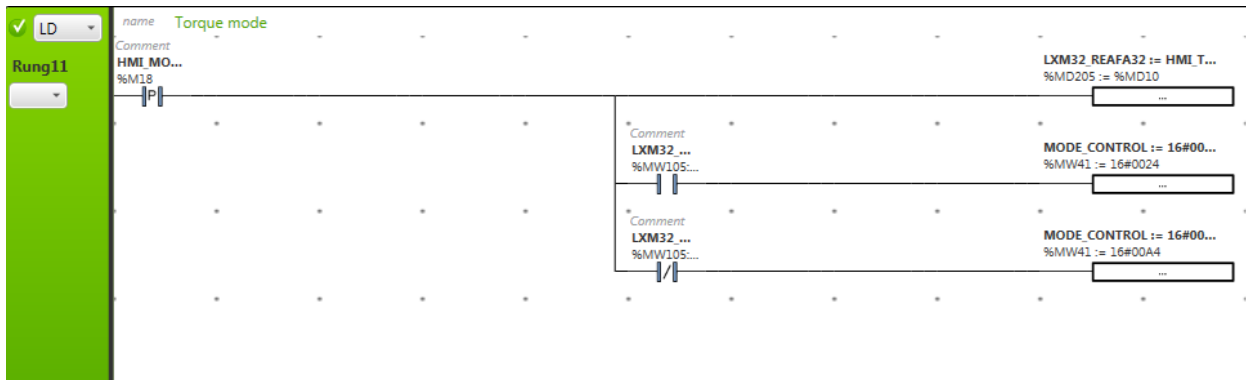
Bit	Name	Meaning
13	X_INFO	0: Target torque not reached 1: Target torque reached
14	X_END	0: Operating mode active 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode

The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

The torque mode works in the same way. Now Ref32A contains the target torque and REF32B contains the setting of the motion profile for torque.





Profile position (point to point)

The product manual contains detailed information.

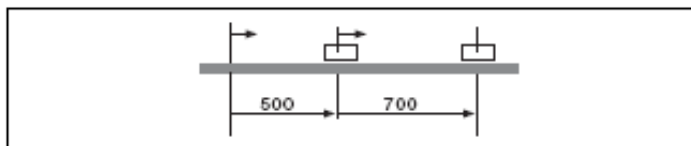
8.3.6 Operating mode Profile Position

Description In the operating mode Profile Position, a movement to a desired target position is performed.

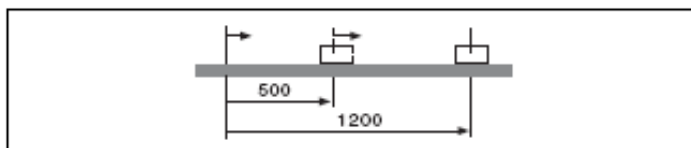
A movement can be made using one of 2 methods:

- Relative movement
- Absolute movement

Relative movement In the case of a relative movement, the movement is relative with reference to the previous target position or the current motor position.



Absolute movement In the case of an absolute movement, the movement is absolute with reference to the zero point.



A zero point must be defined with the operating mode Homing prior to the first absolute movement.



Starting the operating mode The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

Method	dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
absolute	01 _h	As PPv_target	As PPp_target
Relative with reference to the currently set target position	21 _h	As PPv_target	As PPp_target
Relative with reference to the current motor position	41 _h	As PPv_target	As PPp_target

Status information The word "driveStat" provides information on the operating mode.

Bit	Name	Meaning
13	X_INFO	0: Target position not reached 1: Target position reached
14	X_END	0: Operating mode active 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Target position reached
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error



The source code works also in the same way than the other operation modes. Now Ref32A contains the set speed and Ref32B the set position.





Jog mode (manual mode)

The product manual contains detailed information.

8.3.2 Operating mode Jog

Description In the operating mode Jog, a movement is made from the current motor position in the desired direction.

A movement can be made using one of 2 methods:

- Continuous movement
- Step movement

In addition, the product features 2 parameterizable velocities.

Continuous movement As long as the signal for the direction ("Jog Positive" or "Jog Negative") is available, a continuous movement is made in the desired direction.

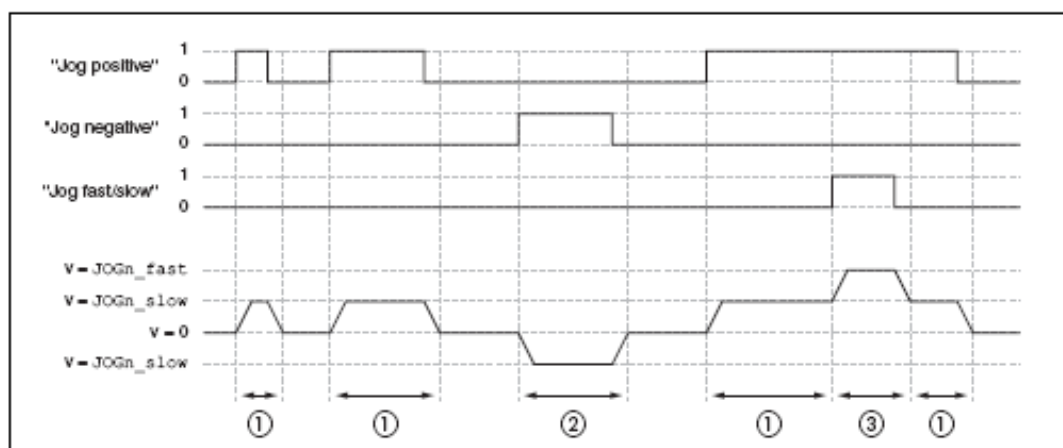


Figure 8.3 Continuous movement

- (1) Slow movement in positive direction
- (2) Slow movement in negative direction
- (3) Fast movement in positive direction

Step movement If the signal for the direction ("Jog Positive" or "Jog Negative") is available, a movement by a parameterizable number of user-defined units is made in the desired direction. After this movement, the motor stops for a defined time. Then a continuous movement is made in the desired direction.



Availability The operating mode is only available with the drive profile "Drive Profile Lexium".

Starting the operating mode The operating mode is started in the process data channel with the transmit data (master to slave) and "Mode Toggle". The value in the word "dmControl" is without set "Mode Toggle" bit.

dmControl MODE+ACTION	Reference value RefA32	Reference value RefB32
1F _h	As JOGactivate	-

Status information The word "driveStat" provides information on the operating mode.

Bit	Name	Meaning
13	X_INFO	Reserved
14	X_END	0: Operating mode active 1: Operating mode terminated
15	X_ERR	0: No error 1: Error

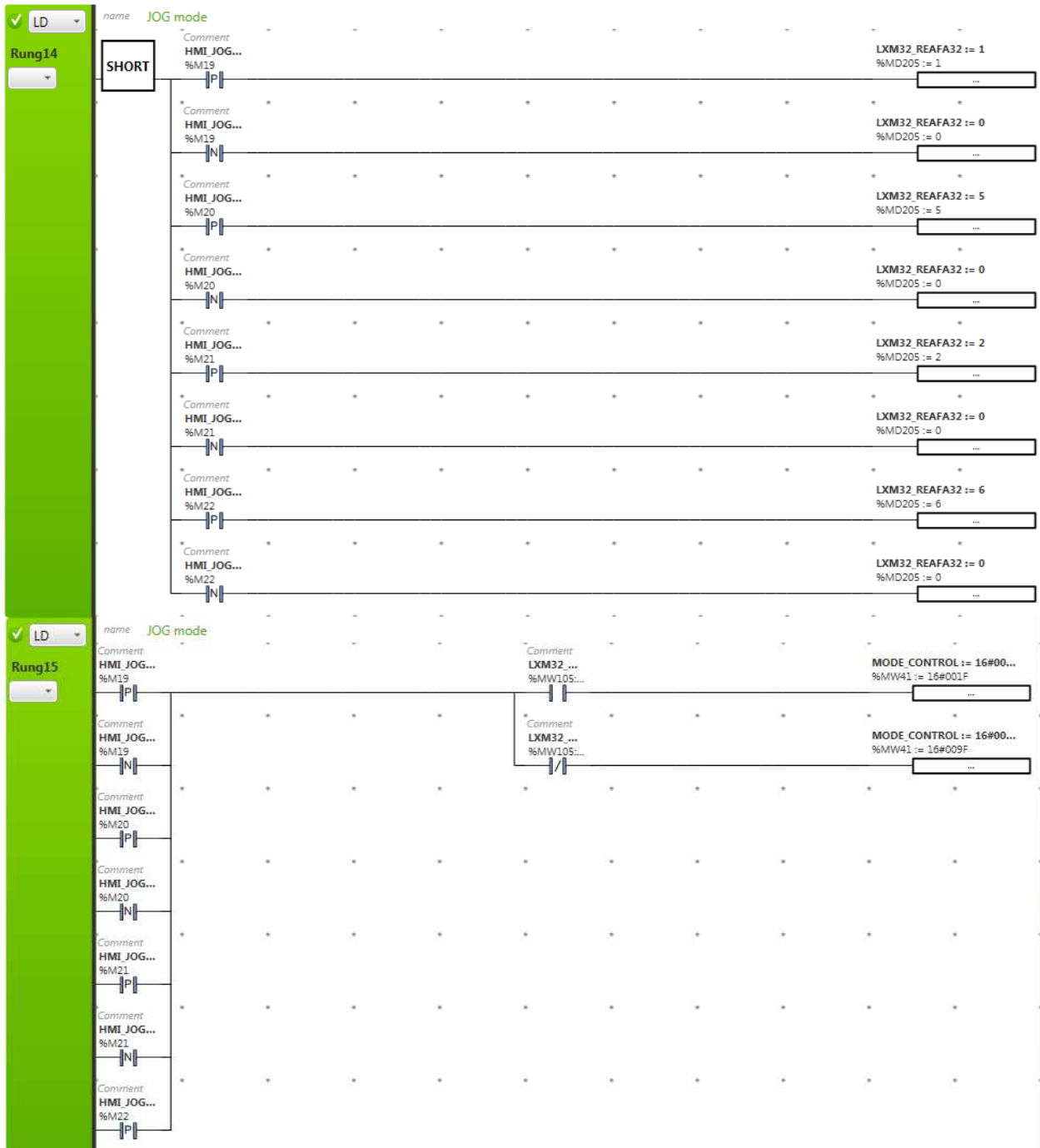
Terminating the operating mode The operating mode is terminated when the motor is at a standstill and one of the following conditions is met:

- Value 0 in RefA32
- Stop caused by "Halt" or "Quick Stop"
- Stop caused by an error

The bits %M19 to %M22 are the signals to control JOG mode. For the manual mode Ref32A has the meaning of parameter *JOGactivate* (Modbus 6930). The resulting bit combinations to written in REF32A are:

no movement	000	=	0
move right slow	001	=	1
move left slow	010	=	2
move right fast	101	=	5
move left fast	110	=	6

A rising edge of a push button starts a movement. The falling edge sets REF32A to zero, the movement stops.





Reading and writing parameter via the implicit parameter channel

In this program example the PLC request or change a parameter value of the slave via parameter channel. Every parameter can be addressed via parameter address and parameter subaddress.

Byte 0-3					Byte 4-7
Control (PCTRLxx)					Parameter value (PVxx)
AC	reserved	AT	Parameter sub address	Parameter address (PAXx)	

Byte 0-3: Control of parameter channel (action control + bus type + parameter sub address + parameter address)

Byte 4-7: parameter value

Control (PCTRLxx):

- The action control **AC** is placed in the first 4 bits (bit31 to 28) of a request or response.

The AT item contains the bus type that shall be used to decode the ID and the parameter sub address. (In this example “Modbus”)

31 30 29 28	27	26 25 24	23 22 21 20 19 18 17 16
AC	reserved		Parameter sub address

15 14 13 12	11	10 9 8	7 6 5 4 3 2 1 0
Parameter address			

- Structure of parameter channel byte 0 – 3.



Action control AC:

- The action control **AC** is used to indicate the action to be performed.
(Request, or change parameter; WORD or DWORD)

Action Control	Function	Positive response identifier	Negative response identifier
0	No job	0	7
1	Request (word)	1	7
1	Request (Dword)	2	7
2	Change (word)	1	7
3	Change (Dword)	2	7

- Action control AC and its possible response.

- The addressing type identifier AT is used to describe the coding of the values stored in parameter address and parameter sub address. The following coding types can be used:

Parameter coding	Bus system	Data
0 (000)	Profibus	Modbus address (Parameter sub address = 0)
1 (001)	CANopen	Parameter sub address Parameter address
2 (010)	Modbus	Modbus address (Parameter sub address = 0)

- Possible bus types.



Word no.	Name	Modbus address	PLC address
1	PCTRLms	6982	%MW200
2			%MW201
3	PVms	6984	%MW202
4			%MW203

xxx ms:
data from PLC to drive
(request to drive)

Word no.	Name	Modbus address	PLC address
1	PCTRLsm	6994	%MW100
2			%MW101
3	PVsm	6996	%MW102
4			%MW103

xxx sm:
data from drive to PLC
(response from drive)

