



# Siemens® TIA Portal LinMot® Library (SCL)

---

## **Supported LinMot Drives**

Profibus: E1430-DP-QN-xS, E1230-DP-UC, E1130-DP-xx  
Profinet: E1450-PN-QN-xS, E1250-PN-UC, C1450-PN-VS-xS, C1250-PN-XC-xS,  
C1150-PN-XC-xS, B1150-ML-xx at B8050-ML-PN

## **Supported Siemens® CPUs**

S7-1200, S7-1500

Axis Control, MC Commands & Configuration Modules

© 2018 NTI AG

This work is protected by copyright.

Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, microfilm, storing in an information retrieval system, not even for didactical use, or translating, in whole or in part, without the prior written consent of NTI AG.

LinMot® is a registered trademark of NTI AG.

#### Note

The information in this documentation reflects the stage of development at the time of press and is therefore without obligation. NTI AG reserves itself the right to make changes at any time and without notice to reflect further technical advance or product improvement.

NTI AG  
LinMot®  
Haerdlistrasse 15  
CH-8957 Spreitenbach

Tel.: +41 (0)56 419 91 91  
Fax: +41 (0)56 419 91 92  
Email: [office@LinMot.com](mailto:office@LinMot.com)  
Homepage: [www.LinMot.com](http://www.LinMot.com)

**Content**

<b>Content</b>	<b>3</b>
<b>Document Version</b>	<b>5</b>
<b>File List</b>	<b>5</b>
<b>Use Of The Library</b>	<b>6</b>
<b>Recommended Documentation</b>	<b>6</b>
<b>Compatibility To Older Library Versions</b>	<b>7</b>
<b>General</b>	<b>8</b>
<b>1 Hardware Configuration</b>	<b>9</b>
1.1 Hardware Configuration Profibus (E1430-DP-QN, E1230-DP-UC, E1130-DP-xx)	9
1.2 Hardware Configuration Profinet (E1450-PN-QN, E1250-PN-UC, C1450-PN-VS-xS, C1x50-PN-XC-xS)	11
1.2.1 B8050-ML-PN & M8050-ML-PN	13
1.3 Configuration LinMot Drive	14
<b>2 Data Types (UDT)</b>	<b>15</b>
2.1 Axis Communication	15
2.1.1 tstLM_Axis	15
2.2 Data Types of the Config Function Blocks	16
2.2.1 tstLM_CfgUPIDListEntry	16
2.2.2 tstLM_CfgCTEntry	16
<b>3 Function Blocks</b>	<b>17</b>
3.1 Overview and Dependencies	17
3.2 IO and Axis Control	18
3.2.1 Profinet: LMct_RdAxisCom_PN	18
3.2.2 Profinet: LMct_WrAxisCom_PN	19
3.2.3 Profibus: LMct_RdAxisCom_DP_PN	20
3.2.4 Profibus: LMct_WrAxisCom_DP_PN	21
3.2.5 LMct_AxisControl	22
3.3 MC Function Blocks	23
3.3.1 LMmt_MoveAbs & LMmt_MoveAbsJ	23
3.3.2 LMmt_MoveRel & LMmt_MoveRelJ	24
3.3.3 LMmt_StartCTCommand	25
3.3.4 LMmt_Stop	26
3.3.5 LMmt_WriteLivePar	27
3.3.6 LMmt_GenericMC	28
3.3.7 LMmt_IncrActPos & LMmt_IncrActPosJ	29
3.4 MC Function Blocks (Advanced)	30
3.4.1 LMav_Mod16BitCTPar	30
3.4.2 LMav_Mod32BitCTPar	31
3.4.3 LMav_RunCurve	32
3.4.4 LMav_MoveBestehorn	33
3.4.5 LMav_MoveSin	34
3.4.6 LMav_PVStream	35
3.4.7 LMav_PVStream	36
3.5 MC Function Blocks Force Control	37
3.5.1 LMfc_ChangeTargetForce	37
3.5.2 LMfc_GoToPosForceCtrlHL	38
3.5.3 LMfc_GoToPosForceCtrlLL	39
3.5.4 LMfc_GoToPosRstForceCtrl	40
3.6 MC Function Blocks CAM	41
3.6.1 LMcm_CAMCtrl	41

3.6.2 LMcm_LoadCAM1_mc1130h .....	42
3.6.3 LMcm_LoadCAM2_mc1230h .....	43
3.7 Config Function Blocks .....	44
3.7.1 LMcf_ParaAccess .....	45
3.7.2 LMcf_GetModUPIDList .....	46
3.7.3 LMcf_WriteUPIDList.....	47
3.7.4 LMcf_StopStartDefault.....	48
3.7.5 LMcf_CurveAccess .....	49
3.7.6 LMcf_CTAccess .....	50
3.7.7 LMcf_GetErrorTxt (FB206) .....	51
<b>4 Function Blocks for Rotary Motors connected to LinMot Drives .....</b>	<b>52</b>
4.1 Parameters & Conversions .....	52
4.1.1 GearFactor (Data type REAL).....	52
4.1.2 UnitScale (Data type REAL) .....	52
4.1.3 ModuloFactor (Data type DInt).....	53
4.1.4 Examples .....	53
4.2 Function Blocks .....	54
4.2.1 LMct_RotaryMotorControl .....	54
4.2.2 LMmt_RotMoveAbs.....	57
4.2.3 LMmt_RotMoveRel .....	58
4.2.4 LMmt_RotIncrActAngle .....	59
4.2.5 LMmt_RotMoveInfPos.....	60
4.2.6 LMmt_RotMoveInfNeg .....	61
<b>5 Error Descriptions.....</b>	<b>62</b>
5.1 Error Codes of the Axis Control Function Block.....	62
5.2 Error ID's of the MC Function Blocks .....	62
5.3 Error ID's of the Config Function Blocks .....	62
<b>6 Example Projects .....</b>	<b>63</b>
6.1 CPU S7-1500 (Profinet), LinMot_TIA_SCL_S71500_Example_Project_PN_...zip .....	64
6.2 CPU S7-1500 (Profibus), LinMot_TIA_SCL_S71500_Example_Project_DP_...zip .....	65
6.3 CPU S7-1200 (Profinet), LinMot_TIA_SCL_S71200_Example_Project_PN_...zip .....	66
<b>Notes .....</b>	<b>67</b>
<b>Contact &amp; Support.....</b>	<b>68</b>

**Document Version**

Version	Date	Author	Library version	Description
1.0.0	20 Jan 2014	fj	1.0.0	Initial version
1.0.1	20 Mai 2014	fj	1.0.1	LMct_AxisControl: optimized
1V2	11 Jun 2015	fj	1V2	Function blocks for rotary motors added Library and example projects upgraded to TIA13 SP1 Update 3
1V3	11 Aug. 2016	fj	1V3	Added jerk limited function blocks (C1200, C1400 & E1400V2 series only). New Error ID's for config function blocks (LMcf_) LMcf_ParaAccess: optimized execution speed Updated example projects
1V4	18 Sep 2017	fj	1V4	LMav_MoveSin: MC Header was wrong > fixed Updated documentation Updated example projects
1V5	27 Nov 2018	fj	1V5	LMfc_GoToPosForceCtrlHL: MC Header was wrong > fixed Updated library to TIA V15 Updated examples to TIA V15

**File List**

The package of which this documentation is part of should contain the following files and folders:

\Library\

- LinMot\_TIA\_SCL\_Library\_1V5\_V15.zal15 (global library)

\ExampleProjects\

- LinMot\_TIA\_SCL\_S71200\_Example\_Project\_PN\_....zap15
- LinMot\_TIA\_SCL\_S71500\_Example\_Project\_PN\_....zap15
- LinMot\_TIA\_SCL\_S71500\_Example\_Project\_DP\_....zap15

\GSD\

- GSD files for LinMot Profibus drives

\GSDML\

- GSDML (xml) files for LinMot Profinet drives and bus modules

\0185-0007-D\_1V5\_MA\_Siemens\_TIA\_LinMot\_Library\_SCL.pdf

\0185-0007-E\_1V5\_MA\_Siemens\_TIA\_LinMot\_Library\_SCL.pdf

**Note:**

The most recent GSD and GSDML files are always part of the LinMot-Talk software.

See chapters:

1.1 Hardware Configuration Profibus (E1430-DP-QN, E1230-DP-UC, E1130-DP-xx)

1.2 Hardware Configuration Profinet (E1450-PN-QN, E1250-PN-UC, C1450-PN-VS-xS, C1x50-PN-XC-xS)

### Use Of The Library

The presented library for Siemens® TIA Portal provides function blocks to control LinMot drives over Profibus or Profinet interface.

The library is provided by NTI AG / LinMot free of charge with no warranty for updates.

Also, LinMot accepts no liability for damages that may be caused by using this library.

**Supported Drives:** E1430-DP-QN, E1230-DP-UC, E1130-DP-xx  
E1450-PN-QN, E1250-PN-UC, C1450-PN-VS-xS, C1250-PN-XC-xS  
C1150-PN-XC-xS  
B1150-ML-xx connected to B8050-ML-PN or MB8050-ML-PN

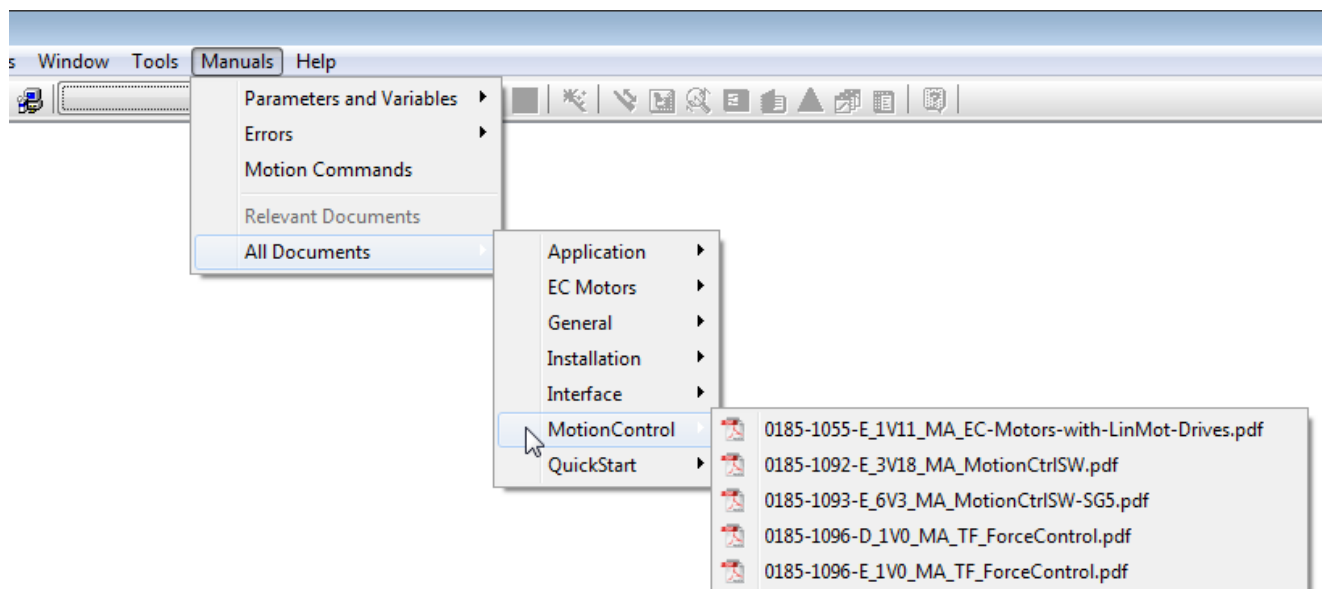
**Classification:** [ ] LinMot internally  
[x] Customer manual

### Recommended Documentation

Reading the following user manuals is essential to understand the communication between the PLC and the LinMot drive.

The manuals are included in the LinMot-Talk software (Menu *Manuals* -> *All Documents*, or *Relevant Documents* if logged into drive) or can be downloaded from the LinMot eCatalogue (Search for document reference): <http://shop.linmot.com>

Name Manual	Document Reference
LinMot-Talk	0185-1059
Motion Control Software (E11x0, B11x0)	0185-1092
Motion Control Software (E14x0, E12x0, C1x00)	0185-1093
Profibus (E1430, E1230, E1130)	0185-1089
Profinet (E1450, E1250, C1x50)	0185-1090
Profinet (B8050-ML-PN, MB8050-ML-PN)	0185-1087
Drive Configuration over Fieldbus (E11x0, B11x0)	0185-1072
Drive Configuration over Fieldbus SG4 (B1150-ML)	0185-1073
Drive Configuration over Fieldbus SG5-SG7 (E1400, E1200, C1x00)	0185-1074



## Compatibility To Older Library Versions

The recent library is compatible to older version.

In case of problems please contact our support.

In chapter Contact & Support you can find the contact details

## General

The LinMot drives can be connected over Profinet or Profibus interface to a Siemens CPU. This library is presented to simplify the integration of the drive into the PLC program and to show general control methods.

The drives supported by each function block can be found in the detailed description of the function blocks.

The package includes the following function blocks and data types:

### Axis Control:

- LMct\_RdAxisCom\_PN & LMct\_RdAxisCom\_DP\_PN
- LMct\_WrAxisCom\_PN & LMct\_WrAxisCom\_DP\_PN
- LMct\_AxisControl

### MC Function Blocks:

- LMmt\_MoveAbs & LMmt\_MoveAbsJ
- LMmt\_MoveRel & LMmt\_MoveRelJ
- LMmt\_StartCTCommand
- LMmt\_Stop
- LMmt\_WriteLivePar
- LMmt\_GenericMC
- LMmt\_IncrActPos & LMmt\_IncrActPosJ

### Config Function Blocks:

- LMcf\_ParaAccess
- LMcf\_GetModUPIDList
- LMcf\_WriteUPIDList
- LMcf\_StopStartDefault
- LMcf\_CurveAccess
- LMcf\_CTAcess
- LMcf\_GetErrorTxt

### MC Function Blocks (Advanced):

- LMav\_Mod16BitCTPar
- LMav\_Mod32BitCTPar
- LMav\_PVStream
- LMav\_PVStream
- LMav\_RunCurve (B Serie Drives nur mit freigeschalteter Technologiefunktion: Kurven)
- LMav\_MoveBesthorn
- LMav\_MoveSin

### Function Blocks for rotary motors:

- LMct\_RotaryMotorControl
- LMmt\_RotMoveAbs
- LMmt\_RotMoveRel
- LMmt\_RotIncrActAngle
- LMmt\_RotMoveInfPos
- LMmt\_RotMoveInfNeg

### MC Function Blocks Force Control

- LMfc\_ChangeTargetForce
- LMfc\_GoToPosForceCtrlHighLim
- LMfc\_GoToPosForceCtrlLowLim
- LMfc\_GoToPosRstForceCtrl

### Data Types (UDT):

- tstLM\_Axis
- tstLM\_CfgCTEntry
- tstLM\_CfgUPIDListEntry
- tstLM\_CfgCTPresenceList

### MC Function Blocks CAM:

- LMcm\_CAMCtrl
- LMcm\_LoadCAM1\_mc1130
- LMcm\_LoadCAM2\_mc1230

The function blocks are multi instance capable.

The library was created using *TIA Portal V15 Update 2* using **SCL**.

### CPUs used:

- CPU 1212C AC/DC/Rly (Firmware V3.0)
- CPU 1516-3 PN/DP (Firmware V1.1.2)



## 1 Hardware Configuration

### 1.1 Hardware Configuration Profibus (E1430-DP-QN, E1230-DP-UC, E1130-DP-xx)

The following procedure can be used for both CPU S7-1200 and CPU S7-1500.  
Consider the notes at the end of this chapter.

- First of all, install the GSD file for the drive.  
(TIA Portal → Options → Install general station description file (GSD))

The required GSD file can be found by default in the following folder:

C:\Programme\LinMot\LinTalk X.X Build XXXXXXXX\Firmware\Profibus\GSD\

- Drag and drop the desired drive from the catalogue to the net view.

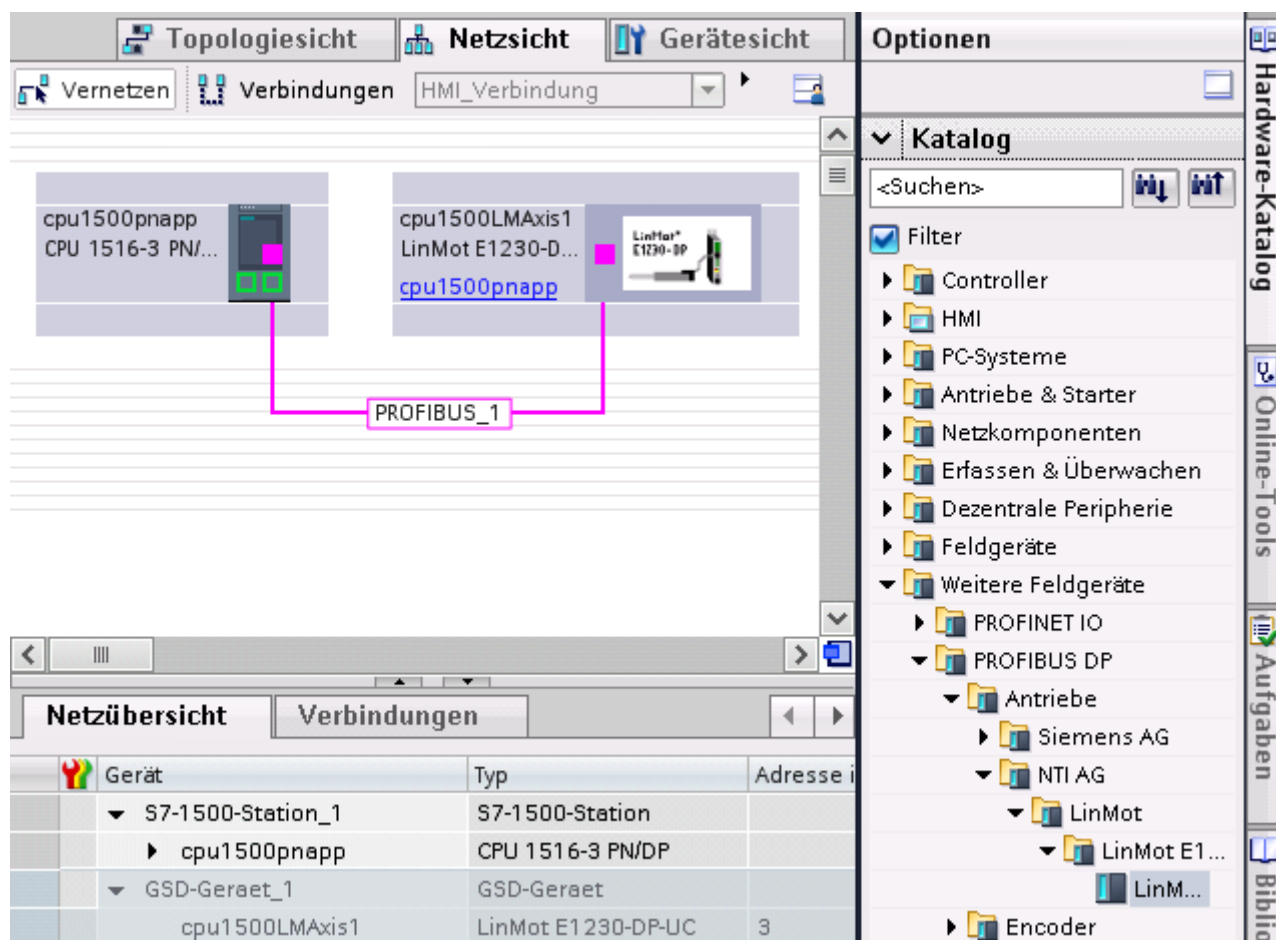


Figure 1: TIA Portal network view, insert LinMot E1230-DP-UC Drive

- Afterwards insert the required modules into the available slots as shown in Figure 2 (Slots 1 – 8), and assign a name (e.g. LMAxis1\_...).

Drag and drop modules by mouse.

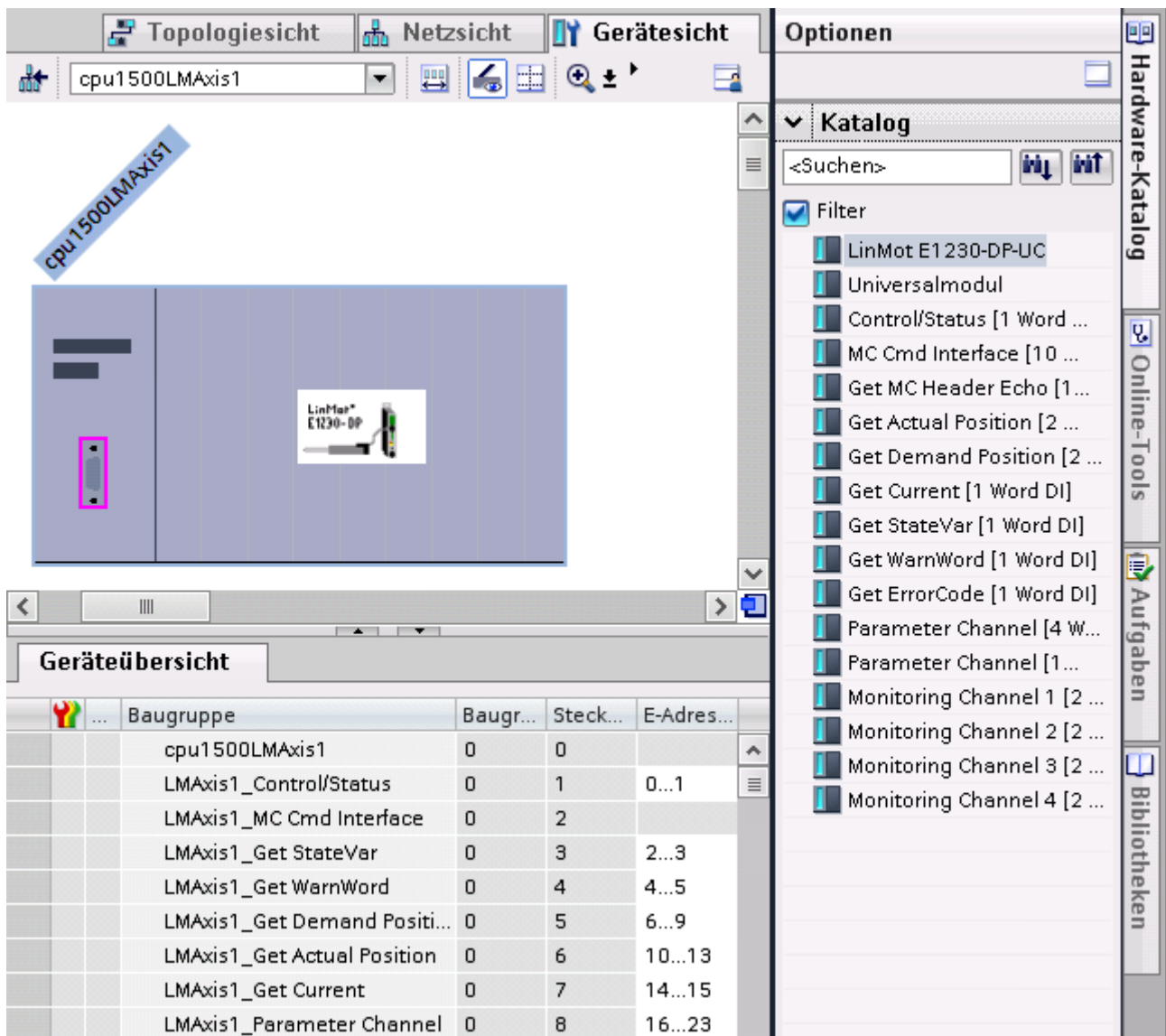


Figure 2: TIA Portal device view, Insert modules



**Note:**

For CPU S7-1200 and S7-1500 the TIA Portal creates a system type *HW\_MODULE* for each module that can be passed to the read and write function blocks.

See chapters:

3.2.3 Profibus: LMct\_RdAxisCom\_DP\_PN

3.2.4 Profibus: LMct\_WrAxisCom\_DP\_PN

**1.2 Hardware Configuration Profinet (E1450-PN-QN, E1250-PN-UC, C1450-PN-VS-xS, C1x50-PN-XC-xS)**

The following procedure can be used for both CPU S7-1200 and CPU S7-1500.  
Consider the notes at the end of this chapter.

- First of all install the GSD file for the drive.  
(TIA Portal → Options → Install general station description file (GSD))

The required GSDML file can be found by default in the following folder:  
C:\Programme\LinMot\LinTalk X.X Build XXXXXXXX\Firmware\Profinet\GSDML\

- Drag and drop the desired drive from the catalogue to the net view.

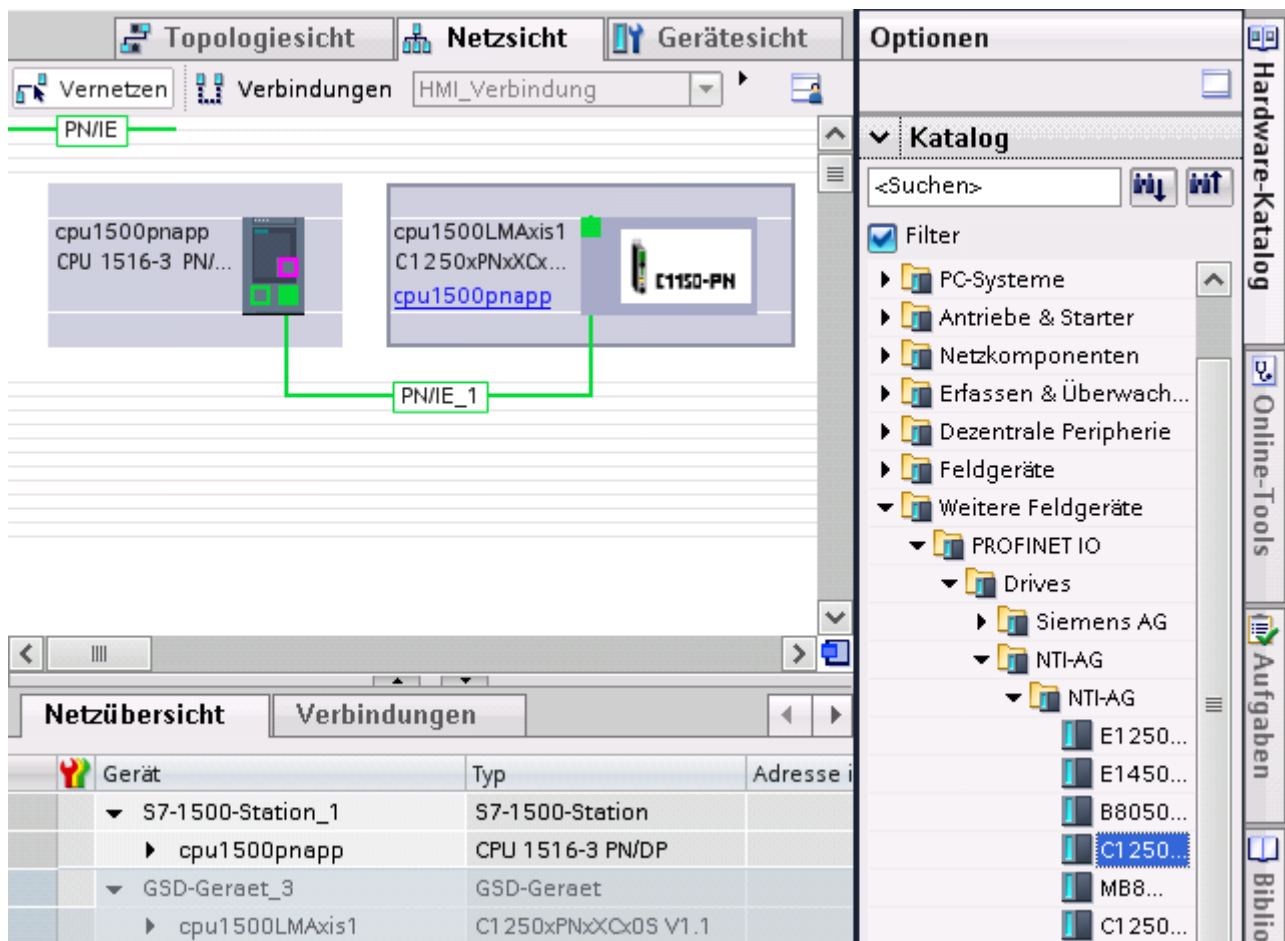


Figure 3: TIA Portal network view, Insert LinMot C1250-PN-XC-xS Drive

- Afterwards (if not already there) insert the „Default IO mapping with Config“ module as shown in Figure 4 into slot 1 and assign a name (e.g. LMAxis1\_IO).

Drag and drop modules by mouse.

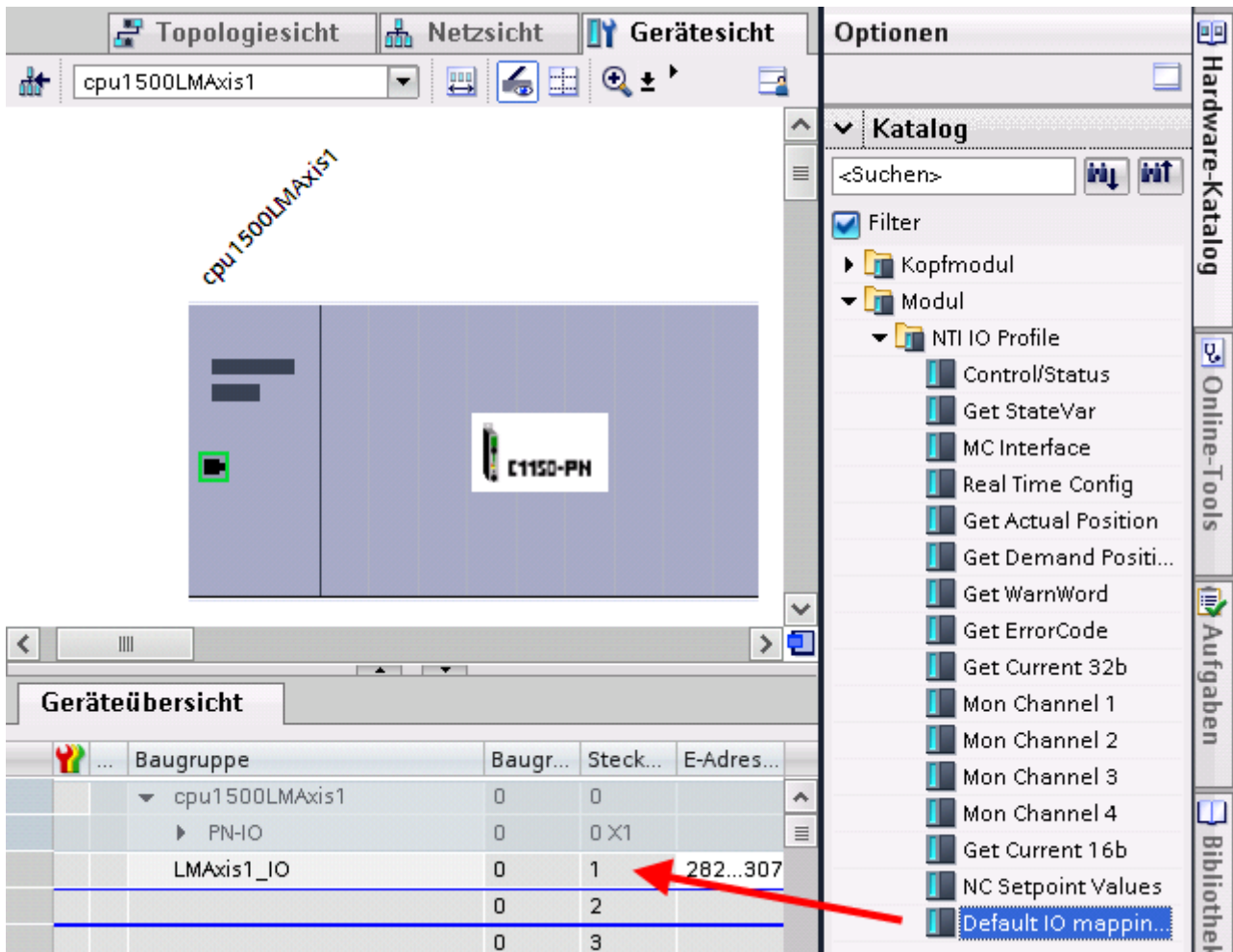


Figure 4: TIA Portal device view, Insert "Default IO mapping with Config" module



**Note:**

Assign a name and an IP to the device according to the requirements of your application.



**Note:**

For CPU S7-1200 and S7-1500 the TIA Portal creates a system type *HW\_MODULE* for each module that can be passed to the read and write function blocks.

See chapters:

3.2.1 Profinet: LMct\_RdAxisCom\_PN

3.2.2 Profinet: LMct\_WrAxisCom\_PN

**1.2.1 B8050-ML-PN & M8050-ML-PN**

Using one of the bus modules (with connected B1150-ML-XX drives) the drives are dropped to the available slots (max. 8). The modules (Axis 1, Axis 2, ...) contain the same data as the „Default IO mapping with Config“ module and are therefore compatible.

- The required GSDML files can be found by default in the following folder: C:\Programme\LinMot\LinTalk X.X Build XXXXXXXX\Firmware\Profinet\_ML\GSDML\

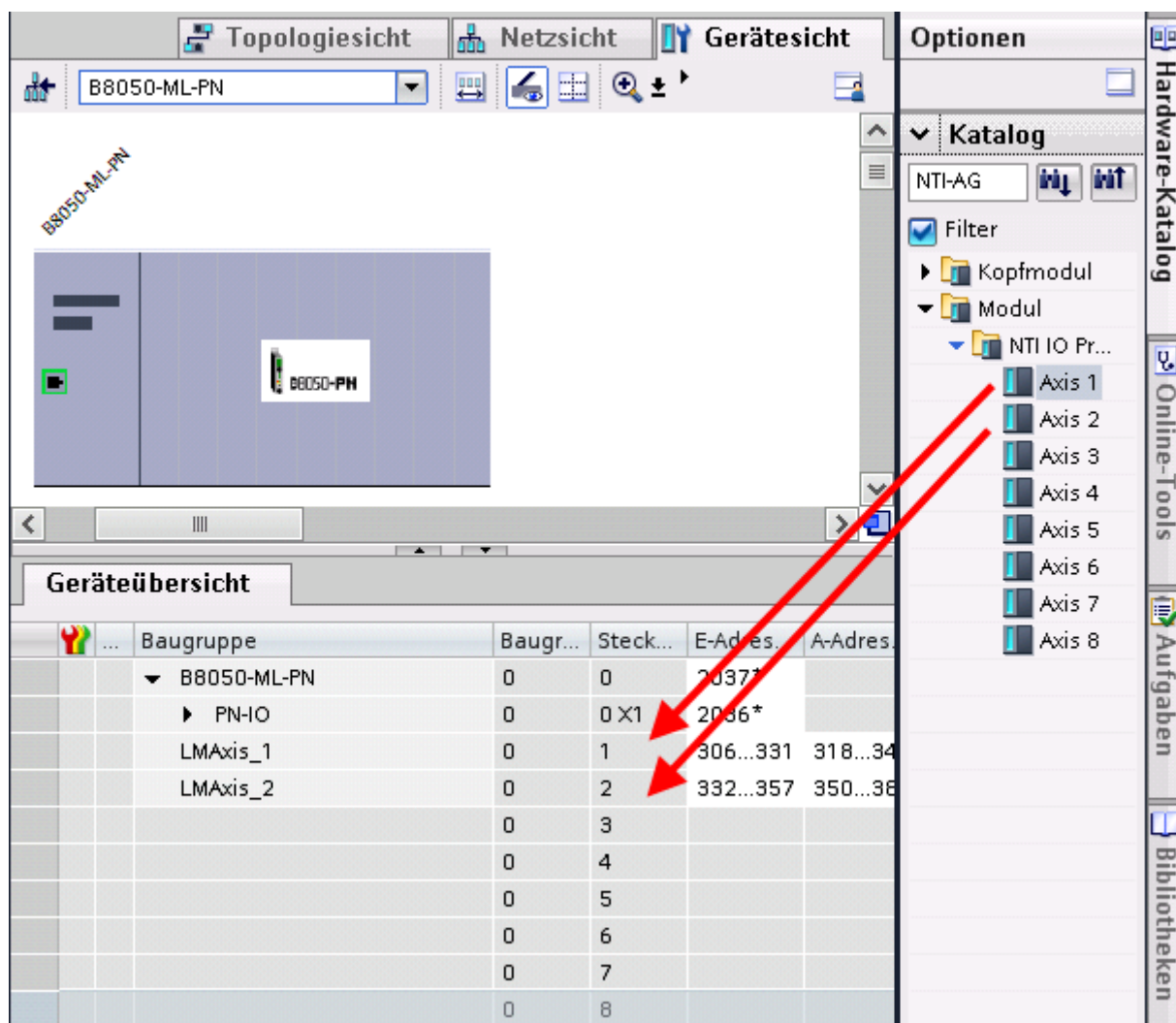


Figure 5: TIA Portal device view. Insert modules for the individual axis drives (B1150-ML-XX)

### 1.3 Configuration LinMot Drive

The LinMot drive is configured with LinMot-Talk:

<http://www.linmot.com/download/linmot-talk-drive-configuration/linmot-talk-6/>

It is assumed that the motor attached to the drive is already configured with the motor wizard.

#### Profibus Drives:

The only setting that must be done on the drive according to the Profibus interface is setting the node address. By default it is set with the rotary hex switches S1 (ID High) and S2 (ID Low) on the front of the drive.

Alternatively it can be set with the parameter "Node Address Parameter Value" (UPID 2076h). This requires the parameter "Node Address Selection" (UPID 206Ch) to be set to "On".

All other Profibus interface parameters are left to their default values.

#### Profinet Drives:

The Profinet parameters in the drive are left to default values. The name and IP address is set using the TIA Portal.



#### Note:

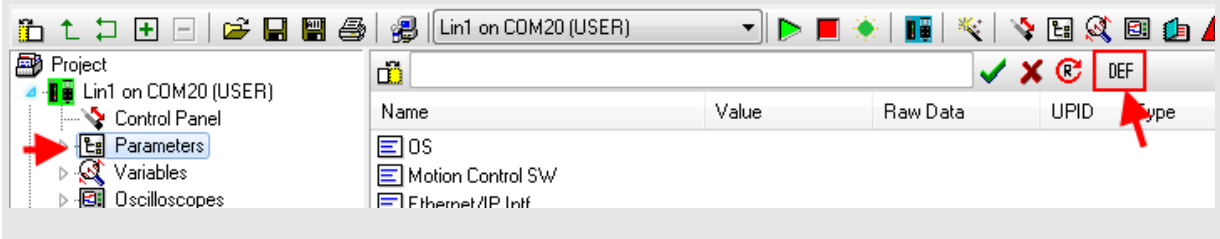
In case of doubt reset the drive to default values and afterwards configure the motor with the motor wizard. Be sure to save your actual configuration before this step!

#### Set drive to default values (E1xx0 series controllers only):

- Remove 24V supply from drive.
- Set both rotary hex switches (S1 and S2) to F.
- Restore 24V power to drive. The ERROR and WARN LED's should blink alternately.
- Set both rotary hex switches (S1 and S2) to 0.
- Wait until EN and WARN led blink together.
- Remove and restore 24V power to drive.

#### Set drive to default values (C1x00, E1200 und E1400 drives only):

Using LinMot-Talk the parameters can be set to default values by pressing the button "DEF"





## 2 Data Types (UDT)

### 2.1 Axis Communication

#### 2.1.1 tstLM\_Axis

The UDT tstLM\_Axis contains all data required for a proper communication between the function blocks and the drive.

tstLM_Axis			
		Name	Datentyp
1		PlcToDrv	Struct
2		ControlWord	Word
3		MCHeader	Word
4		MCPaWord0	Word
5		MCPaWord1	Word
6		MCPaWord2	Word
7		MCPaWord3	Word
8		MCPaWord4	Word
9		MCPaWord5	Word
10		MCPaWord6	Word
11		MCPaWord7	Word
12		MCPaWord8	Word
13		MCPaWord9	Word
14		MCPaWord10	Word
15		MCPaWord11	Word
16		MCPaWord12	Word
17		MCPaWord13	Word
18		CfgControlWord	Word
19		CfgIndexOut	Word
20		CfgValueOut	DWord
21		DrvToPlc	Struct
22		StatusWord	Word
23		StateVar	Word
24		WarnWord	Word
25		ComDemandPosition	DInt
26		ComActualPosition	DInt
27		ComActualCurrent32	DInt
28		ComActualCurrent16	Int
29		CfgStatusWord	Word
30		CfgIndexIn	Word
31		CfgValueIn	DWord
32		CommandRunning	Bool
33		CommandAborted	Bool
34		ConfigChannelBusy	Bool
35		AxisName	String
36		AxisNr	UInt
37		AxisCtrlType	String

Figure 6: tstLM\_Axis



**Note:**

Additional information regarding the meaning of the data can be found in the user manual "Motion Control Software". (Recommended Documentation)

### 2.2 Data Types of the Config Function Blocks

#### 2.2.1 tstLM\_CfgUPIDListEntry

The UDT tstLM\_CfgUPIDListEntry contains the number and the value of a LinMot parameter.

Can be used in a DB (as array) with the following config function blocks:

- LMcf\_GetModUPIDList
- LMcf\_WriteUPIDList

tstLM_CfgUPIDListEntry		
	Name	Datentyp
1	UPID	UInt
2	UPIDValue	UDInt

Figure 7: tstLM\_CfgUPIDListEntry

#### 2.2.2 tstLM\_CfgCTEntry

The UDT tstLM\_CfgCTEntry contains all data for one entry line of the command table on the drive.

Used in the following config function block:

- LMcf\_CTAccess

tstLM_CfgCTEntry		
	Name	Datentyp
1	CTID	Word
2	BlockSize	Word
3	▼ DataBlock	Array [0..15] of DWord
4	■ DataBlock[0]	DWord
5	■ DataBlock[1]	DWord
6	■ DataBlock[2]	DWord
7	■ DataBlock[3]	DWord
8	■ DataBlock[4]	DWord
9	■ DataBlock[5]	DWord
10	■ DataBlock[6]	DWord
11	■ DataBlock[7]	DWord
12	■ DataBlock[8]	DWord
13	■ DataBlock[9]	DWord
14	■ DataBlock[10]	DWord
15	■ DataBlock[11]	DWord
16	■ DataBlock[12]	DWord
17	■ DataBlock[13]	DWord
18	■ DataBlock[14]	DWord
19	■ DataBlock[15]	DWord

Figure 8: tstLM\_CfgCTEntry



#### Note:

Additional information can be found in the user manual "LinMot drive Configuration over Fieldbus Interfaces". (Recommended Documentation)



### 3 Function Blocks

#### 3.1 Overview and Dependencies



### 3.2 IO and Axis Control

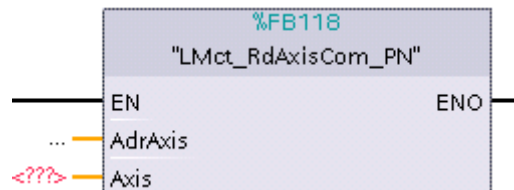
#### 3.2.1 Profinet: LMct\_RdAxisCom\_PN

This function block LMct\_RdAxisCom\_PN reads the input data and puts them to the axis reference (tstLM\_Axis).

Should be called at the beginning of the PLC cycle or at least before all other library function blocks.

Supported drives:

- All LinMot drives with Profinet interface



Inputs		
Name	Data type	Description
AdrAxis	HW_SUBMODULE	HW_SUBMODULE „Default IO mapping with Config“ module (respectively „Axis_x“ when using B8050-ML-PN & MB8050-ML-PN)
Axis	tstLM_Axis	Axis reference (IN_OUT)

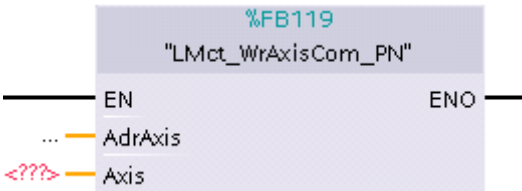
3.2.2 Profinet: LMct\_WrAxisCom\_PN

This function block LMct\_WrAxisCom\_PN writes the output data of the axis reference (tstLM\_Axis), prepares and writes them to the output addresses.

Should be called at the end of the PLC cycle or at least after all other library function blocks.

Supported drives:

- All LinMot drives with Profinet interface



Inputs		
Name	Data type	Description
AdrAxis	HW_SUBMODULE	HW_SUBMODULE of „Default IO mapping with Config“ module (respectively „Axis_x“ when using B8050-ML-PN & MB8050-ML-PN)
Axis	tstLM_Axis	Axis reference (IN_OUT)

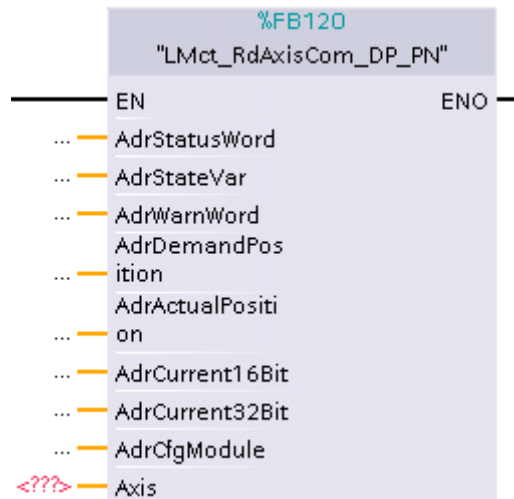
### 3.2.3 Profibus: LMct\_RdAxisCom\_DP\_PN

This function block LMct\_RdAxisCom\_DP\_PN reads the output data of the axis reference (tstLM\_Axis), prepares and writes them to the output addresses.

Should be called at the end of the PLC cycle or at least after all other library function blocks.

Supported drives:

- All LinMot drives with Profibus interface
- **Not recommended:** All LinMot drives with Profinet interface



Inputs		
Name	Data type	Description
AdrStatusWord	HW_SUBMODULE	HW_SUBMODULE of Control/StatusWord module
AdrStateVar	HW_SUBMODULE	HW_SUBMODULE of Get StateVar module
AdrWarnWord	HW_SUBMODULE	HW_SUBMODULE of Get WarnWord module
AdrDemandPosition	HW_SUBMODULE	HW_SUBMODULE of Get Demand Position module
AdrActualPosition	HW_SUBMODULE	HW_SUBMODULE of Get Actual Position module
AdrActualCurrent16b	HW_SUBMODULE	HW_SUBMODULE of Get Current 16b module
AdrActualCurrent32b	HW_SUBMODULE	HW_SUBMODULE of Get Current 32b module
AdrCfgChannel	HW_SUBMODULE	IHW_SUBMODULE of Parameter Channel module
Axis	tstLM_Axis	Axis reference (IN_OUT)

3.2.4 Profibus: LMct\_WrAxisCom\_DP\_PN

This function block LMct\_WrAxisCom\_DP\_PN writes the output data of the axis reference (tstLM\_Axis), prepares and writes them to the output addresses.

Should be called at the end of the PLC cycle or at least after all other library function blocks.

Supported drives:

- All LinMot drives with Profibus interface
- **Not recommended:** All LinMot drives with Profinet interface



Inputs		
Name	Data type	Description
AdrControlWord	HW_SUBMODULE	HW_SUBMODULE of Control/StatusWord module
AdrMCCmdInterface	HW_SUBMODULE	HW_SUBMODULE of MC CMD Interface module
AdrConfigChannel	HW_SUBMODULE	HW_SUBMODULE of Parameter Channel module
Axis	tstLM_Axis	Axis reference (IN_OUT)

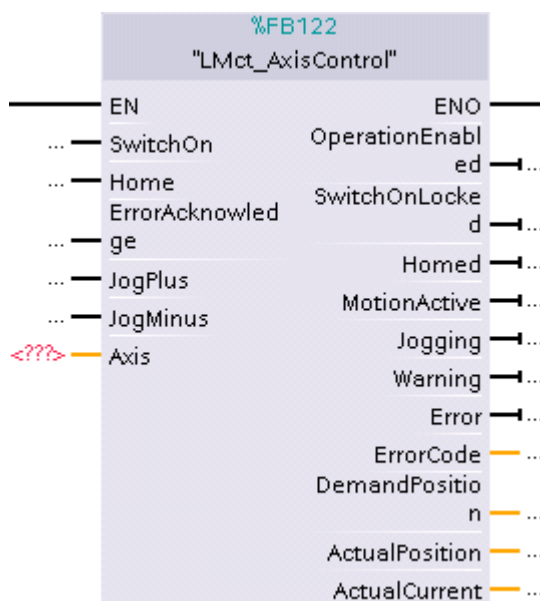
### 3.2.5 LMct\_AxisControl

This function block controls the state machine of a LinMot drive. The outputs show the status of the axis.

**All IO and axis control function blocks (chapter 3.2.1 - 3.2.5) must be called cyclically!**

Supported drives:

- All



Inputs		
Name	Data type	Description
SwitchOn	Bool	Switch on axis
Home	Bool	Start homing of the axis (Has to stay TRUE until the output Homed is set)
ErrorAcknowledged	Bool	Error acknowledge on rising edge
JogPlus	Bool	Jog move positive
JogMinus	Bool	Jog move negative
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
OperationEnabled	Bool	Axis is powered and ready for commands
SwitchOnLocked	Bool	Switch on is locked ( -> Release SwitchOn)
Homed	Bool	Axis is homed (has reference)
MotionActive	Bool	Set point generation (VAI, curve) active (the drive is attempting to move)
Jogging	Bool	Axis is moving in jog mode
Warning	Bool	Warning active
Error	Bool	Error has occurred and drive is in the error state
ErrorCode	UInt	Shows the error code. (See user manual „Motion Control SW“)
DemandPosition	Real	Demand position of the axis in mm
ActualPosition	Real	Actual position of the axis in mm
ActualCurrent	Real	Actual current of the axis in A (Ampère)

### 3.3 MC Function Blocks

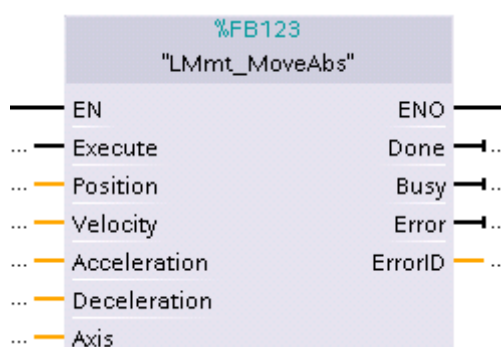
#### 3.3.1 LMmt\_MoveAbs & LMmt\_MoveAbsJ

With this function block a motion to an absolute position with the set maximal velocity, acceleration and deceleration is executed.

An actually running command can be overwritten with new parameters (*Position, Velocity, Acceleration and Deceleration*) by setting a new rising edge on *Execute*.

Supported drives:

- *LMmt\_MoveAbs*: All
- *LMmt\_MoveAbsJ*: C1200, C1400, E1400V2 series only



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Position	Real		Target position in [mm]
Velocity	Real		Max. velocity in [m/s]
Acceleration	Real		Acceleration in [m/s²]
Deceleration	Real		Deceleration in [m/s²]
Jerk	Real	Max. 160'000	Jerk in [m/s³] (LMmt_MoveAbsJ only)
Axis	tstLM_Axis		Axis reference (IN_OUT)

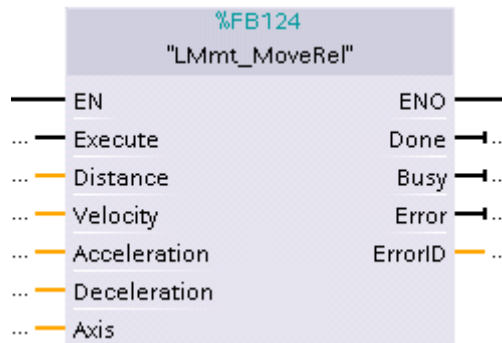
Outputs		
Name	Data type	Description
Done	Bool	Command done and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

### 3.3.2 LMmt\_MoveRel & LMmt\_MoveRelJ

With this function block a relative motion can be done. The motion's dynamics are defined with the inputs velocity, acceleration and deceleration.

Supported drives:

- *LMmt\_MoveRel:* All
- *LMmt\_MoveRelJ:* C1200, C1400, E1400V2 series only



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Distance	Real		Position increment in [mm]
Velocity	Real		Max. velocity in [m/s]
Acceleration	Real		Acceleration in [m/s <sup>2</sup> ]
Deceleration	Real		Deceleration in [m/s <sup>2</sup> ]
Jerk	Real	Max. 160'000	Jerk in [m/s <sup>3</sup> ] (LMmt_MoveRelJ only)
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

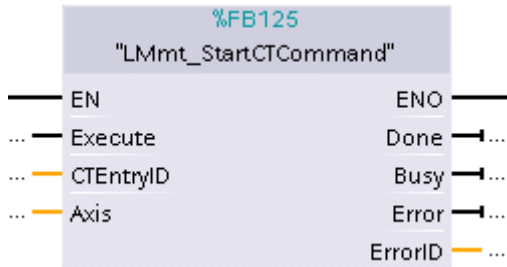


3.3.3 LMmt\_StartCTCommand

This function block starts a line of the Command Table (stored in the drive).

Supported drives:

- All (The range below in brackets stands for B1150-ML-xx drives)



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CTEntryID	UInt	1...255 (1...31)	ID of the line of the Command Table
Axis	tstLM_Axis		Axis reference (IN_OUT)

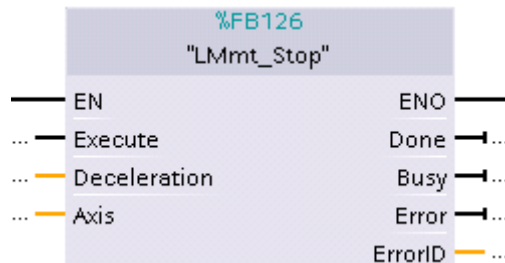
Outputs		
Name	Data type	Description
Done	Bool	Command sent
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

### 3.3.4 LMmt\_Stop

This function block stops the axis immediately with the set deceleration.  
Other active MC function blocks will be aborted!

Supported drives:

- All



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Deceleration	Real		Deceleration in [m/s <sup>2</sup> ]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command sent and axis stopped
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



#### Note:

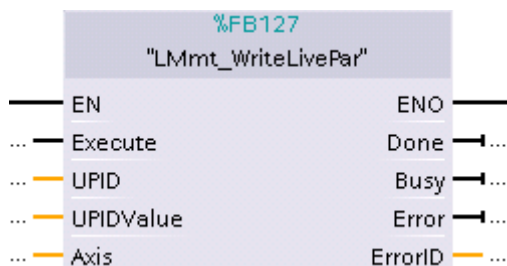
This function block aborts other running MC function blocks. If the input Deceleration is zero or not connected 10m/s<sup>2</sup> is taken by default!

### 3.3.5 LMmt\_WriteLivePar

With this function block a Live Parameter of the drive can be modified/written ("live" parameters can be changed during runtime).

Supported drives:

- All



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
UPID	UInt		Parameter address (Unique Parameter ID)
UPIDValue	DInt		Parameter value
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command sent
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Attention:**

It is highly recommended to use the function block FB200 (LMcf\_ParaAccess) for parameter access. The LMmt\_WriteLiveParameter function block should only be used if the Config Modul / Parameter Channel in the HWConfig is not used for any reason.

The config function blocks (LMcf\_...) can be used in parallel to the other function blocks.

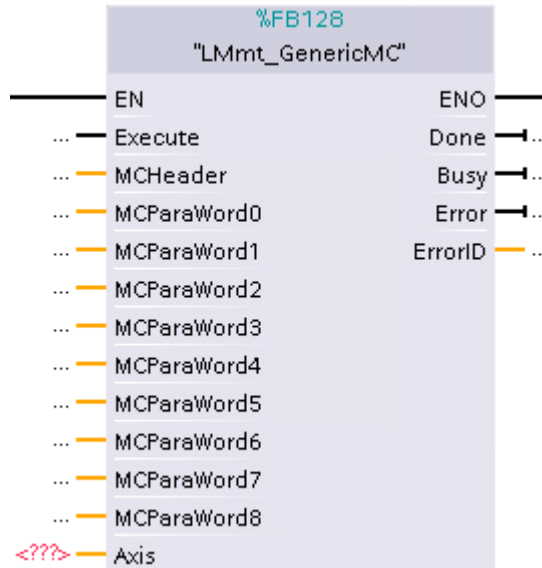
### 3.3.6 LMmt\_GenericMC

With this function block all available motion commands (of the used drive) can be executed.  
The parameters have to be scaled according to the selected *MCHeader*!

A list of all supported motion commands can be found in the user manual "Motion Control SW".

Supported drives:

- All



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
MCHeader	Word		Motion Command Master ID & Sub ID
MCParaWord0	Word		0. Parameter word
MCParaWord1	Word		1. Parameter word
MCParaWord2	Word		2. Parameter word
MCParaWord3	Word		3. Parameter word
MCParaWord4	Word		4. Parameter word
MCParaWord5	Word		5. Parameter word
MCParaWord6	Word		6. Parameter word
MCParaWord7	Word		7. Parameter word
MCParaWord8	Word		8. Parameter word
Axis	tstLM_Axis		Axis reference (IN_OUT)

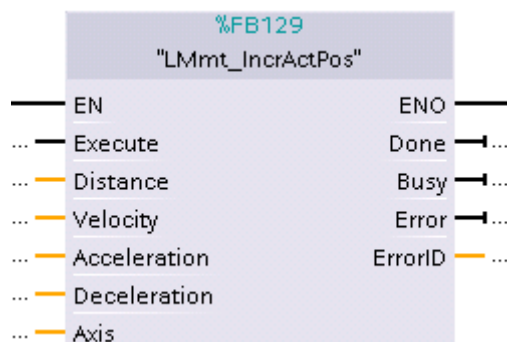
Outputs		
Name	Data type	Description
Done	Bool	Command sent
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**3.3.7 LMmt\_IncrActPos & LMmt\_IncrActPosJ**

This function block increments the actual position of the motor. The motion's dynamics are defined with the inputs *Velocity*, *Acceleration* and *Deceleration*.

Supported Drives:

- *LMmt\_IncrActPos*: All
- *LMmt\_IncrActPosJ*: C1200, C1400, E1400V2 series only



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Distance	Real		Position increment in [mm]
Velocity	Real		Max. velocity in [m/s]
Acceleration	Real		Acceleration in [m/s <sup>2</sup> ]
Deceleration	Real		Deceleration in [m/s <sup>2</sup> ]
Jerk	Real	Max. 160'000	Jerk in [m/s <sup>3</sup> ] (LMmt_IncrActPosJ only)
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

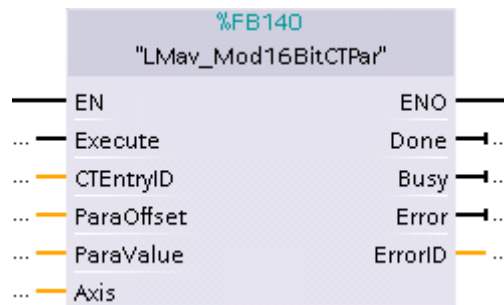
### 3.4 MC Function Blocks (Advanced)

#### 3.4.1 LMax\_Mod16BitCTPar

With this function block the value of a parameter (16Bit) in the command table can be modified (RAM only).

Supported Drives:

- All except B8050-ML-PN and MB8050-ML-PN (with connected B1150-ML-xx)



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CTEntryID	UInt		ID of the command table line
ParaOffset	UInt		Offset of the parameter to be written
ParaValue	Int		Value of the parameter to be written
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command sent
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



#### Note:

Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

ParaOffset:

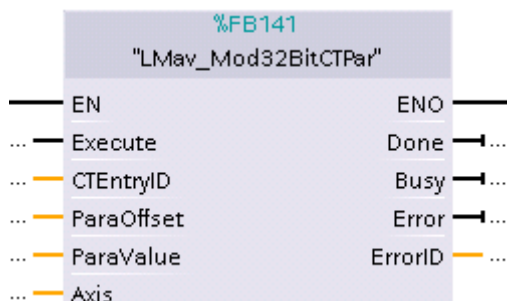
6 = First motion command parameter

### 3.4.2 LMav\_Mod32BitCTPar

With this function block the value of a parameter (32Bit) in the command table can be modified (RAM only).

*Supported Drives:*

- All except B8050-ML-PN and MB8050-ML-PN (with connected B1150-ML-xx)



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CTEntryID	UInt		ID of the command table line
ParaOffset	UInt		Offset of the parameter to be written
ParaValue	DInt		Value of the parameter to be written
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command sent
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Note:**

Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

ParaOffset:

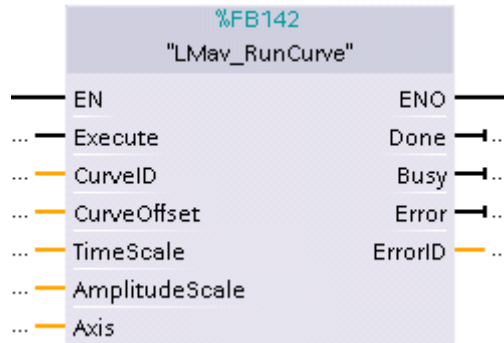
6 = First motion command parameter

### 3.4.3 LMax\_RunCurve

With this function block a motion profile (curve) that is stored in the drive can be executed.

Supported drives:

- E and C series
- B series only with Technology Function: Curves on B1100 (Part Nr. 0150-2504)



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CurveID	UInt	1...99	Curve number (ID)
CurveOffset	Real		Offset of the curve
TimeScale	Real	0.0...200.0	Time scale in [%]
AmplitudeScale	Real	-2000.00...+2000.00	Amplitude scale in [%]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



**Note:**

Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

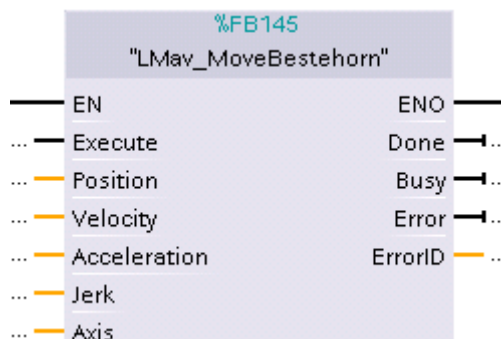


### 3.4.4 LMav\_MoveBestehorn

With this function block the axis can be moved to the target position using a Bestehorn profile.

Supported drives:

- E1400, E1200 & C1x00 series



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Position	Real		Target position in [mm]
Velocity	Real		Max. velocity in [m/s]
Acceleration	Real		Acceleration in [m/s <sup>2</sup> ]
Jerk	Real		Maximum jerk in [m/s <sup>3</sup> ]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Note:**

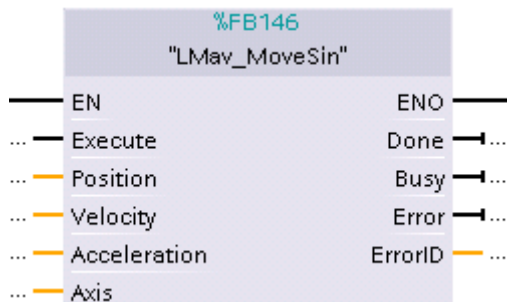
Additional information for this command can be found in the user manual "Motion Control Software".  
(Recommended Documentation)

### 3.4.5 LMax\_MoveSin

With this function block the axis can be moved to the target position using a Sin profile.

Supported drives:

- E1400, E1200 & C1x00 series



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Position	Real		Target position in [mm]
Velocity	Real		Max. velocity in [m/s]
Acceleration	Real		Acceleration in [m/s²]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



#### Note:

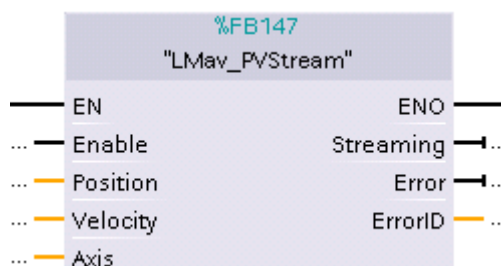
Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

### 3.4.6 LMAV\_PVStream

With this function block the setpoint position and setpoint velocity (e.g. of a virtual axis) can be transmitted cyclically to the drive.

*Supported drives:*

- All



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Position	Real		Setpoint position in [mm]
Velocity	Real		Setpoint velocity in [mm/s]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Streaming	Bool	Streaming active, function block is transmitting setpoint values
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Note:**

Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

**Attention:**

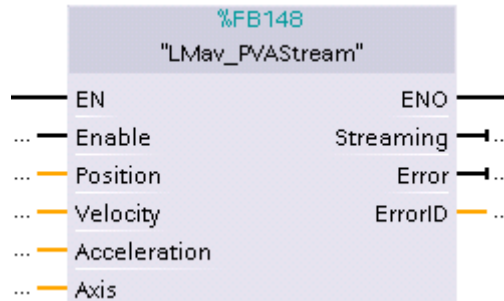
Profibus / Profinet must run synchronous to the PLC task in which this function block is called! The drive expects a new command in every bus cycle.

### 3.4.7 LMAV\_PVAStream

With this function block the setpoint position, setpoint velocity and setpoint acceleration (e.g. of a virtual axis) can be transmitted cyclically to the drive.

Supported drives:

- C1x00, E1200 & E1400 series



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Position	Real		Setpoint position in [mm]
Velocity	Real		Setpoint velocity in [mm/s]
Acceleration	Real		Setpoint acceleration in [mm/s²]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Streaming	Bool	Streaming active, function block is transmitting setpoint values
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



**Note:**  
Additional information for this command can be found in the user manual "Motion Control Software".  
(Recommended Documentation)



**Attention:**  
Profibus / Profinet must run synchronous to the PLC task in which this function block is called!  
The drive expects a new command in every bus cycle.

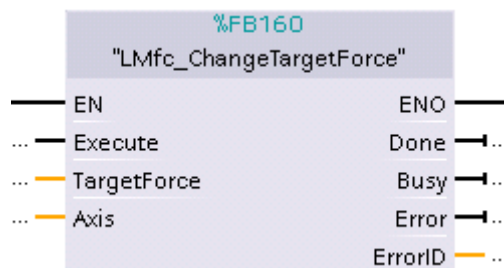
3.5 MC Function Blocks Force Control

3.5.1 LMfc\_ChangeTargetForce

With this function block the target force can be changed if the drive is in force control mode.

Supported drives:

- All with installed Technology Function: Closed Loop FC (Part Nr. 0150-2503)



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
TargetForce	Real	Target force in [N]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



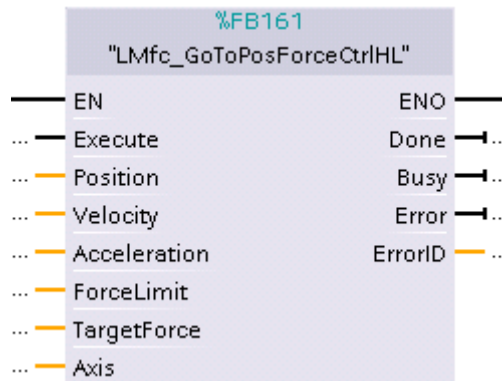
**Note:**  
Additional information for this command can be found in the user manual “Motion Control Software”.  
(Recommended Documentation)

### 3.5.2 LMfc\_GoToPosForceCtrlHL

With this function block the motor starts moving direction target position (*Position*). If the measured force **goes over** the *ForceLimit* the drive switches to force control mode with target force = *TargetForce*. Is the target position reached without switching to force control mode an error is generated (ErrorID = 07h).

Supported drives:

- All with installed Technology Function: Closed Loop FC (Part Nr. 0150-2503)



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
Position	Real	Target position in [mm]
Velocity	Real	Maximal velocity in [m/]
Acceleration	Real	Acceleration in [m/s²]
ForceLimit	Real	Force limit that has to be exceeded to switch to force control mode in [N]
TargetForce	Real	Target force after switching to force control mode in [N]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed and axis in force control mode
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



#### Note:

Additional information for this command can be found in the user manual "Motion Control Software". (Recommended Documentation)

3.5.3 LMfc\_GoToPosForceCtrlLL

With this function block the motor starts moving direction target position (*Position*). If the measured force **goes below** the *ForceLimit* the drive switches to force control mode with target force = *TargetForce*. Is the target position reached without switching to force control mode an error is generated (ErrorID = 07h).

Supported drives:

- All with installed Technology Function: Closed Loop FC (Part Nr. 0150-2503)



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
Position	Real	Target position in [mm]
Velocity	Real	Maximal velocity in [m/s]
Acceleration	Real	Acceleration in [m/s²]
ForceLimit	Real	Force limit that has to be gone below to switch to force control mode in [N]
TargetForce	Real	Target force after switching to force control mode in [N]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed and axis in force control mode
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



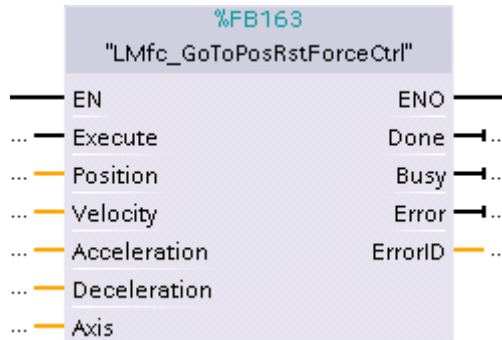
**Note:**  
Additional information for this command can be found in the user manual “Motion Control Software”.  
(Recommended Documentation)

### 3.5.4 LMfc\_GoToPosRstForceCtrl

With this function block the drive can be set back to position control loop and moved from its actual position to the position defined with *Position*, *Velocity*, *Acceleration* and *Deceleration*.

Supported drives:

- All with installed Technology Function: Closed Loop FC (Part Nr. 0150-2503)



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
Position	Real	Target position in [mm]
Velocity	Real	Maximal velocity in [m/s]
Acceleration	Real	Acceleration in [m/s <sup>2</sup> ]
Deceleration	Real	Deceleration in [m/s <sup>2</sup> ]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed, axis position controlled and in target position
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



**Note:**  
Additional information for this command can be found in the user manual "Motion Control Software".  
(Recommended Documentation)



**3.6 MC Function Blocks CAM****3.6.1 LMcm\_CAMCtrl**

With this function block the CAM functionality of the LinMot drives can be controlled.

*Supported drives:*

- *E series*
- *B series only with Technology Function: Curves on B1100 (Part Nr. 0150-2504)*



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
Mode	UInt	0..7	Mode
EncValue	UDInt		Encoder value to be set in counts [Incr] (increments)
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed (& in mode 2: axis at sync position)
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

Mode		
Value	Used Inputs	Description
0	-	CAM Enable
1	-	CAM Disable
2	-	CAM Go To Sync Pos
3	EncValue	Encoder CAM Set Value
4	-	Encoder CAM 1 Enable
5	-	Encoder CAM 1 Disable
6	-	Encoder CAM 2 Enable
7	-	Encoder CAM 2 Disable

**Note:**

Additional information regarding the CAM functionality can be found in the user manual "Motion Control Software". (Recommended Documentation)

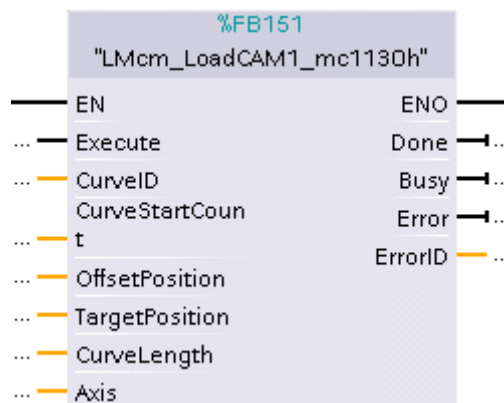
### 3.6.2 LMcm\_LoadCAM1\_mc1130h

With this function block a motion profile (curve) stored on the drive is loaded into CAM1. The motion profile can be moved and scaled using the inputs *CurveStartCount* (at what count the curve should start), the *OffsetPosition* (curve start position), *TargetPosition* (curve target position) and *CurveLength* (length of curve in counts).

**ATTENTION:** The start position of the curve stored in the drive must not be the same as its end position (no scaling possible)! Furthermore, the curve's start position should be 0mm to be scaled in an optimal way.

Supported drives:

- E series
- B series only with Technology Function: Curves on B1100 (Part Nr. 0150-2504)



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CurveID	UInt	1..99	Curve number to be loaded
CurveStartCount	UDInt		Curve start count in counts [Incr]
OffsetPosition	Real		Start position of the CAM in [mm]
TargetPosition	Real		End position of the CAM in [mm]
CurveLength	UDInt		CAM length in counts [Incr]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed, curve loaded into CAM1
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



**Note:**

Additional information for this command (1130h) can be found in the user manual "Motion Control Software". (Recommended Documentation)

**3.6.3 LMcm\_LoadCAM2\_mc1230h**

With this function block a motion profile (curve) stored on the drive is loaded into CAM2. The motion profile can be moved and scaled using the inputs *CurveStartCount* (at what count the curve should start), the *OffsetPosition* (curve start position), *TargetPosition* (curve target position) and *CurveLength* (length of curve in counts).

**ATTENTION:** The start position of the curve stored in the drive must not be the same as its end position (no scaling possible)! Furthermore, the curve's start position should be 0mm to be scaled in an optimal way.

*Supported drives:*

- *E series*
- *B series only with Technology Function: Curves on B1100 (Part Nr. 0150-2504)*



Inputs			
Name	Data type	Range	Description
Execute	Bool		Execute command (rising edge)
CurveID	UInt	1..99	Curve number to be loaded
CurveStartCount	UDInt		Curve start count in counts [Incr]
OffsetPosition	Real		Start position of the CAM in [mm]
TargetPosition	Real		End position of the CAM in [mm]
CurveLength	UDInt		CAM length in counts [Incr]
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed, curve loaded into CAM2
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Note:**

Additional information for this command (1230h) can be found in the user manual "Motion Control Software". (Recommended Documentation)

### 3.7 Config Function Blocks

The config function blocks grant access to parameters, curves and the command table of a LinMot drive. Additionally, they provide functions like restart or stop the firmware or parts of it, restoring the parameters of each firmware layer to default values, or getting the error text as STRING.

**Important:**

If data on the drive (command table, curves) is saved from the RAM to the flash memory the firmware layer MC\_SW must be stopped!

That can be done using the config function block LMcf\_StopStartDefault with mode 5.

With mode 6 it can be restarted afterwards.

This is required for the following function blocks and modes:

LMcf\_CTAccess                      Mode 0

LMcf\_CurveAccess                Mode 0

The function blocks listed in this chapter are compatible with the following drives and interfaces:

E1400 Series                      Profibus / Profinet

E1200 Series                      Profibus / Profinet

C1200 Series                      Profinet

C1100 Series                      Profinet

E1130-DP-xx                      Profibus

B1150-ML-xx                      Profinet (Connected to B8050-ML-PN, usable with limitations)

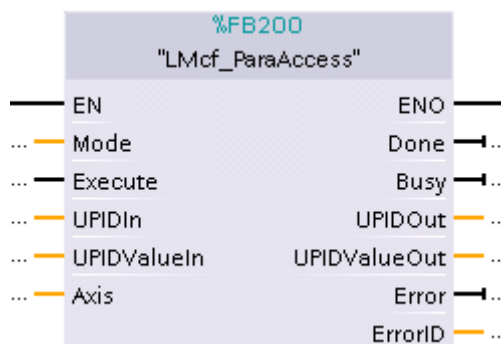
MB1150-ML-xx                    Profinet (Connected to M8050-ML-PN, usable with limitations)

### 3.7.1 LMcf\_ParaAccess

This function block provides access to the parameters of a LinMot drive. Read and write RAM and ROM parameters. Read minimal, maximal and default values of a parameter.

Supported drives:

- E & C series
- B1150-ML-xx: modes 0, 1, 2 & 3 only



Inputs			
Name	Data type	Range	Description
Mode	UInt	0...7	Mode
Execute	Bool		Execute command (rising edge)
UPIDIn	UInt		Parameter ID (UPID)
ValueIn	DInt		Value to be written
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed
Busy	Bool	Command active
ValueOut	DInt	Read value / Feedback of written value
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

Mode		
Value	Used Inputs	Description
0	UPIDIn	Read ROM Value of Parameter by UPID
1	UPIDIn	Read RAM Value of Parameter by UPID
2	UPIDIn, ValueIn	Write ROM Value of Parameter by UPID
3	UPIDIn, ValueIn	Write RAM Value of Parameter by UPID
4	UPIDIn, ValueIn	Write RAM and ROM Value of Parameter by UPID
5	UPIDIn	Get minimal Value of Parameter by UPID
6	UPIDIn	Get maximal Value of Parameter by UPID
7	UPIDIn	Get default Value of Parameter by UPID

### 3.7.2 LMcf\_GetModUPIDList

This function block reads a list of parameters and their values that have been modified (compared to factory defaults) and stores them in the array. Can be used to save the configuration of a drive on the PLC.

Supported drives:

- E & C series



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
NrOfEntries	UInt	Number of parameters to be read. Recommended: 200
Axis	tstLM_Axis	Axis reference (IN_OUT)
UPIDArray	Array[0..200] of tstLM_CfgUPIDListEntry	Array where the parameters are stored



Outputs		
Name	Data type	Description
Done	Bool	Command executed / UPID list read
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)



#### Attention:

Up to now an array of fixed size has to be connected to the function block.

The DB "GLV" (DB1000) in the example projects contains an example array.

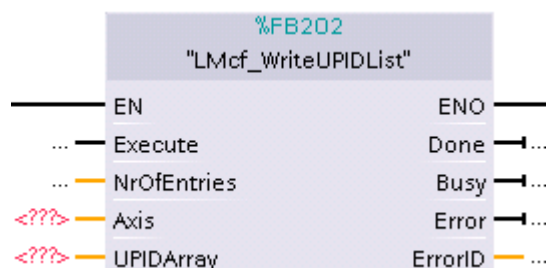
  Axis\_1\_UPIDList    Array[0..200] of "tstLM\_CfgUPIDListEntry"

### 3.7.3 LMcf\_WriteUPIDList

This function block writes a list of parameters (UPID, Value) stored in an array DB to the drive.

Supported drives:

- E & C series
- B1150-ML-xx



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
NrOfEntries	UInt	Maximal number of parameters to be written. Recommended: 200
Axis	tstLM_Axis	Axis reference (IN_OUT)
UPIDArray	Array[0..200] of tstLM_CfgUPIDListEntry	Array where the parameters are stored

Outputs		
Name	Data type	Description
Done	Bool	Command executed / UPID list written
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

**Attention:**

The function block stops writing as soon as it finds an entry with **UPID = 0**. Therefore, the last parameter entry in the DB must be followed by an entry with UPID = 0.  
Or as soon as the number at "NrOfEntries" has been reached.

Up to now an array of fixed size has to be connected to the function block.  
The DB "GLV" (DB1000) in the example projects contains an example array.

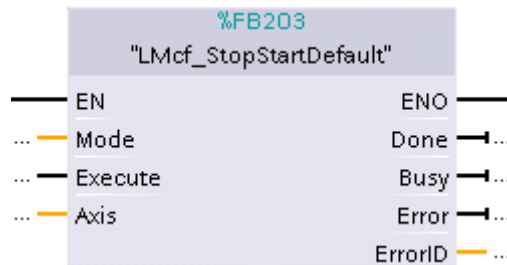
Axis\_1\_UPIDList      Array[0..200] of "tstLM\_CfgUPIDListEntry"

### 3.7.4 LMcf\_StopStartDefault

This function block provides the functionality to restart the drive, stop and start single firmware layers, or set the parameters of each firmware layer to factory defaults.

Supported drives:

- E & C series
- B1150-ML-xx: only modes 0, 5 & 6



Inputs			
Name	Data type	Range	Description
Mode	UInt	0...6	Mode
Execute	Bool		Execute command (rising edge)
Axis	tstLM_Axis		Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

Mode		
Value	Used Inputs	Description
0	-	Restart drive
1	-	Set parameter ROM values to default (OS SW)
2	-	Set parameter ROM values to default (MC SW)
3	-	Set parameter ROM values to default (Interface SW)
4	-	Set parameter ROM values to default (Application SW)
5	-	Stop MC and Application Software (for Flash Access)
6	-	Start MC and Application Software



#### Attention:

Mode 5 & 6 are important when using the config function blocks LMcf\_CurveAccess and LMcf\_CTAccess. Before curves or command table entries are saved from RAM to the flash memory of the drive **the MC\_SW must be stopped!**

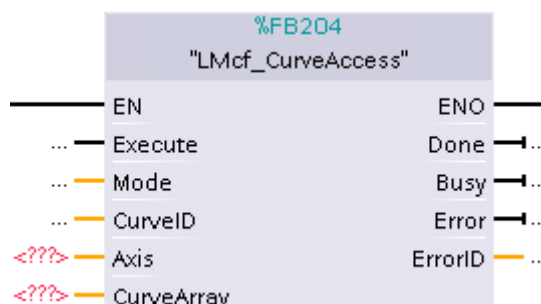


### 3.7.5 LMcf\_CurveAccess

This function provides access to motion profiles (curves) on a LinMot drive. It is possible to read, write, modify and delete curves as well as saving all curves from the RAM to the flash memory.

Supported drives:

- E & C series



Inputs			
Name	Data type	Range	Description
Mode	UInt	0...4	Mode
Execute	Bool		Execute command (rising edge)
CurveID	UInt	1...99	Curve number (ID)
Axis	tstLM_Axis		Axis reference (IN_OUT)
CurveArray	Array[0..1018] of DWord		Array with/for curve data (max. 1001 curve set-points)

Outputs		
Name	Data type	Description
Done	Bool	Command executed
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

Mode		
Value	Used Inputs	Description
0	-	Save all Curves from RAM to Flash MC_SW must be stopped!
1	-	Delete all Curves (RAM)
2	CurveArray	Add Curve (RAM)
3	CurveArray	Modify Curve (RAM)
4	CurveID, CurveArray	Get Curve (data is stored in array at CurveArray)

**Attention:**

Up to now an array of fixed size has to be connected to the function block

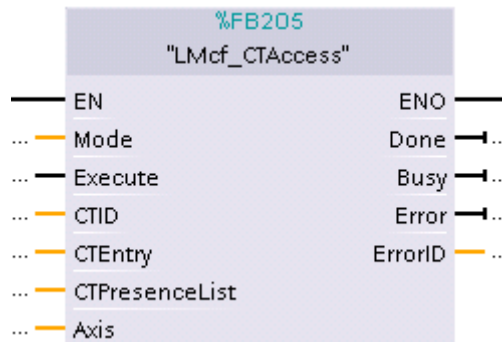
The curve stored on the LinMot drive **must not exceed 1001** set-points!

### 3.7.6 LMcf\_CTAccess

This function block provides access to the command table of a LinMot drive. Read and write entries, delete entries, delete the complete command table, save the command table from RAM to flash memory.

Supported drives:

- E & C series



Inputs			
Name	Data type	Range	Description
Mode	UInt	0...5	Mode
Execute	Bool		Execute command (rising edge)
CTID	UInt	1...255	Command Table ID (line number)
CTEntry	tstLM_CfgCTEntry		Command Table entry (IN_OUT)
CTPresenceList	Array[0..7] of DWord		Array for the presence list (IN_OUT)
Axis	tstLM_Axis		Axis reference (IN_OUT)

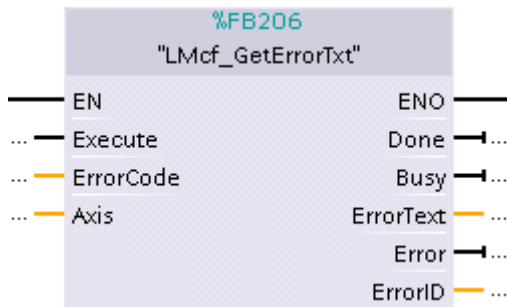
Outputs		
Name	Data type	Description
Done	Bool	Command executed
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

Mode		
Value	Used Inputs	Description
0	-	Save to Flash: Saves the Command Table to the flash memory MC_SW must be stopped!
1	-	Delete all Entries (RAM)
2	CTID	Delete Entry (RAM)
3	CTID, CTEntry	Write Entry (RAM)
4	CTID, CTEntry	Get Entry (is stored in CTEntry)
5	CTPresenceList	Read presence list from drive

3.7.7 LMcf\_GetErrorTxt (FB206)

This function blocks returns a STRING containing the error text according to the input *ErrorCode*. The *ErrorCode* is shown at the output *ErrorCode* of the *LMct\_AxisControl* function block. See chapter 3.2.5 *LMct\_AxisControl*.

- Supported drives:
- E & C series



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
ErrorCode	UInt	Error code (See axis control function block)
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command executed / Error text read
Busy	Bool	Command active
ErrorText	String[32]	Error text as STRING
Error	Bool	Error in function block
ErrorID	UInt	ErrorID (See chapter 5 Error Descriptions)

### 4 Function Blocks for Rotary Motors connected to LinMot Drives

LinMot Drives are able to drive rotary motors of different manufacturers. This chapter describes the function blocks specifically for this kind of motors. The modulo calculation is done in the PLC. That's why some new parameters have to be introduced.



#### Attention:

The function blocks described in this chapter are thought to be used with rotary motors only.

#### 4.1 Parameters & Conversions

##### 4.1.1 GearFactor (Data type REAL)

If a rotary motor with gear box is used the *GearFactor* has to be set to its ratio.

Example: 30:1 Gear box => GearFactor = 30

##### 4.1.2 UnitScale (Data type REAL)

UnitScale is a factor that is calculated based on the configuration of the LinMot Drive, the GearFactor and the drive internal position resolution. This factor is applied to the input parameters of the different function blocks (see chapter 4.1.4 Examples).

$$\text{UnitScale} = \frac{\text{mm per revolution (LinMot – Talk MotorWizard)} * \text{GearFactor} * 10000}{360}$$

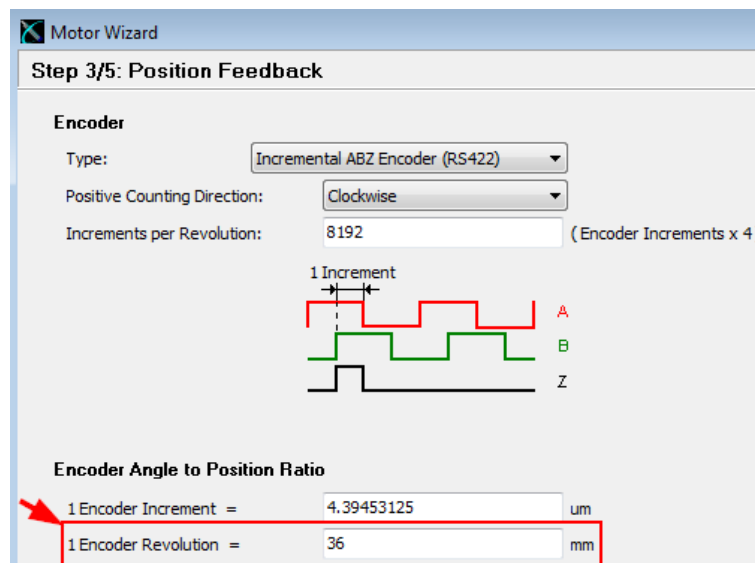


Figure 9: mm per revolution in the LinMot-Talk Motor Wizard

Example (no Gear box, 36mm per revolution in the LinMot-Talk Motor Wizard):

$$\Rightarrow \text{UnitScale} = 36\text{mm} * 1 * 10'000 / 360 = \mathbf{1'000}$$

Example (30:1 Gear box, 12mm per revolution in the LinMot-Talk Motor Wizard -> 360mm per revolution of the gear box):

$$\Rightarrow \text{UnitScale} = 12\text{mm} * 30 * 10'000 / 360 = \mathbf{10'000}$$

### 4.1.3 ModuloFactor (Data type DInt)

Standardization to 360° per motor / gear box revolution. The *ModuloFactor* is required to calculate the number of revolutions of the motor and to turn absolute within one revolution.

$$\text{ModuloFactor} = 360 * \text{UnitScale}$$

Example: **ModuloFactor** = 360 \* 1'000 = **360000**

### 4.1.4 Examples

Function block input	Data Type	Scaling
Distance or TargetAngle	DInt	Grad * UnitScale (E.g.: 20.5° * 1'000 = 20'500)
Velocity / Speed:	Real	Grad/s * UnitScale (E.g.: Velocity = 360°/s * 1'000 = 360'000)
Acceleration:	Real	Grad/s² * UnitScale (E.g.: Acceleration = 360°/s² * 1'000 = 360'000)
Deceleration:	Real	Grad/s² * UnitScale (E.g.: Acceleration = 360°/s² * 1'000 = 360'000)
Actual demand angle in degrees =		LMct_RotaryMotorControl.MotorDemandAngle / UnitScale
Actual actual angle in degrees =		LMct_RotaryMotorControl.MotorActualAngle / UnitScale

**Note:**

The *Distance* and *TargetAngle* inputs are of data type DINT. Depending of the use case a typecast may be required.

E.g. *TargetAngle* := REAL\_TO\_DINT(20.5 \* *UnitScale*);

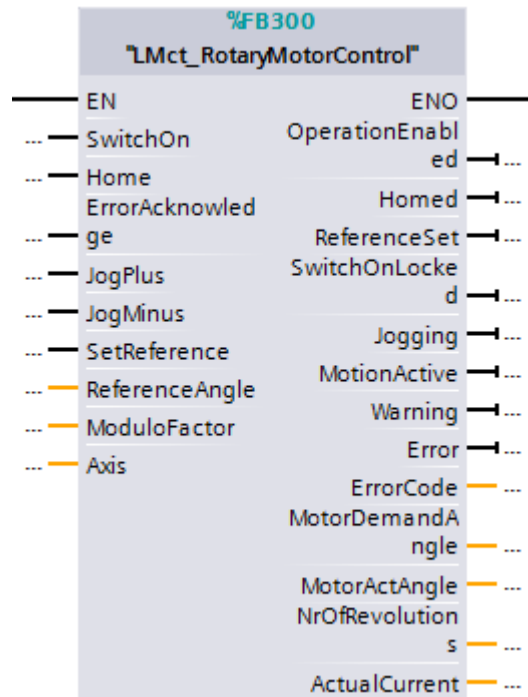
### 4.2 Function Blocks

#### 4.2.1 LMct\_RotaryMotorControl

Using this function block the state machine of the LinMot drive is controlled. The outputs show the status of the drive.

Supported Drives:

- All

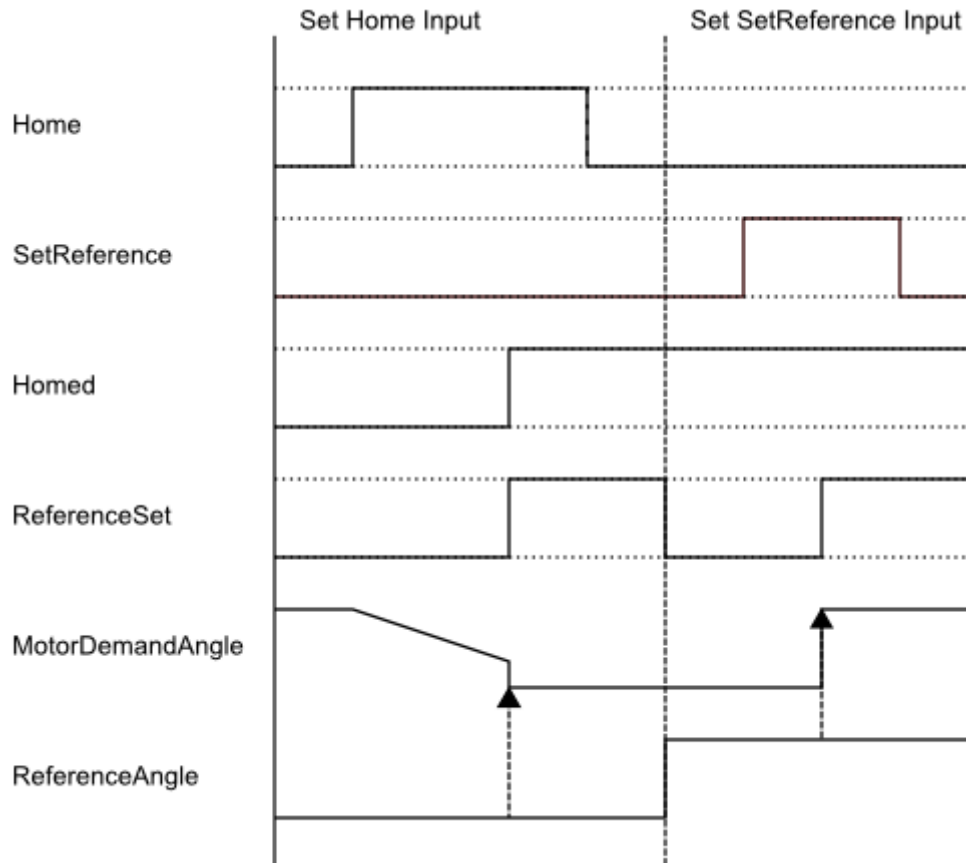


Inputs		
Name	Data type	Description
SwitchOn	Bool	Switch on axis
Home	Bool	Start homing of the axis
ErrorAcknowledge	Bool	Error acknowledge on rising edge
JogPlus	Bool	Jog move positive
JogMinus	Bool	Jog move negative
SetReference	Bool	Set <i>MotorDemandAngle</i> to value of <i>ReferenceAngle</i>
ReferenceAngle	DInt	Reference angle in [Units]
ModuloFactor	DInt	Modulo factor in [Units]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
OperationEnabled	Bool	Axis is powered and ready for commands
SwitchOnLocked	Bool	Switch on is locked ( -> Release <i>SwitchOn</i> )
Homed	Bool	Axis is homed
ReferenceSet	Bool	Axis has reference
MotionActive	Bool	Setpoint generation (VAI, curve) active (the drive is attempting to move)
Jogging	Bool	Axis is moving in jog mode
Warning	Bool	Warning active
Error	Bool	Error has occurred and drive is in the error state
ErrorCode	Int	Shows the error code. (See user manual „Motion Control SW“)
MotorDemandAngle	DInt	Demand angle of the motor in [Units]
MotorActualAngle	DInt	Actual angle of the motor in [Units]
NrOfRevolutions	DInt	Number of revolutions of the motor
ActualCurrent	Real	Actual current of the motor in A (Ampère)

### Difference between *Home* and *SetReference*

- Setting the Home inputs starts the homing sequence set in the LinMot drive (E.g., Index On X13 Positive Search). After completing the homing sequence, the outputs *Homed* and *ReferenceSet* are set to TRUE and the *MotorDemandAngle* is set to the value of *ReferenceAngle*.
- Setting the input *SetReference* sets the *MotorDemandAngle* to the value of *ReferenceAngle* and the output *ReferenceSet* is set TRUE.



#### Attention:

Using a motor with absolute position feedback:

- The *Home* input must not be used! Otherwise the coordinate system of the motor is shifted.
- The output *Homed* is always set

The rotary part of LinMot Linear Rotary Motors (PR01-...) has an absolute position sensor (Single-Turn)!



#### Attention:

The LMct\_RotaryMotorControl function block must be called cyclically!

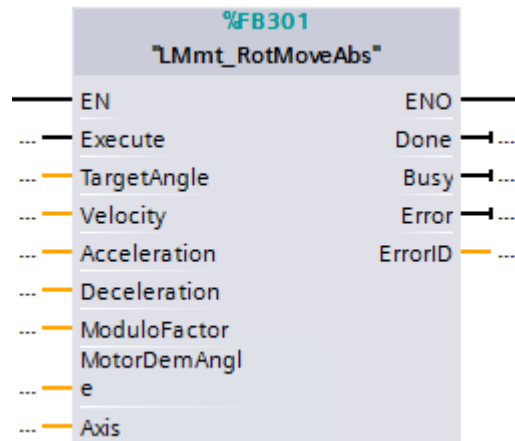


#### 4.2.2 LMmt\_RotMoveAbs

With this function block a motion to an absolute angle (*TargetAngle*), within the actual motor revolution, with the set *Velocity*, *Acceleration* and *Deceleration* is executed.

Supported drives:

- All



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
TargetAngle	DInt	Target angle in [Units]
Velocity	Real	Maximal velocity in [Units/s]
Acceleration	Real	Acceleration in [Units/s²]
Deceleration	Real	Deceleration in [Units/s²]
ModuloFactor	DInt	Modulo factor, see chapter 4.1.3 ModuloFactor (Data type DInt)
MotorDemAngle	DInt	Actual motor demand angle from the axis control function block: LMct_RotaryMotorControl.MotorDemandAngle
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis at target angle
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	Int	ErrorID (See chapter 5 Error Descriptions)

**Attention:**

The function block can only move to the right target angle when the inputs **ModuloFactor** and **MotorDemAngle** are connected correctly.

**Note:**

Please refer to chapter 4.1.4 Examples for examples of how to connect the function block inputs.

Example:

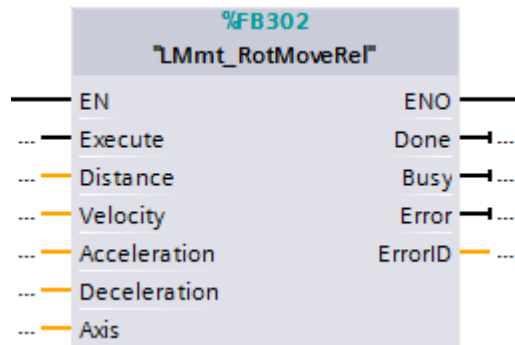
LMmt\_RotMoveAbs.TargetAngle := Angle(20.0°) \* UnitScale = REAL\_TO\_DINT(20.0 \* UnitScale)

### 4.2.3 LMmt\_RotMoveRel

With this function block the actual *DemandAngle* of the motor can be incremented (*Distance*). The motion's dynamics are defined with the inputs *Velocity*, *Acceleration* and *Deceleration*.

Supported Drives:

- All



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
Distance	DInt	Angle increment in [Units]
Velocity	Real	Maximal velocity in [Units/s]
Acceleration	Real	Acceleration in [Units/s²]
Deceleration	Real	Deceleration in [Units/s²]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis at target angle
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	Int	ErrorID (See chapter 5 Error Descriptions)



#### Note:

Please refer to chapter 4.1.4 Examples for examples of how to connect the function block inputs.

Example:

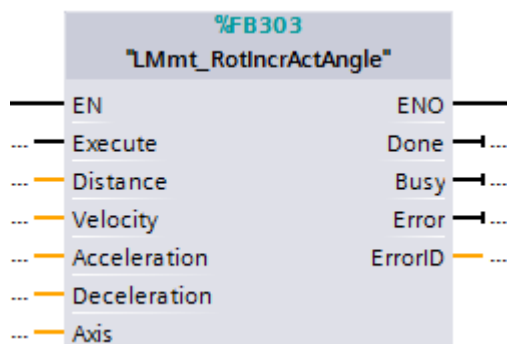
LMmt\_RotMoveRel.Distance:= Angle(20.0°) \* UnitScale = REAL\_TO\_DINT(20.0 \* UnitScale)

#### 4.2.4 LMmt\_RotIncrActAngle

With this function block the actual *ActualAngle* of the motor can be incremented (*Distance*). The motion's dynamics are defined with the inputs *Velocity*, *Acceleration* and *Deceleration*.

Supported Drives:

- All



Inputs		
Name	Data type	Description
Execute	Bool	Execute command (rising edge)
Distance	DInt	Angle increment in [Units]
Velocity	Real	Maximal velocity in [Units/s]
Acceleration	Real	Acceleration in [Units/s²]
Deceleration	Real	Deceleration in [Units/s²]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Done	Bool	Command done and axis at target angle
Busy	Bool	Command active
Error	Bool	Error in function block
ErrorID	Int	ErrorID (See chapter 5 Error Descriptions)

**Note:**

Please refer to chapter 4.1.4 Examples for examples of how to connect the function block inputs.

Example:

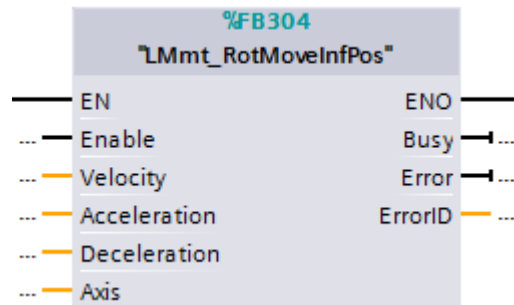
LMmt\_RotIncrActAngle.Distance := Angle(20.0°) \* UnitScale = REAL\_TO\_DINT(20.0 \* UnitScale)

### 4.2.5 LMmt\_RotMoveInfPos

With this function block the motor can be moved **infinite positive** with *Velocity*, *Acceleration* and *Deceleration*. As soon as the *Enable* input is set the motor accelerates to the set velocity. Resetting the *Enable* input stops the motor using the set deceleration.

Supported Drives:

- All



Inputs		
Name	Data type	Description
Enable	Bool	Start / Stop motion
Velocity	Real	Maximal velocity in [Units/s]
Acceleration	Real	Acceleration in [Units/s²]
Deceleration	Real	Deceleration in [Units/s²]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Busy	Bool	Command active, motor is moving
Error	Bool	Error in function block
ErrorID	Int	ErrorID (See chapter 5 Error Descriptions)



#### Note:

Please refer to chapter 4.1.4 Examples for examples of how to connect the function block inputs.

Example:

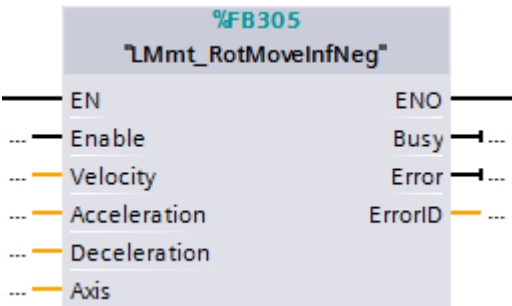
```
LMmt_RotMoveInfPos.Velocity := Angular speed(20.0°/s) * UnitScale
= REAL_TO_DINT(20.0 * UnitScale)
```

4.2.6 LMmt\_RotMoveInfNeg

With this function block the motor can be moved **infinite negative** with *Velocity*, *Acceleration* and *Deceleration*. As soon as the *Enable* input is set the motor accelerates to the set velocity. Resetting the *Enable* input stops the motor using the set deceleration.

Supported Drives:

- All



Inputs		
Name	Data type	Description
Enable	Bool	Start / Stop motion
Velocity	Real	Maximal velocity in [Units/s]
Acceleration	Real	Acceleration in [Units/s²]
Deceleration	Real	Deceleration in [Units/s²]
Axis	tstLM_Axis	Axis reference (IN_OUT)

Outputs		
Name	Data type	Description
Busy	Bool	Command active, motor is moving
Error	Bool	Error in function block
ErrorID	Int	ErrorID (See chapter 5 Error Descriptions)



**Note:**  
Please refer to chapter 4.1.4 Examples for examples of how to connect the function block inputs.

Example:  
LMmt\_RotMoveInfNeg.Velocity := Angular speed(20.0°/s) \* UnitScale  
= REAL\_TO\_DINT(20.0 \* UnitScale)

## 5 Error Descriptions

### 5.1 Error Codes of the Axis Control Function Block

A list of error codes can be found in the user manual „Motion Control SW“ and also in the user manual for each individual interface (Recommended Documentation).

### 5.2 Error ID's of the MC Function Blocks

Error ID	Error text	Description
01h	Axis not ready	Axis is not ready for motion commands. Check axis control function block if "OperationEnabled" output is set TRUE
02h	Axis already has command running	Axis has a running command. Check if another MC function block is busy. Note: By resetting the SwitchOn input on the axis control function block the CommandRunning flag is reset in the axis reference
03h	Axis has error	The axis has an error. Check ErrorCode on the axis control function block
04h	Command interrupted	Command has been interrupted (axis is not "OperationEnabled" any more)
05h	Command aborted	Command has been aborted (e.g. function block LinMotFB_Stop)
06h	Invalid inputs(s)	Invalid values at the function block inputs
07h	Target Position reached but not switched to force control (Limit Force was not detected)	LMfc... function blocks only. The motor has reached the set target position without switching to force control mode → force limit was not reached/measured

### 5.3 Error ID's of the Config Function Blocks

Error ID	Error text	Description
01h	TimeOut (No response from drive)	Drive is not responding within the requested time. Check fieldbus connection
02h	ConfigChannel already busy	Config channel is already busy. Check if another config function block is busy
03h	Invalid Mode selected	Invalid mode selected. Please check Mode input
06h	DB size to small	Connected DB/Array is to small
C0h	UPID error	Unknown UPID selected, check UPID input
C1h	Parameter Type Error	
C2h	Range Error	The value to be written is outside the parameters range
C3h	Address Usage Error	There is an attempt to write a read only parameter
C5h	Error: Command 21h	
D0h	Odd Address	
D1h	Size Error (Curve Service)	
D4h	Curve already defined / Curve not present (Curve Service)	
D7h	MC Software is still running, should be stopped	To store the Command Table / Curves into the Flash memory of the drive the MC_SW must be stopped (LMcf_StopStartDefault, Mode 5)
other		Contact LinMot technical support

## 6 Example Projects

There are some example projects included in the library package to show the general integration to a CPU S7-1200 and a CPU S7-1500.

Each project integrates the library provided with this document (see chapter File List).

### 6.1 CPU S7-1500 (Profinet), LinMot\_TIA\_SCL\_S71500\_Example\_Project\_PN\_...zip

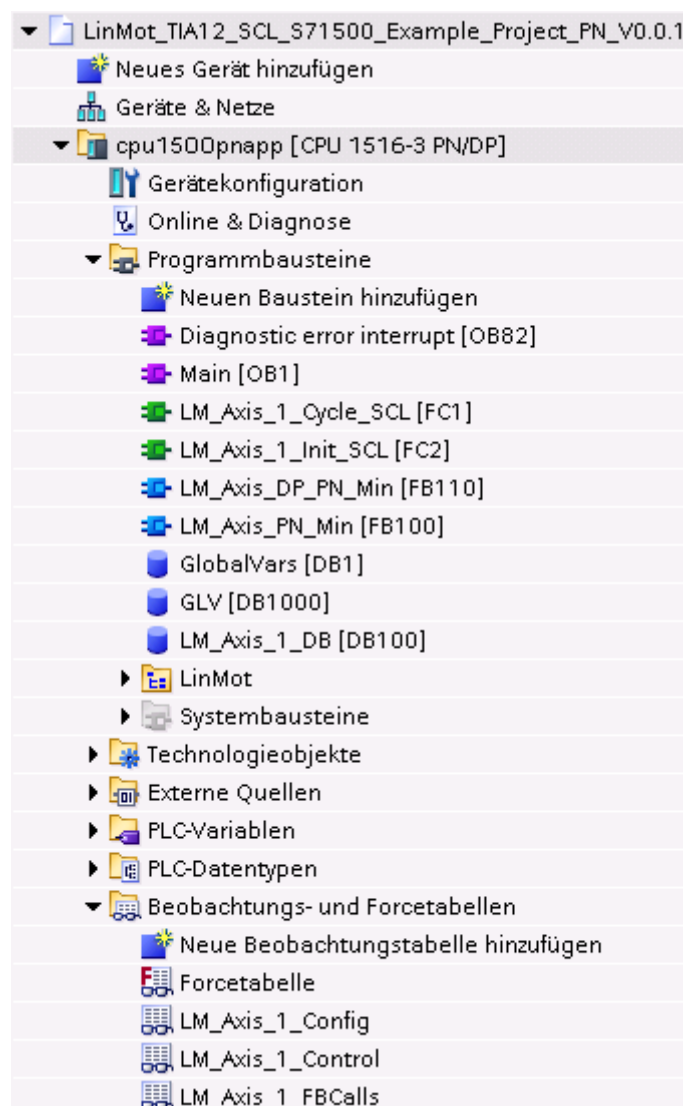
In the project all function blocks of the library for the specific CPU are included in the project library. But only the following are called in FB100 (LM\_Axis\_PN\_Min):

- |         |                   |                                |
|---------|-------------------|--------------------------------|
| • FB118 | LMct_RdAxisCom_PN | Read axis data from bus        |
| • FB119 | LMct_WrAxisCom_PN | Write axis data to bus         |
| • FB122 | LMct_AxisControl  | Axis control function block    |
| • FB123 | LMmt_MoveAbs      | Move absolute                  |
| • FB124 | LMmt_MoveRel      | Move relative                  |
| • FB126 | LMmt_Stop         | Stop axis                      |
| • FB200 | LMcf_ParaAccess   | Access parameters of the drive |

Additionally, to FB100 (DB100) in OB1 the two functions FC1 (LM\_Axis\_1\_Cycle\_SCL) and FC2 (LM\_Axis\_1\_Init\_SCL) are called:

- FC1 is an example of how to move between four positions.
- FC2 shows the initialisation and error acknowledge of the axis.

Included as well are some watch tables to control the above-mentioned functions and function blocks.





**6.2 CPU S7-1500 (Profibus), LinMot\_TIA\_SCL\_S71500\_Example\_Project\_DP\_...zip**

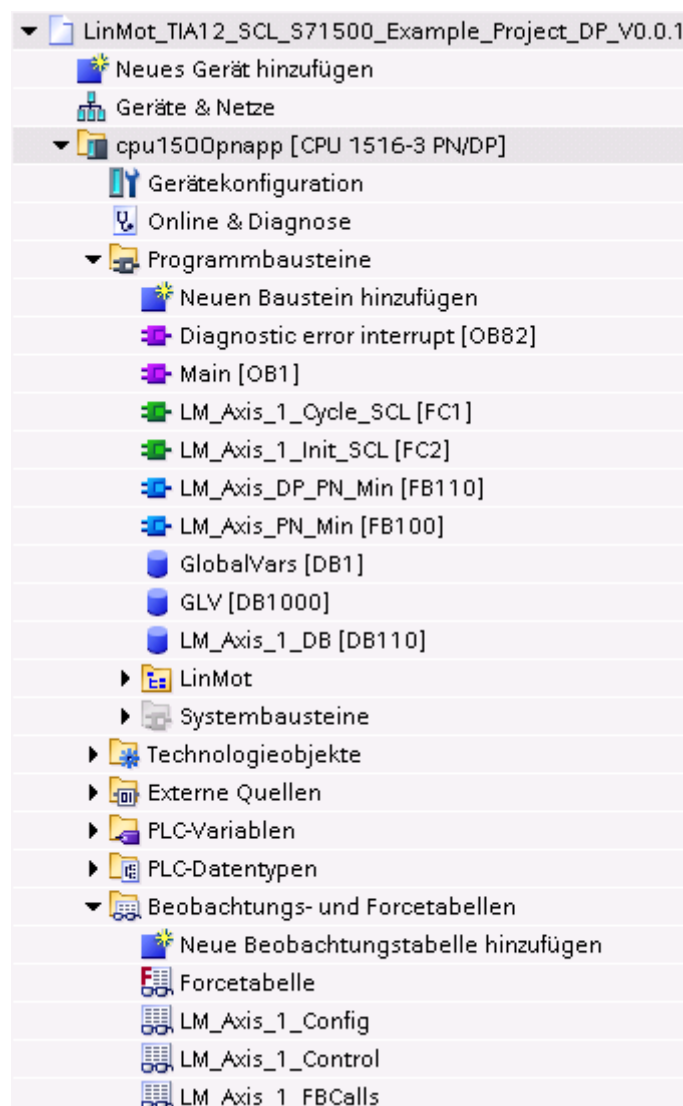
In the project all function blocks of the library for the specific CPU are included in the project library. But only the following are called in FB110 (LM\_Axis\_DP\_PN\_Min):

- |         |                      |                                |
|---------|----------------------|--------------------------------|
| • FB120 | LMct_RdAxisCom_DP_PN | Read axis data from bus        |
| • FB121 | LMct_WrAxisCom_DP_PN | Write axis data to bus         |
| • FB122 | LMct_AxisCtrl        | Axis control function block    |
| • FB123 | LMmt_MoveAbs         | Move absolute                  |
| • FB124 | LMmt_MoveRel         | Move relative                  |
| • FB126 | LMmt_Stop            | Stop axis                      |
| • FB200 | LMcf_ParaAccess      | Access parameters of the drive |

Additionally, to FB110 (DB110) in OB1 the two functions FC1 (LM\_Axis\_1\_Cycle\_SCL) and FC2 (LM\_Axis\_1\_Init\_SCL) are called:

- FC1 is an example of how to move between four positions.
- FC2 shows the initialisation and error acknowledge of the axis.

Included as well are some watch tables to control the above-mentioned functions and function blocks.



### 6.3 CPU S7-1200 (Profinet), LinMot\_TIA\_SCL\_S71200\_Example\_Project\_PN\_...zip

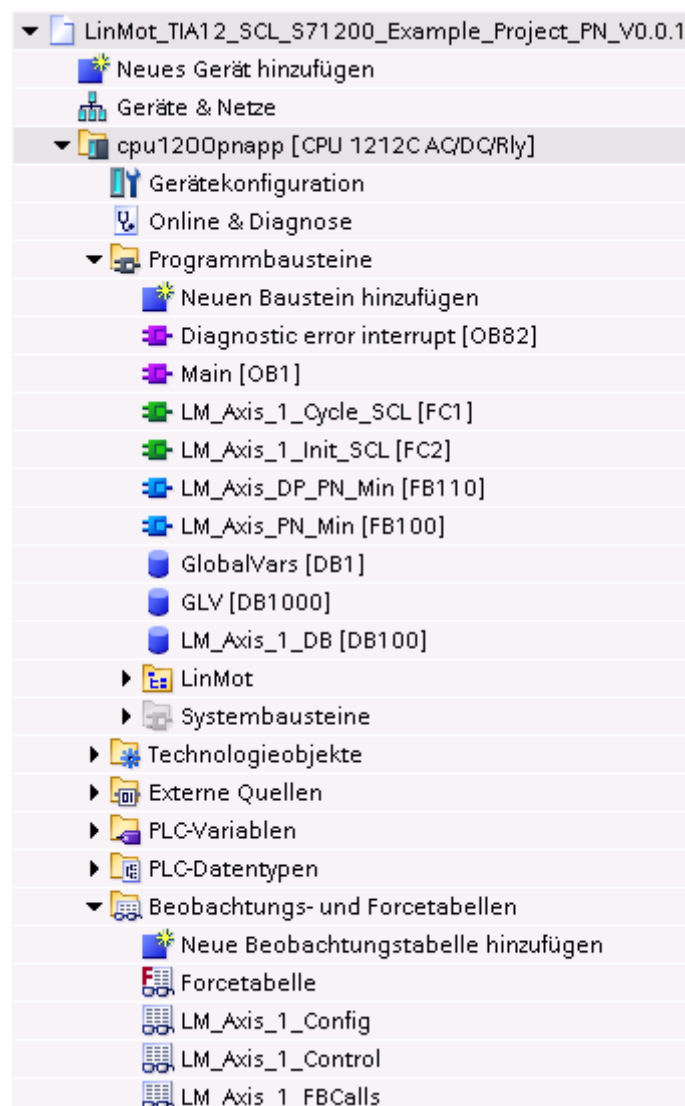
In the project all function blocks of the library for the specific CPU are included in the project library. But only the following are called in FB100 (LM\_Axis\_PN\_Min):

- |         |                   |                                |
|---------|-------------------|--------------------------------|
| • FB118 | LMct_RdAxisCom_PN | Read axis data from bus        |
| • FB119 | LMct_WrAxisCom_PN | Write axis data to bus         |
| • FB122 | LMct_AxisControl  | Axis control function block    |
| • FB123 | LMmt_MoveAbs      | Move absolute                  |
| • FB124 | LMmt_MoveRel      | Move relative                  |
| • FB126 | LMmt_Stop         | Stop axis                      |
| • FB200 | LMcf_ParaAccess   | Access parameters of the drive |

Additionally, to FB100 (DB100) in OB1 the two functions FC1 (LM\_Axis\_1\_Cycle\_SCL) and FC2 (LM\_Axis\_1\_Init\_SCL) are called:

- FC1 is an example of how to move between four positions.
- FC2 shows the initialisation and error acknowledge of the axis.

Included as well are some watch tables to control the above-mentioned functions and function blocks.



[illegible]

**Contact & Support****SCHWEIZ****NTI AG**

Bodenaeckerstrasse 2  
CH-8957 Spreitenbach

Sales and Administration:

+41 56 419 91 91  
[office@linmot.com](mailto:office@linmot.com)

Tech. Support:

+41 56 544 71 00  
[support@linmot.com](mailto:support@linmot.com)  
<http://www.linmot.com/support>

Tech. Support (Skype):

<skype:support.linmot>

Fax:

+41-(0)56-419 91 92

Web:

<http://www.linmot.com/>

**USA****LinMot USA Inc.**

204 E Morrissey Dr.  
Elkhorn, WI 53121  
USA

Sales and Administration:

877-546-3270  
262-743-2555

Tech. Support:

877-804-0718  
262-743-1284  
[usasupport@linmot.com](mailto:usasupport@linmot.com)

Fax:

800-463-8708  
262-723-6688

E-Mail:

[usasales@linmot.com](mailto:usasales@linmot.com)

Web:

<http://www.linmot-usa.com/>

Please visit <http://www.linmot.com/contact> to find the distribution close to you.

Smart solutions are...

