



Subset of the

Technical Specification

PLCopen - Technical Committee 2 – Task Force

Function blocks for motion control

Version 1.1

Appendix A :

Compliance Procedure and Compliance List

DISCLAIMER OF WARRANTIES

THIS DOCUMENT IS PROVIDED ON AN “AS IS” BASIS AND MAY BE SUBJECT TO FUTURE ADDITIONS, MODIFICATIONS, OR CORRECTIONS. PLCOPEN HEREBY DISCLAIMS ALL WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, FOR THIS DOCUMENT. IN NO EVENT WILL PLCOPEN BE RESPONSIBLE FOR ANY LOSS OR DAMAGE ARISING OUT OR RESULTING FROM ANY DEFECT, ERROR OR OMISSION IN THIS DOCUMENT OR FROM ANYONE’S USE OF OR RELIANCE ON THIS DOCUMENT.

July 2005

Appendix A. Compliance Procedure and Compliance List

Listed in this Appendix are the requirements for the compliance statement from the supplier of the Motion Control Function Blocks. The compliance statement consists of two main groups: supported data types (see Appendix A 2 Supported Data types) and supported Function Blocks, in combination with the applicable inputs and outputs (see Appendix A 3 Overview of the Function Blocks and its paragraphs). The supplier is required fill out the tables for the used data types and Function Blocks, according to their product, committing their support to the specification.

By submitting these tables to PLCopen, and after approval by PLCopen, the list will be published on the PLCopen website, www.plcopen.org , as well as a shortform overview, as specified in Appendix A 2 Supported Data types and Appendix A 3 Overview of the Function Blocks.

In addition to this approval, the supplier is granted access and usage rights of the PLCopen Motion Control logo, as described in chapter Appendix A 4 The PLCopen Motion Control Logo and Its Usage.

Data types

The data type REAL listed in the Function Blocks and parameters (e.g. for velocity, acceleration, distance, etc.) may be exchanged to SINT, INT, DINT or LREAL without to be seen as incompliant to this standard, as long as they are consistent for the whole set of Function Blocks and parameters.

Implementation allows the extension of data types as long as the basic data type is kept. For example: WORD may be changed to DWORD, but not to REAL.

Function Blocks and Inputs and Outputs

An implementation which claims compliance with this PLCopen specification shall offer a set of Function Blocks for motion control, meaning one or more Function Blocks, with at least the **basic** input and output variables, marked as “**B**” in the tables. These inputs and outputs have to be supported to be compliant.

For higher-level systems and future extensions any subset of the **extended** input and output variables, marked as “**E**” in the tables can be implemented.

Vendor specific additions are marked with “**V**”, and can be listed as such in the supplier documentation.

- Basic input/output variables are mandatory	Marked in the tables with the letter “ B ”
- Extended input/output variables are optional	Marked in the tables with the letter “ E ”
- Vendor Specific additions	Marked in the vendor’s compliance documentation with “ V ”

All the vendor specific items will not be listed in the comparison table on the PLCopen website, but in the detailed vendor specific list, which also is published.

All vendor specific in- and outputs of all FBs must be listed in the certification list of the supplier. With this, the certification listing from a supplier describes all the I/Os of the relevant FBs, including vendor-specific extensions, and thus showing the complete FBs as used by the supplier.

Appendix A 1. Statement of Supplier

Supplier name	Bosch Rexroth AG, Drives & Controls
Supplier address	Bgm.- Dr.- Nebel - Str. 2
City	97816 Lohr am Main
Country	Germany
Telephone	09352 40-4337
Fax	09352 40-34337
Email address	wolfgang.czech@boschrexroth.de
Product Name	ML PLCopen.library
Product version	V10.6.0,0
Release date	December 2009

I hereby state that the following tables as filled out and submitted do match our product as well as the accompanying user manual, as stated above.

Name of representation (person): Wolfgang Czech

Date of signature (dd/mm/yyyy): 11/01/2010

Signature:

Appendix A 2. Supported Data types

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL	Y	
INT		DINT
WORD		DINT
REAL	Y	
ENUM	Y	

Table 1: Supported datatypes

Within the specification the following derived datatypes are defined. Which structure is used in this system:

Derived datatypes:	Where used	Supported	Which structure
Axis_Ref	Nearly all FBs	Y	<pre> TYPE AXIS_REF : STRUCT CntrlNo: CONTROLS := LOCAL_CNTRL; AxisNo: OBJECTS := AXIS_1; END_STRUCT END_TYPE </pre>
MC_Direction (extended)	MC_MoveAbsolute MC_MoveVelocity	N	
MC_TP_REF	MC_PositionProfile	N	
MC_TV_REF	MC_VelocityProfile	N	
MC_TA_REF	MC_AccelerationProfile	N	
MC_CAM_REF	MC_CamTableSelect	N	
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn	Y	
MC_StartMode (extended)	MC_CamIn	Y	
MC_BufferMode	Buffered FBs	N	

Table 2: Supported derived datatypes

Appendix A 3. Overview of the Function Blocks

Single Axis Function Blocks	Supported Yes / No	Comments (<= 48 char.)
MC_MoveAbsolute	Yes	
MC_MoveRelative	Yes	
MC_MoveAdditive	Yes	
MC_MoveSuperimposed	No	
MC_MoveVelocity	Yes	
MC_Home	No	
MC_Stop	Yes	
MC_Power	Yes	
MC_ReadStatus	Yes	
MC_ReadAxisError	Yes	
MC_Reset	Yes	
MC_ReadParameter	No	
MC_ReadBoolParameter	No	
MC_WriteParameter	No	
MC_WriteBoolParameter	No	
MC_ReadActualPosition	Yes	
MC_PositionProfile	No	
MC_VelocityProfile	No	
MC_AccelerationProfile	No	
Multi-Axis Function Blocks	Supported Yes / No	Comments (<= 48 char.)
MC_CamTableSelect	No	
MC_CamIn	Yes	
MC_CamOut	Yes	
MC_GearIn	Yes	
MC_GearOut	Yes	
MC_Phasing	No	

Table 3: Short overview of the Function Blocks

Appendix A 3.1 MoveAbsolute

If Supported	MC_MoveAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	N	
E	Direction	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	DINT
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.2 MoveRelative

If Supported	MC_MoveRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.3 MoveAdditive

If Supported	MC MoveAdditive	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.5 MoveVelocity

If Supported	MC MoveVelocity	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	N	
E	Direction	N	
E	BufferMode	N	
VAR_OUTPUT			
B	InVelocity	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.7 Stop

If Supported	MC_Stop	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	Deceleration	Y	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.8 Power

If Supported	MC_Power	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
E	Enable_Positive	N	
E	Enable_Negative	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Status	Y	
E	Busy	N	
E	Active	N	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.9 ReadStatus

If Supported	MC_ReadStatus	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	N	
E	Busy	N	
V	Done	Y	
V	Active	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific
B	Errorstop	Y	
B	Stopping	Y	
B	StandStill	Y	
B	DiscreteMotion	Y	
B	ContinuousMotion	Y	
E	SynchronizedMotion	Y	
E	Homing	Y	
B	Disabled	Y	
V	PresetMode	Y	
V	CoordinatedMotion	Y	
E	ConstantVelocity	N	
E	Accelerating	N	
E	Decelerating	N	

Appendix A 3.10 ReadAxisError

If Supported	MC_ReadAxisError	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
	Enable	Y	
VAR_OUTPUT			
B	Valid	N	
E	Busy	N	
V	Done	Y	
V	Active	Y	
V	AxisErrorID	Y	
B	Error	Y	
B	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.11 Reset

If Supported	MC_Reset	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
B	Error	Y	
B	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.16 ReadActualPosition

If Supported	MC_ReadActualPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	N	
E	Busy	N	
V	Done	Y	
V	Active	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific
B	Position	Y	

Appendix A 3.21 CamIn

If Supported	MC_CamIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
V	RatioNumerator	Y	
V	RatioDenominator	Y	
V	MasterFineAdjust	Y	
V	CamShaftDistance	Y	
V	SyncMode	Y	
E	MasterOffset	N	
E	SlaveOffset	N	
E	MasterScaling	N	
E	SlaveScaling	N	
E	StartMode	Y	
E	CamTableID	Y	
E	BufferMode	N	
VAR_OUTPUT			
B	InSync	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific
V	CamSctiveID	Y	
E	EndOfProfile	N	

Appendix A 3.22 CamOut

If Supported	MC_CamOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	N	
V	Active	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.23 GearIn

If Supported	MC_GearIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
B	RatioNumerator	Y	
B	RatioDenominator	Y	
V	MasterFineadjust	Y	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	InGear	Y	
E	Busy	N	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 3.24 GearOut

If Supported	MC_GearOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
V	Active	Y	
E	Busy	N	
B	Error	Y	
E	ErrorID	Y	
V	ErrorIdent	Y	Errorstruct, Vendor specific

Appendix A 4. The PLCopen Motion Control Logo and Its Usage

For quick identification of compliant products, PLCopen has developed a logo for the motion control Function Blocks:



Figure 1: The PLCopen Motion Control Logo

This motion control logo is owned and trademarked by PLCopen.

In order to use this logo free-of-charge, the relevant company has to fulfill all the following requirements:

1. the company has to be a voting member of PLCopen;
2. the company has to comply with the existing specification, as specified by the PLCopen Task Force Motion Control, and as published by PLCopen, and of which this statement is a part;
3. this compliance application is provided in written form by the company to PLCopen, clearly stating the applicable software package and the supporting elements of all the specified tables, as specified in the document itself;
4. in case of non-fulfillment, which has to be decided by PLCopen, the company will receive a written statement concerning this from PLCopen. The company will have a one month period to either adopt their software package in such a way that it complies, represented by the issuing of a new compliance statement, or remove all reference to the specification, including the use of the logo, from all their specification, be it technical or promotional material;
5. the logo has to be used as is - meaning the full logo. It may be altered in size providing the original scale and color setting is kept.
6. the logo has to be used in the context of Motion Control.