

**Subset of the**

**Technical Specification**

**PLCopen - Technical Committee 2 – Task Force**

**Function blocks for motion control**

**Version 1.0**

**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.



### **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 datatypes (see Appendix A 2 Supported Datatypes) 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 has to fill out the tables for the used datatypes 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](http://www.plcopen.org), as well as a shortform overview, as specified in Appendix A 2 Supported Datatypes and Appendix A 3 Overview of the Function Blocks here below.

In addition to this approval, the supplier gets 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.

#### **Datatypes**

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 to extend 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, 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.

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

### Appendix A 1. Statement of Supplier

Supplier name	Baumüller Nürnberg GmbH
Supplier address	Ostendstrasse 80-90
City	90482 Nürnberg
Country	Germany
Telephone	+49(0)911/5432-131
Fax	+49(0)911/5432-430
Email address	r.klein@baumueller.de
Product Name	Motion Control Library
Product version	20bd00
Release date	Nov. 2002

I herewith 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):

Ralph Klein

Date of signature (dd/mm/yyyy):

22.11.02

Signature:

### Appendix A 2. Supported Datatypes

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

**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	all FBs	Yes	
MC_Direction (extended)	MC_MoveAbsolute MC_MoveVelocity	No	
MC_TP_REF	MC_PositionProfile	Yes	
MC_TV_REF	MC_VelocityProfile	Yes	
MC_TA_REF	MC_AccelerationProfile	No	
MC_CAM_REF	MC_CamTableSelect	No	
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn	No	
MC_StartMode (extended)	MC_CamIn	No	

**Table 2: Supported derived datatypes**

### Appendix A 3. Overview of the Function Blocks

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

**Table 3: Short overview of the Function Blocks**

### Appendix A 6.1 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	Y	INT SHORTEST WAY
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.2 MoveRelative

If Supported	MC_MoveRelative	Supported 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	
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.3 MoveAdditive

If Supported	MC_MoveAdditive	Supported 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	
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.4 MoveSuperimposed

If Supported	MC_MoveSuperimposed	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	N	
VAR_INPUT			
B	Execute	N	
B	Distance	N	
E	VelocityDiff	N	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
VAR_OUTPUT			
B	Done	N	
B	Busy	N	
E	CommandAborted	N	
B	Error	N	
E	ErrorID	N	

### Appendix A 6.5 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	Y	INT POSITIVE_DIRECTION NEGATIVE_DIRECTION
VAR_OUTPUT			
B	InVelocity	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.6 Home

If Supported	MC_Home	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	



### Appendix A 6.7 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	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.8 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	
VAR_OUTPUT			
B	Status	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.9 ReadStatus

If Supported	MC_ReadStatus	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	
B	Errorstop	Y	
B	Stopping	Y	
B	StandStill	Y	
B	DiscreteMotion	Y	
B	ContinuousMotion	Y	
E	SynchronizedMotion	N	
E	Homing	Y	
E	ConstantVelocity	N	
E	Accelerating	N	
E	Decelerating	N	

### Appendix A 6.10 ReadAxisError

If Supported	MC_ReadAxisError	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
	Enable	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
B	ErrorID	Y	

### Appendix A 6.11 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	
B	Error	Y	
B	ErrorID	Y	

### Appendix A 6.12 ReadParameter

If Supported	MC_ReadParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
B	ParameterNumber	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	
B	Value	Y	

### Appendix A 6.13 ReadBoolParameter

If Supported	MC_ReadBoolParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
B	ParameterNumber	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	
B	Value	Y	

Name	B/E	R/W	Sup. Y/N	Comments
CommandedPosition	B	R	Y	
SWLimitPos	E	R/W	N	available as system function
SWLimitNeg	E	R/W	N	available as system function
EnableLimitPos	E	R/W	N	available as system function
EnableLimitNeg	E	R/W	N	available as system function
EnablePosLagMonitoring	E	R/W	N	available as system function
MaxPositionLag	E	R/W	N	available as system function
MaxVelocitySystem	E	R	N	available as system function
MaxVelocityAppl	B	R/W	Y	
ActualVelocity	B	R	Y	
CommandedVelocity	B	R	Y	
MaxAccelerationSystem	E	R	N	
MaxAccelerationAppl	E	R/W	N	
MaxDecelerationSystem	E	R	N	
MaxDecelerationAppl	E	R/W	N	
MaxJerk	E	R/W	N	

Table 4: Parameters for ReadParameter and WriteParameter

### Appendix A 6.14 WriteParameter

If Supported	MC_WriteParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	ParameterNumber	Y	
B	Value	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.15 WriteBoolParameter

If Supported	MC_WriteBoolParameter	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	ParameterNumber	Y	
B	Value	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.16 ReadActualPosition

If Supported	MC_ReadActualPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Done	Y	
B	Error	Y	
E	ErrorID	Y	
B	Position	Y	

### Appendix A 6.17 PositionProfile

If Supported	MC_PositionProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
B	TimePosition	Y	
VAR_INPUT			
B	Execute	Y	
B	ArraySize	Y	
E	Scale	N	
E	Offset	N	
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.18 VelocityProfile

If Supported	MC_VelocityProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
B	MC_TimeVelocity	Y	
VAR_INPUT			
B	Execute	Y	
B	ArraySize	Y	
E	Scale	N	
E	Offset	N	
VAR_OUTPUT			
B	Done	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.19 AccelerationProfile

If Supported	MC_AccelerationProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	N	
B	MC_TimeAcceleration	N	
VAR_INPUT			
B	Execute	N	
B	ArraySize	N	
E	Scale	N	
E	Offset	N	
VAR_OUTPUT			
B	Done	N	
E	CommandAborted	N	
B	Error	N	
E	ErrorID	N	

### Appendix A 6.20 CamTableSelect

If Supported	MC_CamTableSelect	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	N	
B	Slave	N	
B	CamTable	N	
VAR_INPUT			
B	Execute	N	
E	Periodic	N	
E	MasterAbsolute	N	
E	SlaveAbsolute	N	
VAR_OUTPUT			
B	Done	N	
B	Error	N	
E	ErrorID	N	
E	CamTableID	N	

### Appendix A 6.21 CamIn

If Supported	MC_CamIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	N	
B	Slave	N	
VAR_INPUT			
B	Execute	N	
E	MasterOffset	N	
E	SlaveOffset	N	
E	MasterScaling	N	
E	SlaveScaling	N	
E	StartMode	N	
E	CamTableID	N	
VAR_OUTPUT			
B	InSync	N	
E	CommandAborted	N	
B	Error	N	
E	ErrorID	N	
E	EndOfProfile	N	

### Appendix A 6.22 CamOut

If Supported	MC_CamOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	N	
VAR_INPUT			
B	Execute	N	
VAR_OUTPUT			
B	Done	N	
B	Error	N	
E	ErrorID	N	

### Appendix A 6.23 GearIn

If Supported	MC_GearIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	N	
B	Slave	N	
VAR_INPUT			
B	Execute	N	
B	RatioNumerator	N	
B	RatioDenominator	N	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
VAR_OUTPUT			
B	InGear	N	
E	CommandAborted	N	
B	Error	N	
E	ErrorID	N	

### Appendix A 6.24 GearOut

If Supported	MC_GearOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	N	
VAR_INPUT			
B	Execute	N	
VAR_OUTPUT			
B	Done	N	
B	Error	N	
E	ErrorID	N	

### Appendix A 6.25 Phasing

If Supported	MC_Phasing	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	N	
B	Slave	N	
VAR_INPUT			
B	Execute	N	
B	Phase	N	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
VAR_OUTPUT			
B	Done	N	
E	CommandAborted	N	
B	Error	N	
E	ErrorID	N	

### 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 to 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 is done 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 statement on this from PLCopen in written form. 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 as long as the original scale and color setting is kept.
6. the logo has to be used in the context of Motion Control.