



Subset of the

**Technical Paper**

**PLCopen Technical Committee 2 – Task Force**

**Function Blocks for motion control:**

**Part 4 –Coordinated Motion**

**Version 1.0, Published**

**Appendix I**

**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 OF OR RESULTING FROM ANY DEFECT, ERROR OR OMISSION IN THIS DOCUMENT OR FROM ANYONE’S USE OF OR RELIANCE ON THIS DOCUMENT.

Copyright © 2002 - 2008 by PLCopen. All rights reserved.

Date: December 3, 2008

Total number of pages: 20

## **Appendix 1. 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 1.2 Supported Data types) and supported Function Blocks, in combination with the applicable inputs and outputs see (Appendix 1.2 Supported Data types 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](http://www.plcopen.org) , as well as a shortform overview, as specified in Appendix 1.5 Short overview of the Functions 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 1.6 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.

- |  |   |
|--|---|
| - <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> ” |

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 1.1. Statement of Supplier**

Supplier name	Lenze Automation GmbH	
Supplier address	Hans-Lenze-Straße 1, D-31855 Aerzen	
City	Aerzen (Hameln)	
Country	Deutschland	
Telephone	+49 531 8017826	
Fax	+49 531 8017820	
Email address	Johannes.Kuehn@lenze.com	
Product Name	FAST Motion (PLC Designer)	
Product version	3.x	
Release date	04.08.2017	

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): Dr. Johannes B. Kühn

Date of signature (dd/mm/yyyy): 21.11.2017

Signature:



**Lenze**  
Lenze Automation GmbH  
Postfach 10 13 52, D-31763 Hameln  
Hans-Lenze-Straße 1, D-31855 Aerzen

## Appendix 1.2. Supported Data types

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

**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
AXES_GROUP_REF	Nearly all FBs	Yes	
IDENT_IN_GROUP_REF	MC_AddAxisToGroup MC_RemoveAxisFromGroup	Yes	
MC_BUFFER_MODE	In all buffered FBs	Yes	
MC_KIN_REF	MC_SetKinTransform MC_ReadKinTransform	Yes	
MC_EXECUTION_MODE	MC_SetKinTransform	No	
MC_COORD_REF	MC_SetCoordinateTransformation	Yes	
MC_GROUP_BUFFER_MODE	MC_MoveLinearAbsolute MC_MoveCircularAbsolute	No	Used MC_BUFFER_MODE instead
MC_TRANSITION_MODE	MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircularAbsolute MC_MoveCircularRelative	Yes	
MC_CIRC_PATHCHOICE	MC_MoveCircularAbsolute MC_MoveCircularRelative	Yes	
MC_PATH_DATA_REF MC_PATH_REF	MC_PathSelect MC_MovePath	Yes	

**Table 2: Supported derived datatypes**

## Appendix 1.3. Supported Buffer Modes

No.	MC_BUFFER_MODE	Supported
0	Aborting	Yes
1	Buffered	Yes
2	BlendingLow	Yes
3	BlendingPrevious	Yes
4	BlendingNext	Yes
5	BlendingHigh	Yes

**Table 3: Overview of buffer modes**

## Appendix 1.4. Supported Transition Modes

No.	MC_TRANSITION_MODE	Supported
0	TMNone	Yes
1	TMMaxVelocity	No
2	TMDefinedVelocity	No
3	TMCornerDistance	Yes
4	TMMaxCornerDeviation	No
5 - 9	Reserved by PLCopen	
10 - ...	Supplier specific modes	

**Table 4: Overview of available transition modes**

**Appendix 1.5. Short overview of the Function Blocks**

<b>Coordinated Function Blocks</b>	<b>Supported Yes / No</b>	<b>Comments (&lt;= 48 char.)</b>
MC_AddAxisToGroup	No	
MC_RemoveAxisFromGroup	No	
MC_UngroupAllAxes	No	
MC_GroupReadConfiguration	No	
MC_GroupEnable	Yes	
MC_GroupDisable	Yes	
MC_GroupHome	Yes	
MC_SetKinTransform	Yes	
MC_SetCartesianTransform	Yes	
MC_SetCoordinateTransform	Yes	
MC_ReadKinTransform	No	
MC_ReadCartesianTransform	Yes	
MC_ReadCoordinateTransform	Yes	
MC_GroupSetPosition	Yes	
MC_GroupReadActualPosition	Yes	
MC_GroupReadActualVelocity	Yes	
MC_GroupReadActualAcceleration	Yes	
MC_GroupStop	Yes	
MC_GroupHalt	No	
MC_GroupInterrupt	No	
MC_GroupContinue	No	
MC_GroupReadStatus	Yes	
MC_GroupReadError	Yes	
MC_GroupReset	Yes	
MC_MoveLinearAbsolute	Yes	
MC_MoveLinearRelative	Yes	
MC_MoveCircularAbsolute	Yes	
MC_MoveCircularRelative	Yes	
MC_MoveDirectAbsolute	Yes	
MC_MoveDirectRelative	Yes	
MC_PathSelect	No	
MC_MovePath	Yes	
MC_GroupSetOverride	No	
<b>Coordinated</b>	<b>Supported Yes / No</b>	<b>Comments (&lt;= 48 char.)</b>
MC_SyncAxisToGroup	No	
MC_SyncGroupToAxis	No	
MC_SetDynCoordTransform	No	
MC_TrackConveyorbelt	Yes	
MC_TrackRotaryTable	No	

**Table 5: Short overview of the Function Blocks**

### Appendix A 5.1. MC\_AddAxisToGroup

If Supported	MC_AddAxisToGroup	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
B	Axis		
VAR_INPUT			
B	Execute		
E	IdentInGroup		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.2. MC\_RemoveAxisFromGroup

If Supported	MC_RemoveAxisFromGroup	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Execute		
E	IdentInGroup		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.3. MC\_UngroupAllAxes

If Supported	MC_UngroupAllAxes	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Execute		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.4. MC\_GroupReadConfiguration

If Supported	MC_GroupReadConfiguration	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Enable		
B	IdentInGroup		
E	CoordSystem		
VAR_OUTPUT			
B	Axis		
B	Valid		
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.5. MC\_GroupEnable

If Supported	MC_GroupEnable	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.6. MC\_GroupDisable

If Supported	MC_GroupDisable	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.7. MC\_GroupHome

If Supported	MC_GroupHome	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	CoordSystem	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.8. MC\_SetKinTransform

If Supported	MC_SetKinTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
E	KinTransform	Y	
E	ExecutionMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.9. MC\_SetCartesianTransform

If Supported	MC_SetCartesianTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	TransX	Y	
B	TransY	Y	
B	TransZ	Y	
B	RotAngle1	Y	
B	RotAngle2	Y	
B	RotAngle3	Y	
E	ExecutionMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.10. MC\_SetCoordinateTransform

If Supported	MC_SetCoordinateTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
E	CoordTransform	Y	
E	ExecutionMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	



### Appendix A 5.11. MC\_ReadKinTransform

If Supported	MC_ReadKinTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	KinTransform		
B	Error		
E	ErrorID		

### Appendix A 5.12. MC\_ReadCartesianTransform

If Supported	MC_ReadCartesianTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	TransX	Y	
B	TransY	Y	
B	TransZ	Y	
B	RotAngle1	Y	
B	RotAngle2	Y	
B	RotAngle3	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.13. MC\_ReadCoordinateTransform

If Supported	MC_ReadCoordinateTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	CoordTransform	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.14. MC\_GroupSetPosition

If Supported	MC_GroupSetPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Relative	Y	
E	CoordSystem	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.15. MC\_GroupReadActualPosition

If Supported	MC_GroupReadActualPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
E	CoordSystem		
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
B	Position	Y	

### Appendix A 5.16. MC\_GroupReadActualVelocity

If Supported	MC_GroupReadActualVelocity	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
E	CoordSystem	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
B	Velocity	Y	
E	PathVelocity	Y	

### Appendix A 5.17. MC\_GroupReadActualAcceleration

If Supported	MC_GroupReadActualAcceleration	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
E	CoordSystem	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
B	Acceleration	Y	
E	Path Acceleration	Y	

### Appendix A 5.18. MC\_GroupStop

If Supported	MC_GroupStop	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.19. MC\_GroupHalt

If Supported	MC_GroupHalt	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Execute		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix A 5.20. MC\_GroupInterrupt

If Supported	MC_GroupInterrupt	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Execute		
E	Deceleration		
E	Jerk		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Comman Aborted		
B	Error		
E	ErrorID		

### Appendix A 5.21. MC\_GroupContinue

If Supported	MC_GroupContinue	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Execute		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Comman Aborted		
B	Error		
E	ErrorID		

### Appendix A 5.22. MC\_GroupReadStatus

If Supported	MC_GroupReadStatus	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	GroupMoving	Y	
B	GroupHoming	Y	
B	GroupErrorStop	Y	
B	GroupStandby	Y	
B	GroupStopping	Y	
B	GroupDisabled	Y	
E	ConstantVelocity	Y	
E	Accelerating	Y	
E	Decelerating	Y	
E	InPosition	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.23. MC\_GroupReadError

If Supported	MC_GroupReadError	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
B	GroupErrorID	Y	

### Appendix A 5.24. MC\_GroupReset

If Supported	MC_GroupReset	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.25. MC\_MoveLinearAbsolute

If Supported	MC_MoveLinearAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

**Appendix A 5.26. MC\_MoveLinearRelative**

If Supported	MC_MoveLinearRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

**Appendix A 5.27. MC\_MoveCircularAbsolute**

If Supported	MC_MoveCircularAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	CircMode	Y	
B	AuxPoint	Y	
B	EndPoint	Y	
E	PathChoice	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.28. MC\_MoveCircularRelative

If Supported	MC_MoveCircularRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	CircMode	Y	
B	AuxPoint	Y	
B	EndPoint	Y	
E	PathChoice	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.29. MC\_MoveDirectAbsolute

If Supported	MC_MoveDirectAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.30. MC\_MoveDirectRelative

If Supported	MC_MoveDirectRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.31. MC\_PathSelect

If Supported	MC_PathSelect	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
B	PathData		
B	PathDescription		
VAR_INPUT			
B	Execute		
E	CoordSystem		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.32. MC\_MovePath

If Supported	MC_MovePath	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
B	PathData	N	Used MC_PATH_REF instead
VAR_INPUT			
B	Execute	Y	
E	CoordSystem	Y	
E	BufferMode	Y	
E	TransitionMode	Y	
E	TransitionParameter	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	



### Appendix A 5.33. MC\_GroupSetOverride

If Supported	MC_GroupSetOverride	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
VAR_INPUT			
B	Enable		
B	VelFactor		
E	AccFactor		
E	JerkFactor		
VAR_OUTPUT			
B	Enabled	BOOL	
E	Busy		
B	Error		
E	ErrorID		

### Appendix A 5.34. MC\_SyncAxisToGroup

If Supported	MC_SyncAxisToGroup	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
B	SlaveAxis		
VAR_INPUT			
B	Execute		
E	RatioNumerator		
E	RatioDenominator		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix A 5.35 MC\_SyncGroupToAxis

If Supported	MC_SyncGroupToAxis	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	AxesGroup		
B	PathData		
VAR_INPUT			
B	Execute		
E	Mode		
E	TuCNumerator		
E	TuCDenominator		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix A 5.36. MC\_SetDynCoordTransform

If Supported	MC_SetDynCoordTransform	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
B	MasterAxesGroup		
B	CoordTransform		
VAR_INPUT			
B	Execute		
E	Mode		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix A 5.37. MC\_TrackConveyorBelt

If Supported	MC_TrackConveyorBelt	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
E	ConveyorBelt	Y	
VAR_INPUT			
B	Execute	Y	
B	ConveyorBeltOrigin	Y	
E	InitialObjectPosition	Y	
E	CoordSystem	Y	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 5.38. MC\_TrackRotaryTable

If Supported	MC_TrackRotaryTable	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup		
E	RotaryTable		
VAR_INPUT			
B	Execute		
B	RotaryTableOrigin		
E	InitialObjectPosition		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
E	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

## Appendix 1.6. 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.