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

Date: December 3, 2008

Total number of pages: 22

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 , 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.

- **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 1.1.   Statement of Supplier

Supplier name	KINGSTAR
Supplier address	400 Fifth Avenue
City	Waltham, MA
Country	USA
Telephone	+1 781-996-4481
Fax	
Email address	kingstarsales@kingstar.com
Product Name	KingStar Soft Motion
Product version	3.1
Release date	28 APR 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): Daron Underwood, CTO

Date of signature (dd/mm/yyyy): 26/04/2017

Signature:

Appendix 1.2.   Supported Data types

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL	Yes	
INT	Yes	
WORD	Yes	
REAL	Yes	
ENUM	No	SINT

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	No	
MC_EXECUTION_MODE	MC_SetKinTransform	No	
MC_COORD_REF	MC_SetCoordinateTransformation	No	
MC_GROUP_BUFFER_MODE	MC_MoveLinearAbsolute MC_MoveCircularAbsolute	No	MC_BUFFER_MODE used
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	No	

Table 2: Supported derived datatypes

Appendix 1.3.   Supported Buffer Modes

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

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	No
4	TMMaxCornerDeviation	No
5 - 9	Reserved by PLCopen	No
10 -…	Supplier specific modes	No

Table 4: Overview of available transition modes

Appendix 1.5.　　　Short overview of the Function Blocks

Coordinated Function Blocks	Supported Yes / No	Comments (<= 48 char.)
MC_AddAxisToGroup	Yes	
MC_RemoveAxisFromGroup	Yes	
MC_UngroupAllAxes	Yes	
MC_GroupReadConfiguration	Yes	
MC_GroupEnable	Yes	
MC_GroupDisable	Yes	
MC_GroupHome	No	
MC_SetKinTransform	No	
MC_SetCartesianTransform	No	
MC_SetCoordinateTransform	No	
MC_ReadKinTransform	No	
MC_ReadCartesianTransform	No	
MC_ReadCoordinateTransform	No	
MC_GroupSetPosition	Yes	
MC_GroupReadActualPosition	Yes	
MC_GroupReadActualVelocity	Yes	
MC_GroupReadActualAcceleration	Yes	
MC_GroupStop	Yes	
MC_GroupHalt	Yes	
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	No	
MC_MoveDirectRelative	No	
MC_PathSelect	No	
MC_MovePath	No	
MC_GroupSetOverride	Yes	
Coordinated	**Supported Yes / No**	**Comments (<= 48 char.)**
MC_SyncAxisToGroup	No	
MC_SyncGroupToAxis	No	
MC_SetDynCoordTransform	No	
MC_TrackConveyorbelt	No	
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	Y	
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	IdentInGroup	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	

Appendix A 5.2. MC_RemoveAxixFromGroup

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

Appendix A 5.3. MC_UngroupAllAxes

If Supported	MC_UngroupAllAxes	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 5.4. MC_GroupReadConfiguration

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

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		
VAR_INPUT			
B	Execute		
B	Position		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

Appendix A 5.8. MC_SetKinTransform

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

Appendix A 5.9. MC_SetCartesianTransform

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

Appendix A 5.10. MC_SetCoordinateTransform

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

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		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	TransX		
B	TransY		
B	TransZ		
B	RotAngle1		
B	RotAngle2		
B	RotAngle3		
B	Error		
E	ErrorID		

Appendix A 5.13. MC_ReadCoordinateTransform

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

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	Y	Ignored ACS is used
E	BufferMode	Y	
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	Y	Ignored ACS is used
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	Ignored ACS is used
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	Ignored ACS is used
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	Removed to match the single axis MC_Stop usage
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	Y	
VAR_INPUT			
B	Execute	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

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	N	
E	Decelerating	N	
E	InPosition	N	
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	
V	ContinuousUpdate	Y	
B	Position	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	Ignored ACS is used
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	
V	ContinuousUpdate	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	CoordSystem	Y	Ignored ACS is used
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	
V	ContinuousUpdate	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	Ignored ACS is used
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	
V	ContinuousUpdate	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	Ignored ACS is used
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		
VAR_INPUT			
B	Execute		
B	Position		
E	CoordSystem		
E	BufferMode		
E	TransitionMode		
E	TransitionParameter		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

Appendix A 5.30. MC_MoveDirectRelative

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

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		
B	PathData		
VAR_INPUT			
B	Execute		
E	CoordSystem		
E	BufferMode		
E	TransitionMode		
E	TransitionParameter		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

Appendix A 5.33. MC_GroupSetOverride

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

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		
E	ConveyorBelt		
VAR_INPUT			
B	Execute		
B	ConveyorBeltOrigin		
E	InitialObjectPosition		
E	CoordSystem		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

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.