



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 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: 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.

1.1 Statement of Supplier

Supplier name	3S – Smart Software Solutions GmbH
Supplier address	Memminger Straße 151
City	87439 Kempten
Country	Germany
Telephone	+49-831-54031-10
Fax	+49-831-54031-50
Email address	info@codesys.com
Product Name	CODESYS SoftMotion CNC + Robotics
Product version	4.3.0.0
Release date	25.10.2017

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): Georg Seidel, Product Owner

Date of signature: 02.03.2018

Signature: GS

1.2 Supported Data types

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL	Y	
INT	Y	
WORD	Y	
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
AXES_GROUP_REF	Nearly all FBs	Y	AXIS_GROUP_REF_SM3
IDENT_IN_GROUP_REF	MC_AddAxisToGroup MC_RemoveAxisFromGroup	N	
MC_BUFFER_MODE	In all buffered FBs	Y	MC_BUFFER_MODE
MC_KIN_REF	MC_SetKinTransform MC_ReadKinTransform	Y	TRAFO.MC_KIN_REF_SM3
MC_EXECUTION_MODE	MC_SetKinTransform	Y	MC_EXECUTION_MODE
MC_COORD_REF	MC_SetCoordinateTransformation	Y	MC_COORD_REF
MC_GROUP_BUFFER_MODE	MC_MoveLinearAbsolute MC_MoveCircularAbsolute	N	MC_BUFFER_MODE
MC_TRANSITION_MODE	MC_MoveLinearAbsolute MC_MoveLinearRelative MC_MoveCircularAbsolute MC_MoveCircularRelative	Y	MC_TRANSITION_MODE
MC_CIRC_PATHCHOICE	MC_MoveCircularAbsolute MC_MoveCircularRelative	Y	MC_CIRC_PATHCHOICE
MC_PATH_DATA_REF MC_PATH_REF	MC_PathSelect MC_MovePath	N	

Table 2: Supported derived datatypes

1.3 Supported Buffer Modes

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

Table 3: Overview of buffer modes

1.4 Supported Transition Modes

No.	MC_TRANSITION_MODE	Supported
0	TMNone	Y
1	TMMaxVelocity	N (CODESYS supports TMStartVelocity)
2	TMDefinedVelocity	N
3	TMCornerDistance	Y
4	TMMaxCornerDeviation	N
5 - 9	Reserved by PLCopen	
10 - ...	Supplier specific modes	

Table 4: Overview of available transition modes

1.5 Short overview of the Function Blocks

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

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	N	Axis is always added at the end
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	

Appendix A 5.2. MC_RemoveAxisFromGroup

If Supported	MC_RemoveAxisFromGroup	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Execute	Y	
E	IdentInGroup	N	Always removes the last added axis
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 / not supported

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 / not supported

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	Y	
VAR_INPUT			
B	Execute	Y	
E	KinTransform	Y	
E	ExecutionMode	N	Always queued
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	N	
B	Error	Y	
E	ErrorID	Y	

Appendix A 5.9. MC_SetCartesianTransform / not supported

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	Y	
VAR_INPUT			
B	Execute	Y	
E	CoordTransform	Y	
E	ExecutionMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	N	
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	Y	
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	KinTransform	Y	
B	Error	Y	
E	ErrorID	Y	

Appendix A 5.12. MC_ReadCartesianTransform / not supported

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	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	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 5.15. MC_GroupReadActualPosition

If Supported	MC_GroupReadActualPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	AxesGroup	Y	
VAR_INPUT			
B	Enable	Y	
E	CoordSystem	Y	
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	N	Release in following Steps

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	N	Release in following Steps

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	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 5.20. MC_GroupInterrupt

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

Appendix A 5.21. MC_GroupContinue

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

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	N	Release in following Steps
E	Accelerating	N	Release in following Steps
E	Decelerating	N	Release in following Steps
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	
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 / not supported

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 / not supported

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 / not supported

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 / not supported

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	Y	
B	MasterAxesGroup	Y	
B	CoordTransform	Y	
VAR_INPUT			
B	Execute	Y	
E	Mode	Y	
E	CoordSystem	Y	
E	BufferMode	N	Always buffered
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

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	Always buffered
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	N	
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	Y	
E	RotaryTable	Y	
VAR_INPUT			
B	Execute	Y	
B	RotaryTableOrigin	Y	
E	InitialObjectPosition	Y	
E	CoordSystem	Y	
E	BufferMode	N	Always buffered
VAR_OUTPUT			
E	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	N	
B	Error	Y	
E	ErrorID	Y	

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.