



**Subset of the**

**Technical Specification**

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

**Function blocks for motion control**  
(Formerly Part 1 and Part 2)

**Version 2.0**

**Appendix B**

**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 OR CONDITIONS OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES, OR CONDITIONS OF TITLE, NON-INFRINGEMENT, 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 © 2009 – 2011 by PLCopen. All rights reserved.

March 17, 2011.

## Appendix B. 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 and supported Function Blocks, in combination with the applicable inputs and outputs. The supplier is required to 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 B 2 Supported Data types and Appendix B 3 Overview of the Function Blocks as below.

In addition to this approval, the supplier is granted access and usage rights of the PLCopen Motion Control logo, as described in Appendix B 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.

- |   |   |
|---|---|
| - <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 B 1. Statement of Supplier**

Supplier name	mitsubishi electric corporation
Supplier address	1-14, Yada-minami 5-chome, Higashi-ku
City	Nagoya
Country	Japan
Telephone	+81-52-712-2268
Fax	+81-52-712-3960
Email address	Ikawa.Yasushi@bk.MitsubishiElectric.co.jp
Product Name	iQ-R Series, RD78G4/RD78G8/RD78G16/RD78G32/RD78G64/RD78GHV/RD78GHW, Motion Module, GX Works3
Product version	1.00A
Release date	2019/05/07

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):  
Yasushi Ikawa,  
Director of Drive System Department, Nagoya Works

Date of signature (dd/mm/yyyy):  
19/03/2019

Signature:



## Appendix B 2. Supported Data types

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL	Y	called Bit
INT	Y	called Word[Signed]
WORD	Y	Word[Unsigned]/Bit String[16-bit]
REAL	Y	called FLOAT
ENUM	N	Supported as pre-defined INT labels
UINT	Y	called Word[Unsigned]/Bit String[16-bit]

**Table 1: Supported datatypes**

Within the specification the following derived datatypes are defined. Define which of these structures are used in this system:

Derived datatypes:	Where used	Supported	Which structure
AXIS_REF	Nearly all FBs	Y	Structured Data Types
MC_DIRECTION (extended)	MC_MoveAbsolute MC_MoveVelocity MC_TorqueControl MC_MoveContinuousAbsolute	Y Y Y N	Supported as pre-defined INT label with the following values: mcPositiveDirection 1 mcNegativeDirection 2 mcShortestWay 3 mcCurrentDirection 4
MC_TP_REF	MC_PositionProfile	N	
MC_TV_REF	MC_VelocityProfile	N	
MC_TA_REF	MC_AccelerationProfile	N	
MC_CAM_REF	MC_CamTableSelect	Y	Structured Data Types
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn	Y Y	Structured Data Types
MC_START_MODE (extended)	MC_CamIn MC_CamTableSelect	Y N	Supported as pre-defined INT label with the following values: mcImmediate 0 mcAbsolute 1 mcRelative 2
MC_BUFFER_MODE	Buffered FBs	Y	Supported as pre-defined INT with the following values: mcAborting 0 mcBuffered 1 mcBlendingLow 2 mcBlendingPrevious 3 mcBlendingNext 4 mcBlendingHigh 5
MC_EXECUTION_MODE	MC_SetPosition MC_WriteParameter MC_WriteBoolParameter MC_WriteDigitalOutput MC_CamTableSelect	Y N N N Y	Supported as pre-defined INT with the following values: mcImmediately 0 mcQueued 1 mcNextExecute 2 mcSpeculatively 3
MC_SOURCE	MC_ReadMotionState MC_CamIn MC_GearIn MC_GearInPos MC_CombineAxes MC_DigitalCamSwitch	N Y Y N Y N	Supported as pre-defined INT with the following values: mcSetValue 1 mcActualValue 2 mcLatestSetValue 101 mcLatestActualValue 102
MC_SYNC_MODE	MC_GearInPos	N	
MC_COMBINE_MODE	MC_CombineAxes	Y	Supported as pre-defined INT with the following values: mcAddAxes 0 mcSubAxes 1
MC_TRIGGER_REF	MC_TouchProbe MC_AbortTrigger	N N	
MC_INPUT_REF	MC_ReadDigitalInput	N	
MC_OUTPUT_REF	MC_DigitalCamSwitch	N	

	MC_ReadDigitalOutput	N	
	MC_WriteDigitalOutput	N	
MC_CAMSWITCH_REF	MC_DigitalCamSwitch	N	
MC_TRACK_REF	MC_DigitalCamSwitch	N	

**Table 2: Supported derived datatypes**

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

Single Axis Function Blocks	Supported as V1.0/ V1.1/ V2.0 or Not	Comments (<= 48 char.)
MC_Power	V2.0	
MC_Home	V2.0	
MC_Stop	V2.0	
MC_Halt	Not	
MC_MoveAbsolute	V2.0	
MC_MoveRelative	V2.0	
MC_MoveAdditive	Not	
MC_MoveSuperimposed	Not	
MC_HaltSuperimposed	Not	
MC_MoveVelocity	V2.0	
MC_MoveContinuousAbsolute	Not	
MC_MoveContinuousRelative	Not	
MC_TorqueControl	V2.0	
MC_PositionProfile	Not	
MC_VelocityProfile	Not	
MC_AccelerationProfile	Not	
MC_SetPosition	V2.0	
MC_SetOverride	Not	
MC_ReadParameter & MC_ReadBoolParameter	Not	
MC_WriteParameter & MC_WriteBoolParameter	Not	
MC_ReadDigitalInput	Not	
MC_ReadDigitalOutput	Not	
MC_WriteDigitalOutput	Not	
MC_ReadActualPosition	Not	
MC_ReadActualVelocity	Not	
MC_ReadActualTorque	Not	
MC_ReadStatus	Not	
MC_ReadMotionState	Not	
MC_ReadAxisInfo	Not	
MC_ReadAxisError	Not	
MC_Reset	Not	
MC_DigitalCamSwitch	Not	
MC_TouchProbe	Not	
MC_AbortTrigger	Not	
Multi-Axis Function Blocks	Supported as V1.0/ V1.1/ V2.0 or Not	Comments (<= 48 char.)
MC_CamTableSelect	V2.0	
MC_CamIn	V2.0	
MC_CamOut	Not	
MC_GearIn	V2.0	
MC_GearOut	Not	
MC_GearInPos	Not	
MC_PhasingAbsolute	Not	
MC_PhasingRelative	Not	
MC_CombineAxes	V2.0	

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

### Appendix B 3.1 MC\_Power

If Supported	MC_Power	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Enable	Y	
E	EnablePositive	N	
E	EnableNegative	N	
V	ServoON	Y	Datatype is BOOL Specifies the servo ON request
VAR_OUTPUT			
B	Status	Y	
V	ReadyStatus	Y	Datatype is BOOL Indicates the ready ON/OFF status
V	Busy	Y	Datatype is BOOL Indicates that the FB is in execution
E	Valid	N	
B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.2 MC\_Home

If Supported	MC_Home	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
V	AbsSwitch	Y	Structured Data Type MC_INPUT_REF Set the proximity dog signal that is transmitted to the slave device in the driver homing method.
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.3 MC\_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	Y	
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
V	Active	Y	Datatype is BOOL Indicates that the FB is controlling the axis.
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	





### Appendix B 3.4 MC\_Halt

If Supported	MC_Halt	Sup. Y/N	
VAR_IN_OUT			
B	Axis		
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 B 3.5 MC\_MoveAbsolute

If Supported	MC_MoveAbsolute	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
B	Position	Y	
B	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
B	Direction	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.6 MC\_MoveRelative

If Supported	MC_MoveRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
B	Distance	Y	
B	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.7 MC\_MoveAdditive

If Supported	MC_MoveAdditive	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

**Appendix B 3.8 MC\_MoveSuperimposed**

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

**Appendix B 3.9 MC\_HaltSuperimposed**

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

**Appendix B 3.10 MC\_MoveVelocity**

If Supported	MC_MoveVelocity	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	Direction	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	InVelocity	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	

B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.11 MC\_MoveContinuousAbsolute

If Supported	MC_MoveContinuousAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Position		
B	EndVelocity		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	Direction		
E	BufferMode		
VAR_OUTPUT			
B	InEndVelocity		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix B 3.12 MC\_MoveContinuousRelative

If Supported	MC_MoveContinuousRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
B	Distance		
B	EndVelocity		
B	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	InEndVelocity		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix B 3.13 MC\_TorqueControl

If Supported	MC_TorqueControl	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
B	Torque	Y	
E	TorqueRamp	N	
V	TorquePositiveRamp	Y	Datatype is LREAL Specify the ramp from the current set torque until the positive target torque is reached. Setting range: 0 to 1000.00 [%/s]
V	TorqueNegativeRamp	Y	Datatype is LREAL Specify the ramp from the current set torque until the negative target torque is reached. Setting range: 0 to 1000.00 [%/s]
E	Velocity	Y	Called LimitVelocity
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	Direction	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	InTorque	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix B 3.14 MC\_PositionProfile

If Supported	MC_PositionProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimePosition		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	PositionScale		
E	Offset		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

**Appendix B 3.15 MC\_VelocityProfile**

If Supported	MC_VelocityProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimeVelocity		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	VelocityScale		
E	Offset		
E	BufferMode		
VAR_OUTPUT			
B	ProfileCompleted		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

**Appendix B 3.16 MC\_AccelerationProfile**

If Supported	MC_AccelerationProfile	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	TimeAcceleration		
VAR_INPUT			
B	Execute		
E	ContinuousUpdate		
E	TimeScale		
E	AccelerationScale		
E	Offset		
E	BufferMode		
VAR_OUTPUT			
B	ProfileCompleted		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

**Appendix B 3.17 MC\_SetPosition**

If Supported	MC_SetPosition	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Relative	Y	
E	ExecutionMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	

V	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	CancelAccepted	Y	Datatype is BOOL Indicates that the FB accepted cancel.

### Appendix B 3.18 MC\_SetOverride

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

### Appendix B 3.19 MC\_ReadParameter & MC\_ReadBoolParameter

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

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

Name	B/E	R/W	Supp. Y/N	Comments
CommandedPosition	B	R		
SWLimitPos	E	R/W		
SWLimitNeg	E	R/W		
EnableLimitPos	E	R/W		
EnableLimitNeg	E	R/W		
EnablePosLagMonitoring	E	R/W		
MaxPositionLag	E	R/W		
MaxVelocitySystem	E	R		
MaxVelocityAppl	B	R/W		
ActualVelocity	B	R		
CommandedVelocity	B	R		
MaxAccelerationSystem	E	R		
MaxAccelerationAppl	E	R/W		
MaxDecelerationSystem	E	R		
MaxDecelerationAppl	E	R/W		
MaxJerkSystem	E	R		
MarkJerkAppl	E	R/W		
SWLimitValid	V	R/W		

**Table 4: Parameters for MC\_Read(Bool)Parameter and MC\_Write(Bool)Parameter**



### Appendix B 3.20 MC\_WriteParameter & MC\_WriteBoolParameter

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

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

### Appendix B 3.21 MC\_ReadDigitalInput

If Supported	MC_ReadDigitalInput	Sup. Y/N	Comments
VAR_IN_OUT			
B	Input		
VAR_INPUT			
B	Enable		
E	InputNumber		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Value		

**Appendix B 3.22 MC\_ReadDigitalOutput**

If Supported	MC_ReadDigitalOutput	Sup.Y/N	Comments
VAR_IN_OUT			
B	Output		
VAR_INPUT			
B	Enable		
E	OutputNumber		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Value		

**Appendix B 3.23 MC\_WriteDigitalOutput**

If Supported	MC_WriteDigitalOutput	Sup.Y/N	Comments
VAR_IN_OUT			
B	Output		
VAR_INPUT			
B	Execute		
E	OutputNumber		
B	Value		
E	ExecutionMode		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

**Appendix B 3.24 MC\_ReadActualPosition**

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

**Appendix B 3.25 MC\_ReadActualVelocity**

If Supported	MC_ReadActualVelocity	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Velocity		

### Appendix B 3.26 MC\_ReadActualTorque

If Supported	MC_ReadActualTorque	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	Torque		

### Appendix B 3.27 MC\_ReadStatus

If Supported	MC_ReadStatus	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
B	ErrorStop		
B	Disabled		
B	Stopping		
E	Homing		
B	Standstill		
E	DiscreteMotion		
E	ContinuousMotion		
E	SynchronizedMotion		

### Appendix B 3.28 MC\_ReadMotionState

If Supported	MC_ReadMotionState	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
E	Source		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
E	ConstantVelocity		
E	Accelerating		
E	Decelerating		
E	DirectionPositive		
E	DirectionNegative		

### Appendix B 3.29 MC\_ReadAxisInfo

If Supported	MC_ReadAxisInfo	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
E	ErrorID		
E	HomeAbsSwitch		
E	LimitSwitchPos		
E	LimitSwitchNeg		
E	Simulation		
E	CommunicationReady		
E	ReadyForPowerOn		
E	PowerOn		
E	IsHomed		
E	AxisWarning		

### Appendix B 3.30 MC\_ReadAxisError

If Supported	MC_ReadAxisError	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Enable		
VAR_OUTPUT			
B	Valid		
E	Busy		
B	Error		
B	ErrorID		
E	AxisErrorID		

### Appendix B 3.31 MC\_Reset

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

### Appendix B 3.32 MC\_DigitalCamSwitch

If Supported	MC_DigitalCamSwitch	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
B	Switches		
E	Outputs		
E	TrackOptions		
VAR_INPUT			
B	Enable		
E	EnableMask		
E	ValueSource		
VAR_OUTPUT			
B	InOperation		
E	Busy		
B	Error		
E	ErrorID		

Basic elements within the array structure of MC\_CAMSWITCH\_REF

B/E	Parameter	Sup. Y/N	Comments
B	TrackNumber		
B	FirstOnPosition [u]		
B	LastOnPosition [u]		
E	AxisDirection		
E	CamSwitchMode		
E	Duration		

Basic elements within the array structure of MC\_TRACK\_REF

B/E	Parameter	Sup. Y/N	Comments
E	OnCompensation		
E	OffCompensation		
E	Hysteresis [u]		

### Appendix B 3.33 MC\_TouchProbe

If Supported	MC_TouchProbe	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
E	TriggerInput		
VAR_INPUT			
B	Execute		
E	WindowOnly		
E	FirstPosition		
E	LastPosition		
VAR_OUTPUT			
B	Done		
E	Busy		
E	CommandAborted		
B	Error		
E	ErrorID		
B	RecordedPosition		

### Appendix B 3.34 MC\_AbortTrigger

If Supported	MC_AbortTrigger	Sup.Y/N	Comments
VAR_IN_OUT			
B	Axis		
E	TriggerInput		
VAR_INPUT			
B	Execute		
VAR_OUTPUT			
B	Done		
E	Busy		
B	Error		
E	ErrorID		

### Appendix B 3.35 MC\_CamTableSelect

If Supported	MC_CamTableSelect	Sup. Y/N	Comments
VAR_IN_OUT			
E	Master	Y	
E	Slave	Y	
B	CamTable	Y	
VAR_INPUT			
B	Execute	Y	
E	Periodic	Y	
E	MasterAbsolute	Y	
E	SlaveAbsolute	Y	
E	ExecutionMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
E	CamTableID	Y	

### Appendix B 3.36 MC\_CamIn

If Supported	MC_CamIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
E	MasterOffset	Y	
E	SlaveOffset	Y	
E	MasterScaling	Y	
E	SlaveScaling	Y	
E	MasterStartDistance	Y	
E	MasterSyncPosition	Y	
E	StartMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	MasterValueSource	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	CamTableID	Y	
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	InSync	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
E	EndOfProfile	Y	
V	InputPerCycle	Y	Datatype is LREAL Stores the current value per cycle.
V	Reference	Y	Datatype is LREAL Stores the reference value.
V	Output	Y	Datatype is LREAL Stores the output value.
V	InstanceID	Y	Datatype is Word [Unsigned]/Bit String[32-bit] Instance ID that is automatically set when an instance is created. Can be used in other FBs as input.

### Appendix B 3.37 MC\_CamOut

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

**Appendix B 3.38 MC\_GearIn**

If Supported	MC_GearIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
B	RatioNumerator	Y	
B	RatioDenominator	Y	
E	MasterValueSource	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	InGear	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

**Appendix B 3.39 MC\_GearOut**

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



### Appendix B 3.40 MC\_GearInPos

If Supported	MC_GearInPos	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
B	RatioNumerator		
B	RatioDenominator		
E	MasterValueSource		
B	MasterSyncPosition		
B	SlaveSyncPosition		
E	SyncMode		
E	MasterStartDistance		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
E	StartSync		
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix B 3.41 MC\_PhasingAbsolute

If Supported	MC_PhasingAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
B	PhaseShift		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		
E	AbsolutePhaseShift		

## Appendix B 3.42 MC\_PhasingRelative

If Supported	MC_PhasingRelative	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
B	PhaseShift		
E	Velocity		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		
E	CoveredPhaseShift		

## Appendix B 3.43 CombineAxes

If Supported	MC_CombineAxes	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master1	Y	
B	Master2	Y	
B	Slave	Y	
VAR_INPUT			
B	Execute	Y	
E	ContinuousUpdate	Y	
E	CombineMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	GearRationNumeratorM1	Y	
E	GearRatioDenominatorM1	Y	
E	GearRatioNumeratorM2	Y	
E	GearRatioDenominatorM2	Y	
E	MasterValueSourceM1	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	MasterValueSourceM2	Y	Supported as pre-defined INT label Refer to Table 2 for their values
E	BufferMode	Y	Supported as pre-defined INT label Refer to Table 2 for their values
V	Options	Y	Datatype is Double Word [Unsigned]/Bit String[32-bit] Set additional function options
VAR_OUTPUT			
B	InSync	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

#### **Appendix B 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.