

Subset of the

**Technical Specification**

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

**Function blocks for motion control**

**Version 1.1**

**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 OF 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	Siemens AG
Supplier address	Frauenauracher Straße 80
City	91056 Erlangen
Country	Germany
Telephone	09131/98-0
Fax	09131/98-2568
Email address	<a href="mailto:Manfred.Popp@siemens.com">Manfred.Popp@siemens.com</a>
Product Name	SIMOTION
Product version	V4.0
Release date	04.2006

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): Erwin Neis

### Appendix A 2. Supported Datatypes

Defined datatypes with MC library:	Supported	If not supported, which datatype used
BOOL	YES	
INT	YES	
WORD	NO	DWORD
REAL	NO	LREAL
ENUM	YES	

**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	Nearly all FBs	YES	_AXIS_REF Reference to a vendor specific data format
MC_Direction (extended)	MC_MoveAbsolute MC_MoveVelocity	YES	_MC_Direction: USER_DEFAULT, POSITIVE, SHORTEST_WAY, NEGATIVE, EFFECTIVE
MC_TP_REF	MC_PositionProfile	YES	_MC_TP_REF Reference to a vendor specific data format
MC_TV_REF	MC_VelocityProfile	YES	_MC_TV_REF Reference to a vendor specific data format
MC_TA_REF	MC_AccelerationProfile	NO	
MC_CAM_REF	MC_CamTableSelect MC_CamIn	YES	_MC_CAM_REF Reference to a vendor specific data format
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn	NO	
MC_StartMode (extended)	MC_CamIn	NO	
MC_BufferMode	Buffered FBs	NO	
MC_HomingMode (extended)	MC_Home	YES	_MC_HomingMode: ACTIVE_HOMING, PASSIVE_HOMING, DIRECT_HOMING, ENABLE_OFFSET_OF_ABSOLUTE_ENCODER
_MC_Mode (vendor specific)	MC_Power	YES	ALL, DRIVE
_MC_StopMode (vendor specific)	MC_Power	YES	WITH_COMMAND_VALUE_ZERO, WITH_MAXIMAL_DECELERATION, IN_DEFINED_TIME, DISABLE_DRIVE_IMMEDIATELY
_MC_CamInMode (vendor specific)	MC_CamIn	YES	USER_DEFAULT, IMMEDIATELY_BY_TIME_PROFILE
_MC_GearInMode (vendor specific)	MC_GearIn	YES	USER_DEFAULT, IMMEDIATELY_BY_TIME_PROFILE

**Table 2: Supported derived datatypes**

All SIMOTION system datatypes begin with a leading underscore. Therefore also the PLCopen datatypes was realized with a leading underscore.

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

Since SIMOTION V4.0 the PLCopen Function Blocks are integrated directly as system function blocks in the SIMOTION SCOUT command library. All SIMOTION system functions begin with a leading underscore. Therefore also the PLCopen Function Blocks was realized with a leading underscore.

Single Axis Function Blocks	Supported Yes / No	Comments (<= 48 char.)
MC_MoveAbsolute	YES	_MC_MoveAbsolute
MC_MoveRelative	YES	_MC_MoveRelative
MC_MoveAdditive	YES	_MC_MoveAdditive
MC_MoveSuperimposed	YES	_MC_MoveSuperimposed
MC_MoveVelocity	YES	_MC_MoveVelocity
MC_Home	YES	_MC_Home
MC_Stop	YES	_MC_Stop
MC_Power	YES	_MC_Power
MC_ReadStatus	YES	_MC_ReadStatus
MC_ReadAxisError	YES	_MC_ReadAxisError
MC_Reset	YES	_MC_Reset
MC_ReadParameter	YES	_MC_ReadParameter
MC_ReadBoolParameter	YES	_MC_ReadBoolParameter
MC_WriteParameter	YES	_MC_WriteParameter
MC_WriteBoolParameter	YES	_MC_WriteBoolParameter
MC_ReadActualPosition	YES	_MC_ReadActualPosition
MC_PositionProfile	YES	_MC_PositionProfile
MC_VelocityProfile	YES	_MC_VelocityProfile
MC_AccelerationProfile	NO	
Multi-Axis Function Blocks	Supported Yes / No	Comments (<= 48 char.)
MC_CamTableSelect	NO	implicit in _MC_CamIn
MC_CamIn	YES	_MC_CamIn
MC_CamOut	YES	_MC_CamOut
MC_GearIn	YES	_MC_GearIn
MC_GearOut	YES	_MC_GearOut
MC_Phasing	YES	_MC_Phasing

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

In older versions as SIMOTION V4.0 the PLCopen Function Blocks (only for Single-Axis) was included in the SIMOTION Function Library.

### Appendix A 6.1 MoveAbsolute

If Supported	MC_MoveAbsolute	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	Direction	Y	_MC_Direction
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 6.2 MoveRelative

If Supported	MC_MoveRelative	Supported Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	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 6.3 MoveAdditive

If Supported	MC_MoveAdditive	Supported Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	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 6.4 MoveSuperimposed

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



### Appendix A 6.5 MoveVelocity

If Supported	MC_MoveVelocity	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	Direction	Y	_MC_Direction
E	BufferMode	N	
V	Current	Y	BOOL; programming mode for velocity
VAR_OUTPUT			
B	InVelocity	Y	
E	Busy	Y	
E	Active	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	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	HomingMode	Y	_MC_HomingMode
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 6.7 Stop

If Supported	MC_Stop	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
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 6.8 Power

If Supported	MC_Power	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
E	Enable_Positive	N	Via input "Mode" and "StopMode"
E	Enable_Negative	N	Via input "Mode" and "StopMode"
E	BufferMode	N	
V	Mode	Y	_MC_Mode
V	StopMode	Y	_MC_StopMode
VAR_OUTPUT			
B	Status	Y	
E	Busy	Y	
E	Active	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	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
E	ErrorID	Y	
B	Disabled	Y	
B	Errorstop	Y	
B	Stopping	Y	
B	StandStill	Y	
B	DiscreteMotion	Y	
B	ContinuousMotion	Y	
E	SynchronizedMotion	Y	
E	Homing	Y	
E	ConstantVelocity	Y	
E	Accelerating	Y	
E	Decelerating	Y	

### Appendix A 6.10 ReadAxisError

If Supported	MC_ReadAxisError	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	Y	
B	Error	Y	
B	ErrorID	Y	
B	AxisErrorId	Y	

### Appendix A 6.11 Reset

If Supported	MC_Reset	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
V	Restart		BOOL; initialization mode
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
V	CommandAborted	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	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
B	ParameterNumber	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	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	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
B	ParameterNumber	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	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	Y	
SWLimitNeg	E	R/W	Y	
SWLimitState	V	R/W	Y	BOOL; enable both switches
EnableLimitPos	E	R/W	N	see parameter 'SWLimitState'
EnableLimitNeg	E	R/W	N	see parameter 'SWLimitState'
EnablePosLagMonitoring	E	R/W	Y	BOOL
MaxPositionLag	E	R/W	Y	
MaxVelocitySystem	E	R	N	
MaxVelocityAppl	B	R/W	Y	
ActualVelocity	B	R	Y	
CommandedVelocity	B	R	Y	
MaxAccelerationSystem	E	R	N	
MaxAccelerationAppl	E	R/W	Y	
MaxDecelerationSystem	E	R	N	
MaxDecelerationAppl	E	R/W	N	
MaxJerk	E	R/W	Y	

**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	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	ParameterNumber	Y	
B	Value	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Buffered	N	
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	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	ParameterNumber	Y	
B	Value	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Buffered	N	
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	_AXIS_REF declared as input
VAR_INPUT			
B	Enable	Y	
VAR_OUTPUT			
B	Valid	Y	
E	Busy	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	_AXIS_REF declared as input
B	TimePosition	Y	_MC_TP_REF declared as input
VAR_INPUT			
B	Execute	Y	
E	TimeScale	N	available as system function
E	PositionScale	N	available as system function
E	Offset	N	available as system function
E	BufferMode	N	
V	Absolute	Y	BOOL; position mode
V	Velocity	Y	LREAL; maximum velocity
V	Acceleration	Y	LREAL; Maximum acceleration
V	Deceleration	Y	LREAL; maximum deceleration
V	Jerk	Y	LREAL; maximum jerk
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	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	_AXIS_REF declared as input
B	MC_TimeVelocity	Y	_MC_TV_REF declared as input
VAR_INPUT			
B	Execute	Y	
E	TimeScale	N	available as system function
E	VelocityScale	N	available as system function
E	Offset	N	available as system function
E	BufferMode	N	
V	Acceleration	Y	LREAL; maximum acceleration
V	Deceleration	Y	LREAL; maximum acceleration
V	Jerk	Y	LREAL; maximum jerk
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	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		
B	MC_TimeAcceleration		
VAR_INPUT			
B	Execute		
B	ArraySize		
E	Scale		
E	Offset		
E	BufferMode		
VAR_OUTPUT			
B	Done		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

### Appendix A 6.20 CamTableSelect

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

### Appendix A 6.21 CamIn

If Supported	MC_CamIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	_AXIS_REF declared as input
B	Slave	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
E	MasterOffset	Y	
E	SlaveOffset	Y	
E	MasterScaling	Y	
E	SlaveScaling	Y	
E	StartMode	N	see vendor specific parameters
E	CamTableID	N	see vendor specific parameter CamTable
E	BufferMode	N	
V	CamTable	Y	Reference to CAM description (_MC_CAM_REF).

V	MasterAbsolute	Y	BOOL; master mode
V	SlaveAbsolute	Y	BOOL; slave mode
V	CyclicMode	Y	BOOL; periodic property
V	Velocity	Y	LREAL; maximum slave velocity
V	Acceleration	Y	LREAL; maximum slave acceleration
V	Jerk	Y	LREAL; maximum slave jerk
V	Mode	Y	_MC_CamInMode
VAR_OUTPUT			
B	InSync	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
E	EndOfProfile	N	

### Appendix A 6.22 CamOut

If Supported	MC_CamOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
V	Active	Y	
V	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.23 GearIn

If Supported	MC_GearIn	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	_AXIS_REF declared as input
B	Slave	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	RatioNumerator	Y	DINT
B	RatioDenominator	Y	DINT
V	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	N	
V	PhaseShift	Y	LREAL; shift between master and slave position
V	Absolute	Y	BOOL; relationship between master and slave
V	Mode	Y	_MC_GearInMode
VAR_OUTPUT			
B	InGear	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	



### Appendix A 6.24 GearOut

If Supported	MC_GearOut	Sup. Y/N	Comments
VAR_IN_OUT			
B	Slave	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
V	Active	Y	
V	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### Appendix A 6.25 Phasing

If Supported	MC_Phasing	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master	Y	_AXIS_REF declared as input
B	Slave	Y	_AXIS_REF declared as input
VAR_INPUT			
B	Execute	Y	
B	PhaseShift	Y	
E	Velocity	Y	
E	Acceleration	Y	
E	Deceleration	Y	
E	Jerk	Y	
E	BufferMode	N	
V	Absolute	Y	BOOL; shift relationship
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	Y	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	

### 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.