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

July 2005

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 data types (see Appendix A 2 Supported Data types) 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 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 A 2 Supported Data types and Appendix A 3 Overview of the Function 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 A 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.

- **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 A 1. Statement of Supplier

Supplier name	Siemens AG
Supplier address	Gleiwitzer Str. 555
City	90475 Nuremberg
Country	Germany
Telephone	+49 (911) 895-3503
Fax	+49 (911) 895-133503
Email address	markus.kempf@siemens.com
Product Name	SIMATIC S7-1200
Product version	V01.00.00
Release date	10.06.2009

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.

:
Markus Kempf

16.06.2009

Signature:

Appendix A 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	INT

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	TO_Axis_PTO
MC_Direction (extended)	MC_MoveAbsolute MC_MoveVelocity	Yes	INT
MC_TP_REF	MC_PositionProfile		
MC_TV_REF	MC_VelocityProfile		
MC_TA_REF	MC_AccelerationProfile		
MC_CAM_REF	MC_CamTableSelect		
MC_CAM_ID (extended)	MC_CamTableSelect MC_CamIn		
MC_StartMode (extended)	MC_CamIn		
MC_BufferMode	Buffered FBs		

Table 2: Supported derived datatypes

Appendix A 3. Overview of the Function Blocks

Single Axis Function Blocks	**Supported Yes / No**	**Comments (<= 48 char.)**
MC_MoveAbsolute	Yes	
MC_MoveRelative	Yes	
MC_MoveAdditive		
MC_MoveSuperimposed		
MC_MoveVelocity	Yes	
MC_Home	Yes	
MC_Stop		
MC_Power	Yes	
MC_ReadStatus		Is supported by system function
MC_ReadAxisError		Is supported by system function
MC_Reset	Yes	
MC_ReadParameter		Is supported by system function
MC_ReadBoolParameter		Is supported by system function
MC_WriteParameter		Is supported by system function
MC_WriteBoolParameter		Is supported by system function
MC_ReadActualPosition		Is supported by system function
MC_PositionProfile		
MC_VelocityProfile		
MC_AccelerationProfile		
MC_Halt	Yes	Function listed in “Part 2 – Extensions”
Multi-Axis Function Blocks	**Supported Yes / No**	**Comments (<= 48 char.)**
MC_CamTableSelect		
MC_CamIn		
MC_CamOut		
MC_GearIn		
MC_GearOut		
MC_Phasing		

Table 3: Short overview of the Function Blocks

Appendix A 3.1 MoveAbsolute

If Supported	**MC_MoveAbsolute**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	Velocity	Y	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
E	Direction	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.2 MoveRelative

If Supported	**MC_MoveRelative**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
B	Distance	Y	
E	Velocity	Y	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.3 MoveAdditive

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

Appendix A 3.4 MoveSuperimposed

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

Appendix A 3.5 MoveVelocity

If Supported	**MC_MoveVelocity**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
E	Velocity	Y	
E	Acceleration	N	
E	Deceleration	N	
E	Jerk	N	
E	Direction	Y	
E	BufferMode	N	
V	Current	Y	Run in “Continuous Motion” with the current velocity and direction
VAR_OUTPUT			
B	InVelocity	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.6 Home

If Supported	**MC_Home**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
B	Position	Y	
E	HomingMode	Y	As input “Mode”
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.7 Stop

If Supported	**MC_Stop**	Sup. Y/N	Comments
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 A 3.8 Power

If Supported	**MC_Power**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Enable	Y	
E	Enable_Positive	N	
E	Enable_Negative	N	
E	BufferMode	N	
V	StopMode	Y	Axis behavior after disabling
VAR_OUTPUT			
B	Status	Y	
E	Busy	Y	
E	Active	N	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.9 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	Disabled		
B	Errorstop		
B	Stopping		
B	StandStill		
B	DiscreteMotion		
B	ContinuousMotion		
E	SynchronizedMotion		
E	Homing		

E	ConstantVelocity		
E	Accelerating		
E	Decelerating		

Appendix A 3.10 ReadAxisError

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

Appendix A 3.11 Reset

If Supported	**MC_Reset**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
B	Error	Y	
B	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

Appendix A 3.12 ReadParameter

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		

Appendix A 3.13 ReadBoolParameter

If Supported	**MC_ReadBoolParameter**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis		
VAR_INPUT			
B	Valid		
B	ParameterNumber		
VAR_OUTPUT			
B	Done		
E	Busy		

B	Error		
E	ErrorID		
B	Value		

Name	**B/E**	**R/W**	**Sup. 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		
MaxJerk	E	R/W		

Table 4: Parameters for ReadParameter and WriteParameter

Appendix A 3.14 WriteParameter

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

Appendix A 3.15 WriteBoolParameter

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

Appendix A 3.16 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 A 3.17 PositionProfile

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

Appendix A 3.18 VelocityProfile

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

Appendix A 3.19 AccelerationProfile

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

Appendix A 3.20 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 3.21 CamIn

If Supported	**MC_CamIn**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
E	MasterOffset		
E	SlaveOffset		
E	MasterScaling		
E	SlaveScaling		
E	StartMode		
E	CamTableID		
E	BufferMode		
VAR_OUTPUT			
B	InSync		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		
E	EndOfProfile		

Appendix A 3.22 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 A 3.23 GearIn

If Supported	**MC_GearIn**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Master		
B	Slave		
VAR_INPUT			
B	Execute		
B	RatioNumerator		
B	RatioDenominator		
E	Acceleration		
E	Deceleration		
E	Jerk		
E	BufferMode		
VAR_OUTPUT			
B	InGear		
E	Busy		
E	Active		
E	CommandAborted		
B	Error		
E	ErrorID		

Appendix A 3.24 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 A 3.25 Phasing

If Supported	**MC_Phasing**	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		

Appendix of Part 2 - Extensions MC_Halt

If Supported	**MC_Halt**	Sup. Y/N	Comments
VAR_IN_OUT			
B	Axis	Y	As input
VAR_INPUT			
B	Execute	Y	
E	Deceleration	N	
E	Jerk	N	
E	BufferMode	N	
VAR_OUTPUT			
B	Done	Y	
E	Busy	Y	
E	Active	N	
E	CommandAborted	Y	
B	Error	Y	
E	ErrorID	Y	
V	ErrorInfo	Y	Detailed error information

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 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.