

SIMOTION

Frequently asked Questions

Detailed Troubleshooting of the PLCopen Block
Behavior

SIEMENS

PLCopen Blocks Behavior

Subject to change without prior notice.

Copyright

The distribution or reproduction in whole or in part in any form or medium without express written permission of Siemens AG is prohibited. Violators shall be held liable for damages. All rights reserved. Especially in the case of a patent grant or registration of a utility model or design.

General information

Remark

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are correctly used. These Application Examples do not relieve you of the responsibility in safely and professionally using, installing, operating and servicing equipment. When using these Application Examples, you recognize that Siemens cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice. If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications - e.g. Catalogs - then the contents of the other documents have priority.

Warranty, liability and support

We do not accept any liability for the information contained in this document.

Any claims against us - based on whatever legal reason - resulting from the use of the examples, information, programs, engineering and performance data etc., described in these Application Examples shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act ("Produkthaftungsgesetz"), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract ("wesentliche Vertragspflichten"). However, claims arising from a breach of a condition which goes to the root of the contract shall be limited to the foreseeable damage which is intrinsic to the contract, unless caused by intent or gross negligence or based on mandatory liability for injury of life, body or health. The above provisions do not imply a change in the burden of proof to your detriment.

Copyright© 2009 Siemens I DT. The reproduction or transmission of these standard applications or excerpts of them is not permitted without the express written permission of Siemens I DT.

For questions regarding this application please contact us at the following e-mail address:

<mailto:applications.erlf@siemens.com>

PLCopen Blocks Behavior

Qualified personnel

In the sense of this documentation, qualified personnel are those who are knowledgeable and qualified to mount/install, commission, operate and service/maintain the products which are to be used. He or she must have the appropriate qualifications to carry out these activities

e.g.

- Trained and authorized to energize and de-energize, ground and tag circuits and equipment according to applicable safety standards.
- Trained or instructed according to the latest safety standards regarding the maintenance and use of the appropriate safety equipment.
- Trained in rendering first aid.

There is no explicit warning information in this documentation. However, reference is made to warning information in the Operating Instructions for the particular product.

Reference regarding export codes

AL: N

ECCN: N

PLCopen Blocks Behavior

Table of contents

1	Question	6
2	Solution	6
2.1	Detailed PLCopen block output parameter behavior as of V4.1 SP4	6
2.2	Detailed PLCopen block error evaluation as of V4.1 SP4	10
2.2.1	PLCopen block output parameters	10
2.2.2	Troubleshooting of PLCopen blocks (see details in Chapt. 4.1)	11
2.3	General rules regarding the PLCopen FB interface.....	12
2.4	Restoring the behavior shown before V4.1 SP4.....	14
Annex.....		15
3	Revisions.....	15
4	Contact partner	16

PLCopen Blocks Behavior

1 Question

What about the behavior of the output parameters of the PLCopen blocks?

Modified PLCopen block output parameter behavior regarding SIMOTION V4.1 SP4

During error correction, the output parameter behavior was modified.

How to proceed when evaluating PLCopen block errors ?

Modified PLCopen block error evaluation behavior regarding SIMOTION V4.1 SP4

During error correction, the error output behavior of PLCopen blocks was also modified (this concerns outputs Error, ErrorID and CommandAborted). So far, the behavior shown was not unambiguous nor documented in detail.

Can the previous behavior be restored?

The behavior shown before V4.1 SP4 can be restored (see Chapt. 2.4, Restoring the behavior shown before V4.1 SP4).

2 Solution

More details on the documentation PLCopen_Funktionen.pdf (PLCopen_Functions.pdf) (5/2009 edition) are provided in the following paragraph.

2.1 Detailed PLCopen block output parameter behavior as of V4.1 SP4

The output parameter behavior has been adapted to SIMOTION V4.1 SP4 according to the PLCopen specification V1.1.

Above all, changes have been made to the output parameters "InGear", "InSync" and "InVelocity" of the blocks MC_GearIn, MC_CamIn and MC_MoveVelocity.

These outputs are reset with the falling edge of "Execute". The falling edge of "Execute", however, does not stop or influence the execution of the real FB. When this situation occurs, the corresponding output parameters are set for at least one cycle, also if "Execute" has been reset before completion of the execution of the FB.

Also see **Status of output parameters** in Chapt. 2.3, General rules regarding the PLCopen FB interface.

The behavior shown before V4.1 SP4 can be restored (see Chapt. 2.4, Restoring the behavior shown before V4.1 SP4).

PLCopen Blocks Behavior

In the following, please find the modified figures for the documentation SIMOTION PLCopen Blocks according to the new behavior:

Example: Gear synchronism

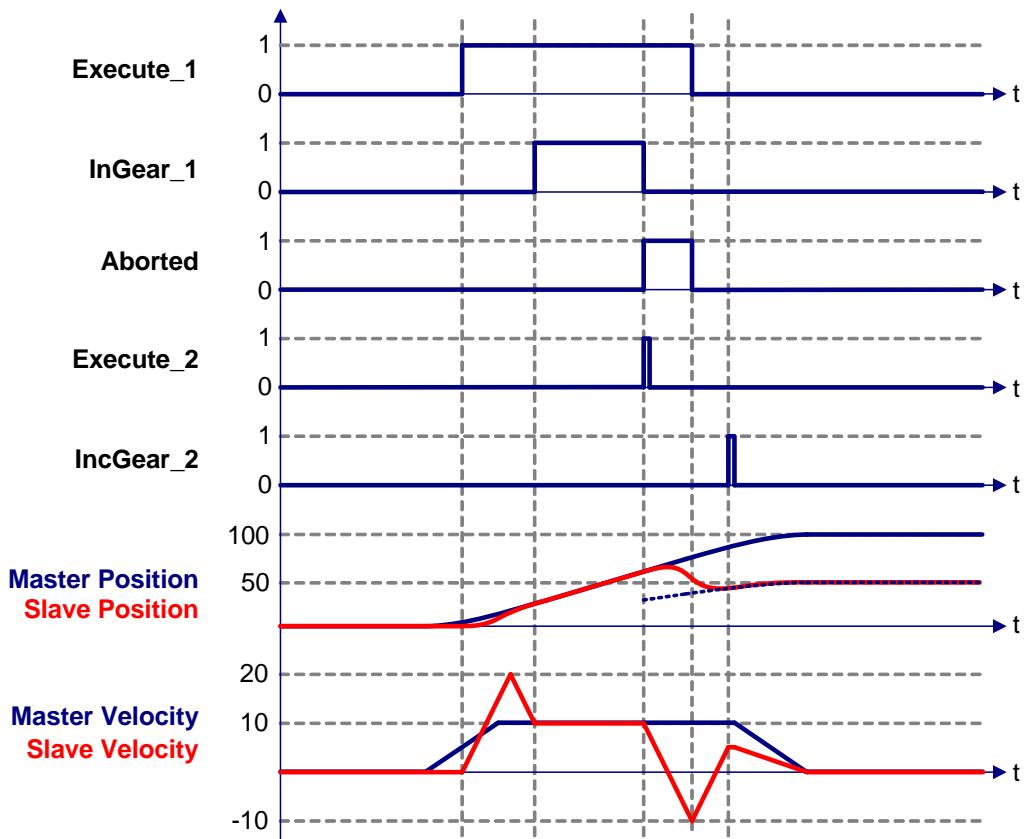
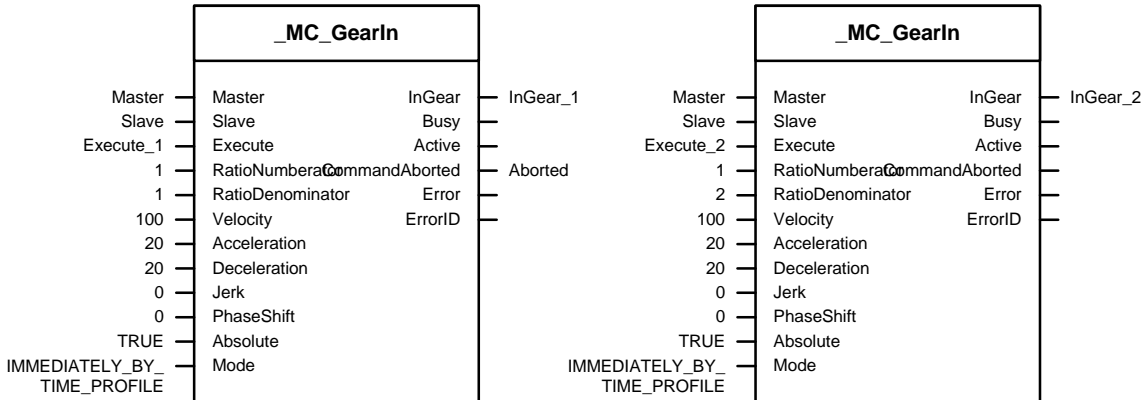


Fig. 3-40 _MC_GearIn Example: Gear synchronism

PLCopen Blocks Behavior

Example: Cam disk synchronism

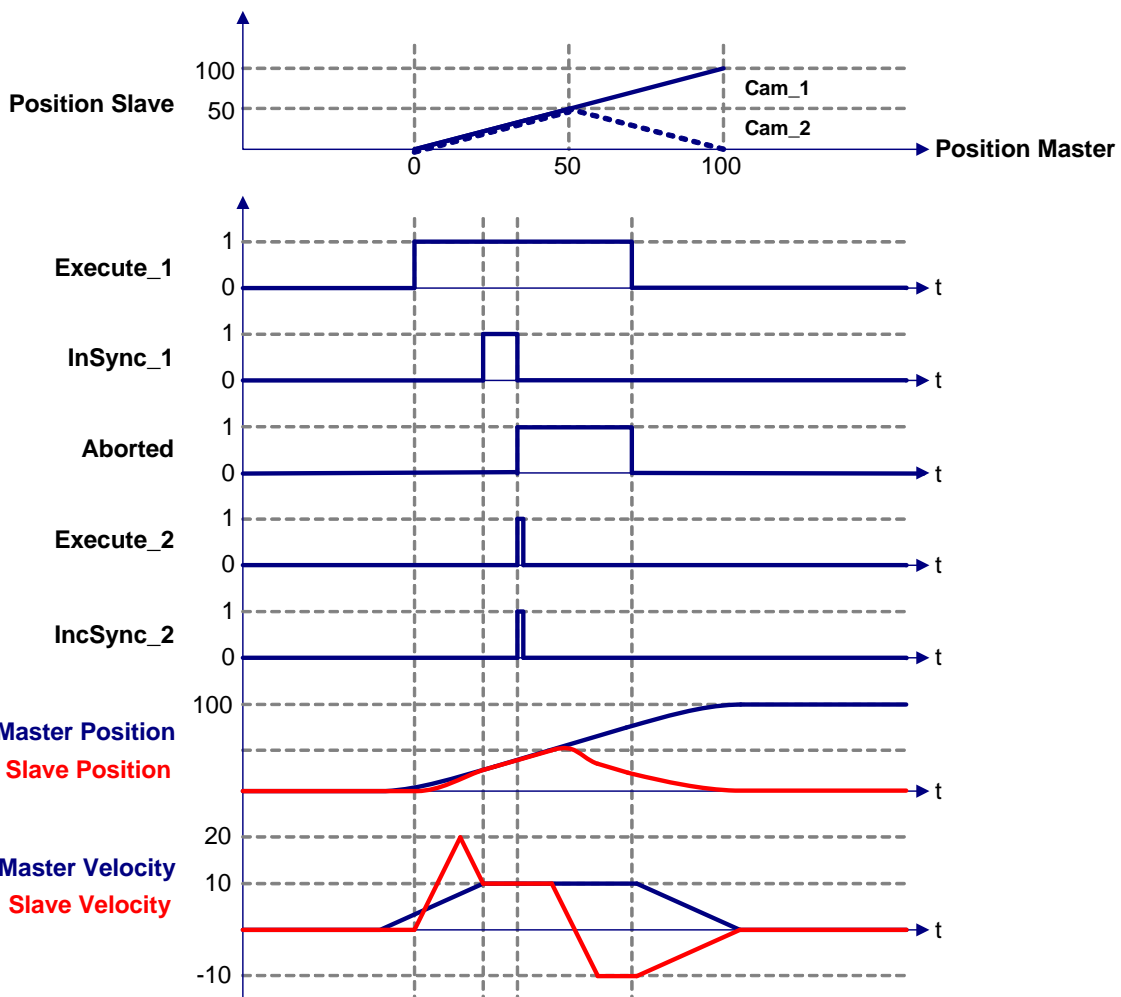
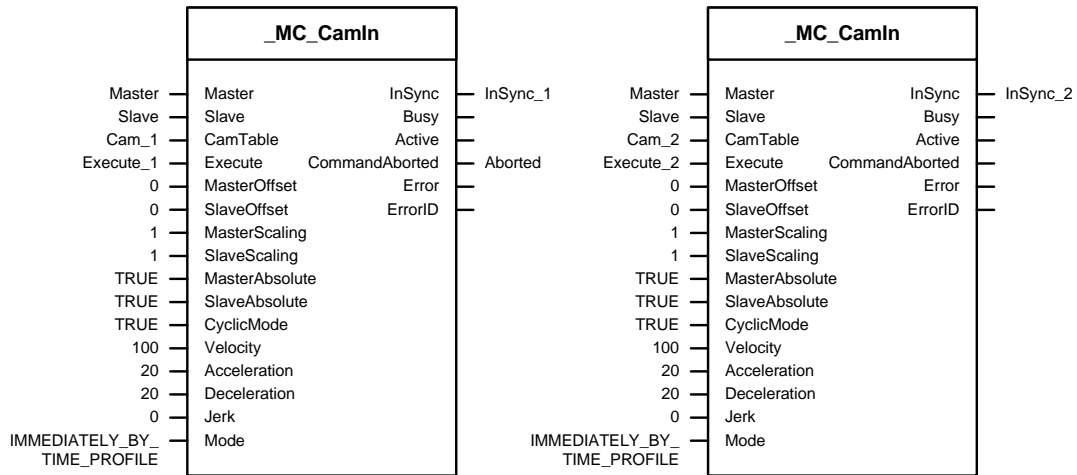


Fig. 3-45 `_MC_CamIn` Example: Cam disk synchronism

PLCopen Blocks Behavior

Example: Cam disk synchronism with offset

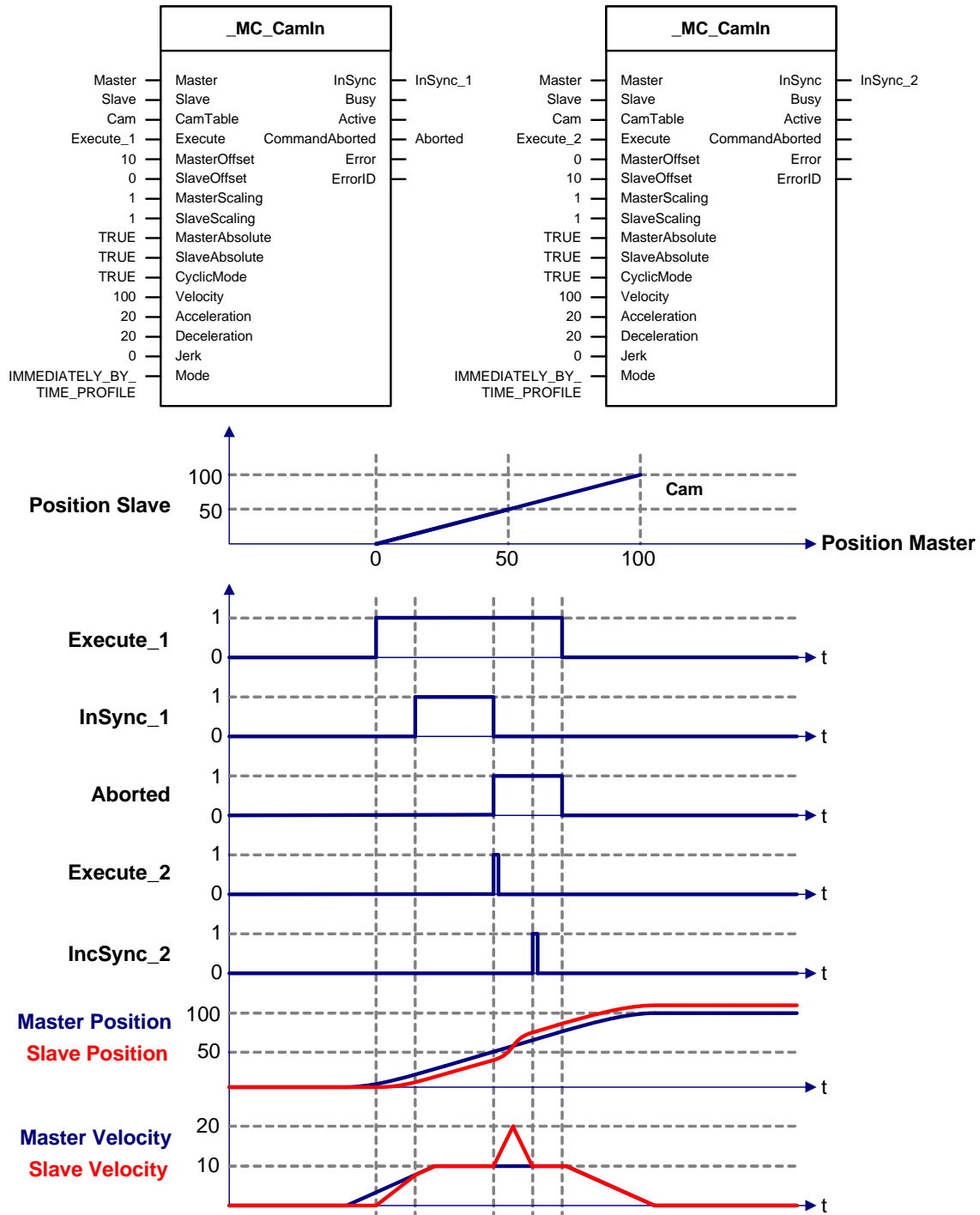


Fig. 3-46 `_MC_CamIn` Example: Cam disk synchronism with offset

PLCopen Blocks Behavior

Behavior with PLCopen block **_MC_STOP**

The output bit "Active" remains TRUE while the block is processed (movement command inhibit active). The block is terminated only when "Execute" = FALSE. The outputs "Done" and "Active" are then set for one cycle.

The modified behavior corresponds to the documentation.

The behavior shown before V4.1 SP4 can be restored (see Chapt. 2.4, Restoring the behavior shown before V4.1 SP4).

2.2 Detailed PLCopen block error evaluation as of V4.1 SP4

2.2.1 PLCopen block output parameters

Output parameters

Parameter	Data type	Initial value	Description
Done	BOOL	FALSE	Displays end of function block With TRUE, the programmed target position was reached.
Busy	BOOL	FALSE	Displays the function block activity With TRUE, the function block was started.
Active	BOOL	FALSE	Displays the function block command activity With TRUE, the command is processed by the command processing system, that means the function block actively controls the axis / technological object. The command is processed in the interpolator.
CommandAborted	BOOL	FALSE	Displays the function block abort With TRUE, the function block was aborted by another replacing function block or TO command.
Error	BOOL	FALSE	Displays a function block error With TRUE, a function block parameterization resp. processing error occurred. The function block is ended. The error description can be read at the output ErrorID.
ErrorID	DWORD	0	Displays a function block error code The error code is always output in combination with the output error (see ErrorIDs).

ErrorIDs

The error code comprises the number and, if available, the relevant reason of the error occurring in the function block. The error number occupies the lower 16 bits of the error code (see ErrorID error codes (LOW word) (Page 152)). The error reason, if available, is also coded by a number and occupies the upper 16 bits of the error code (see ErrorID command abort reason (HIGH word) (Page 153)).

PLCopen Blocks Behavior

Note regarding "Busy" and "Active" status signals with PLCopen-FB_MC_JOG

In continuous JOG mode, the behavior of status signals "Busy" and "Active" in version V4.1 SP4 differs from the one shown in previous versions. In version V4.1 SP4, the status signals go to the FALSE status when the axis has reached the setpoint velocity (so far: until the function was active).

This functional limitation is scheduled to be corrected in V4.1 SP5.

Please also observe the list of supplementary conditions.

2.2.2 Troubleshooting of PLCopen blocks (see details in Chapt. 4.1)

4.1 Troubleshooting - PLCopen blocks

The function blocks display errors occurring in the output parameters **Error** and **ErrorID**, with **CommandAborted** the execution was terminated prematurely.

Output parameters **Error** resp. **ErrorID**

When output parameter **Error** = TRUE, the technology function could not be started by the function block or was aborted when being processed. The error reason is stated in the output parameter value **ErrorID**.

The function block error numbers (ErrorID) have been taken from the ST command values returned in SIMOTION and from the command abort reason. The value returned is entered in the ErrorID LOW word. The command abort reason is stated in the ErrorID HIGH word.

If an error due to incorrect parameterization is displayed at the function block, the function block must be called using the correct parameters or later (when the function is permissible).

It is neither required nor possible to acknowledge errors resp. warnings. An error is displayed until the input parameters **Enable** resp. **Execute** have been reset.

If the function block displays an error with function abort due to a technological object error, the TO error must be acknowledged before calling up the function block again.

An error is displayed until the input parameter **Execute** has been reset.

Warnings and notes at the technological object which do not lead to function abort are not displayed at the output **Error** and **ErrorID**.

Technological object errors can be determined via the TO system variables **error** and **errorGroup**. In addition, you can request the error status of the technological objects via the function block **_MC_ReadAxisError** (see Chapter "Requesting general errors using the function block **_MC_ReadAxisError**").

Example for error handling with following error:

The axis movement has been started via the function block **_MC_MoveAbsolute**.

➔ Busy and Active output parameters are TRUE

A following error is detected in the technological object.

➔ TO error 50102 is output (default error reaction **RELEASE_DISABLE**)

➔ Busy and Active output parameters are FALSE

➔ The Error output parameter is TRUE

PLCopen Blocks Behavior

- ➔ Output parameter ErrorID = 16#00050003 shows "command aborted" together with the cause "abort due to pending error reaction"
- ➔ The Error system variable is TRUE and bit 8 set in ErrorGroup (FollowingError)

CommandAborted output parameter

If the function block was aborted by another replacing command (function block or TO command), this is indicated by the output parameter **CommandAborted**.

Fault acknowledgement on the technological object

1. Acknowledge all errors.

For this, eliminate the error causes and acknowledge the error with `_MC_Reset` (Restart = FALSE).

2. You can then release the technological object again with the function `_MC_Power`.

The behavior shown before V4.1 SP4 can be restored (see Chapt. 2.4, Restoring the behavior shown before V4.1 SP4).

2.3 General rules regarding the PLCopen FB interface

According to the PLCopen specification "Technical Specification, PLCopen - Technical Committee 2 – Task Force, Function blocks for motion control, Version 1.1", the following applies:

Exclusiveness of output parameters

The output parameters "Busy", "Done", "Error" and "CommandAborted" mutually exclude one another: On each FB, only one of these output parameters can be set to TRUE. If "Execute" is TRUE, one of these output parameters must be TRUE as well.

In each case, only one of the output parameters "Active", "Error", "Done" and "CommandAborted" is set.

Status of output parameters

The output parameters "Done", "InGear", "InSync", "InVelocity", "Error", "ErrorID" and "CommandAborted" are reset with the falling edge of "Execute". However, the falling edge of "Execute" does not stop resp. influence the execution of the actual FB. The corresponding output parameters are set for at least one cycle if this situation occurs, even if "Execute" has been reset before terminating the FB processing.

If an FB instance receives a new "Execute" parameter before the processing has been terminated (as a series of commands for the same instance), the FB does not return any feedback such as "Done" or "CommandAborted" for the previous action.

Input parameters

The parameters are used with the rising edge of the input parameter "Execute". If you wish to modify a parameter, change the input parameter(s) and initiate the processing/traversing movement again.

Missing input parameters

According to IEC 61131-3 - provided that an arbitrary parameter of a function block input is missing ("open") - the value from the previous instance call is used. The initialization value is used for the first call.

PLCopen Blocks Behavior

“Position“ versus “Distance“

“Position“ is a value which is defined within a coordinate system. “Distance“ is a relative value referring to technical units. “Distance“ designates the difference between two positions.

Rules of sign

The parameters “Velocity“, “Acceleration“, “Deceleration“ and “Jerk“ always have positive values. “Position“ and “Distance“ can be both positive and negative.

Error handling behavior

All blocks have two output parameters processing errors which may occur when executing the corresponding function block. These output parameters are defined as follows:

“Error“

The rising edge of “Error“ informs you that an error occurred when executing the function block.

“ErrorID“

Error number

“Done“, “InVelocity“, “InGear“ and “InSync“ designate a successful termination. For this reason, these signals and “Error“ logically exclude each other.

Error types:

- Function blocks (e.g. parameter outside the range, State Machine fault)
- Communication
- Drive

Instance errors do not always yield an axis error (setting the axis to “StandStill“).

The error outputs of the relevant FB are reset with the falling edge of “Execute“.

Behavior of output parameter “Done“

The output parameter “Done“ (and the output parameters “InGear“, “InSync“, etc.) is set if the requested action was successfully terminated.

In case of several function blocks working in a sequence on the same axis, the following applies:

If a movement along an axis was aborted by another movement on the same axis - without reaching the terminating target – the output parameter “Done“ of the first FB is not set.

Behavior of output parameter “CommandAborted“

“CommandAborted“ is activated if the requested traversing movement is aborted by another traversing command.

The reset behavior of “CommandAborted“ corresponds to that of “Done“. If “CommandAborted“ occurs, other output signals such as “InVelocity“ are reset.

Input parameters exceeding application limits

1. The parameter violates the parameter-related absolute upper/lower limits (at the related limit date):

If an FB is controlled with parameters leading to a violation of application limits, the FB instance generates an error. The consequences of this error for the axis depend on the relevant application and should therefore be processed by the application program.

2. The parameter is within these upper/lower limits, but not within the “application limit“, that means, it exceeds the TO limit values:

Depending on the input parameter and the general “SIMOTION“ rules.

In general, first of all the parameter is delimited; these limitations lead to an alarm/warning linked to a stop reaction -> used to set the FB to the error state (if required) with ErrorID: Abort via the pending error reaction.

PLCopen Blocks Behavior

2.4 Restoring the behavior shown before V4.1 SP4

There are two alternative options to restore the behavior shown in V4.1 SP4. It suffices to perform this for only one axis in the project.

1. Via a program in the StartUpTask

// Deactivation of the standard-conforming changes of the PLCopen blocks (modified to V4.1.4)
 // -> Behavior as before V4.1.4

```
<axis>.internalToTrace[1].id := 10010;
< axis >.internalToTrace[1].value := 1.0;
```

2. Parameterization in SCOUT

In the expert list -> system variables (offline) set the parameters

internalToTrace[1].id -> 10010

internalToTrace[1].value -> 1.0

(Press ENTER after input of numbers)

Store and download.

Parameter	Parametertext	Offlinewert	Einheit	Datentyp	Minimum	Maximum
forcecontrollermonitorings	Überwachungen der Kraft-/Druckregelung					
forcecontrollersettings	Überlagerungen Kraft-/Druckregler					
forceintgcommand	Bearbeitungsstatus Kraftbegrenzung					
forcepositionpositionprofile	Befehlszustand Kraft-Positions-Profil auf vers					
forcepositionprofilecomma	Bearbeitungsstatus Kraft-/Druck-Position-Profi					
forcestatedata	Dynamikstatus der Kraftregelung an der Achs					
forcetimeprofilecommand	Bearbeitungsstatus Kraft-/Druck-Zeit-Profi					
homingcommand	Bearbeitungsstatus „homing-Bewegung“ an d					
internalservosettings	Schwert- und Stellgrößenbeeinflussung fue					
internaltoTrace	Interne Trace-Variablen					
internaltoTrace[1]	Interne Trace-Variablen					
id	Id	10010	-	DINT	0	200000000
value	Wert	1.0	-	LREAL	-1E+012	1E+012
internaltoTrace[2]	Interne Trace-Variablen					
internaltoTrace[3]	Interne Trace-Variablen					
internaltoTrace[4]	Interne Trace-Variablen					
minuslimitsofdynamics	Begrenzung der Dynamikwerte der Achse fue					
modulo	Modulo-Einstellungen					
motion	Aktive Motion-TO					
motionstatedata	Dynamikstatus der Achse (PO-aktuell)					
motontype	Bewegungsart der Achse	{72} linear	-	'enumaxistype' = en		
movecommand	Bearbeitungsstatus „move-Bewegung“ an der					
movingendstopcommand	Bearbeitungsstatus Fahren auf Endanschlag					
override	Override-Einstellungen					
pathsynchronmotion	Status der Synchronachsbewegung					
pluslimitsofdynamics	Begrenzung der Dynamikwerte der Achse fue					

Annex

3 Revisions

Table 3-1: Changes/Author

Version	Date/Revision
1.0	July 29, 2009
1.1	Dec. 18, 2009

PLCopen Blocks Behavior

4 Contact partner

Application Center

SIEMENS

Siemens AG
Industry Sector
I DT MC
Frauenauracher Str. 80
D-91056 Erlangen
Fax: +49-9131-98-1297
mailto: applications.erlf@siemens.com
