



Funktionsbausteine für PSx-3__ mit EtherNet/IP-Schnittstelle

halstrup-walcher GmbH

Stegener Straße 10
D-79199 Kirchzarten

Phone: +49 (0) 76 61/39 63-0
Fax: +49 (0) 76 61/39 63-99

E-Mail: info@halstrup-walcher.de
Internet: www.halstrup-walcher.de

Inhaltsverzeichnis

| | | |
|----------|--|----------|
| 1 | Sicherheitshinweise | 4 |
| | 1.1 Bestimmungsgemäße Verwendung | 4 |
| | 1.2 Symbolerklärung | 4 |
| 2 | Datenstruktur DRIVE_DATA | 5 |
| 3 | Fehlerbeschreibung (Error ID)..... | 6 |
| 4 | Beschreibung der Funktionsbausteine..... | 8 |
| | 4.1 Gemeinsamkeiten aller Funktionsbausteine | 8 |
| | 4.2 MC_Move | 11 |
| | 4.3 MC_Error..... | 14 |
| | 4.4 MC_ReadParameter..... | 16 |
| | 4.5 MC_WriteParameter | 18 |
| | 4.6 MC_Parametrization | 20 |
| | 4.7 MC_PositionParametrization | 23 |

Bedeutung der Betriebsanleitung

Diese Betriebsanleitung erläutert die Funktionsbausteine für die Positioniersysteme PSx-3__-EI (mit EtherNet/IP-Schnittstelle).

Von diesen Geräten können für Personen und Sachwerte Gefahren durch nicht bestimmungsgemäße Verwendung und durch Fehlbedienung ausgehen. Deshalb muss jede Person, die mit der Handhabung der Geräte betraut ist, eingewiesen sein und die Gefahren kennen. Die Betriebsanleitung und insbesondere die darin gegebenen Sicherheitshinweise müssen sorgfältig beachtet werden. **Wenden Sie sich unbedingt an den Hersteller, wenn Sie Teile davon nicht verstehen.**

Der Hersteller behält sich das Recht vor, diese Funktionsbausteine weiterzuentwickeln, ohne dies in jedem Einzelfall zu dokumentieren. Über die Aktualität dieser Betriebsanleitung gibt Ihnen Ihr Hersteller gerne Auskunft.

© 2015

Das Urheberrecht an dieser Betriebsanleitung verbleibt beim Hersteller. Sie darf weder ganz noch in Teilen vervielfältigt oder Dritten zugänglich gemacht werden.

1 Sicherheitshinweise

1.1 Bestimmungsgemäße Verwendung

Die Positioniersysteme PSx-3__-EI eignen sich besonders zur automatischen Einstellung von Werkzeugen, Anschlägen oder Spindeln bei Holzverarbeitungsmaschinen, Verpackungsmaschinen, Druckmaschinen, Abfüllanlagen und bei Sondermaschinen.

Die PSx-3__-EI sind nicht als eigenständige Geräte zu betreiben, sondern dienen ausschließlich zum Anbau an eine Maschine.

1.2 Symbolerklärung

In dieser Betriebsanleitung wird mit folgenden Hervorhebungen auf die darauf folgend beschriebenen Gefahren bei der Handhabung der Anlage hingewiesen:



WARNUNG!

Sie werden auf eine Gefährdung hingewiesen, die zu Körperverletzungen bis hin zum Tod führen kann, wenn Sie die gegebenen Anweisungen missachten.



ACHTUNG!

Sie werden auf eine Gefährdung hingewiesen, die zu einem erheblichen Sachschaden führen kann, wenn Sie die gegebenen Anweisungen missachten.



INFORMATION!

Sie erhalten wichtige Informationen zum sachgemäßen Betrieb.

2 Datenstruktur DRIVE_DATA

Für jeden Antrieb gibt es eine Datenstruktur, in der einige Daten eines Antriebs abgelegt sind. Für jeden Antrieb wird eine globale Instanz dieser Struktur benötigt. Diese Instanz muss jedem FB übergeben werden, der auf den betr. Antrieb wirkt. Hiermit soll z.B. sichergestellt werden, dass nicht gleichzeitig zwei Zugriffe aus unterschiedlichen FBs auf den Parameterkanal durchgeführt werden können.

| Parametername | Datentyp | geschrieben von | Beschreibung |
|---------------|------------|---------------------|---|
| Name | STRING[16] | Benutzer (optional) | Name der Achse |
| Beschreibung | STRING[32] | Benutzer (optional) | Beschreibung (z.B. Funktion, Aufgabe dieser Achse) |
| | | | |
| State | DINT | Funktionsbausteine | Actual state |

Beispiel:

Im Projekt befinden sich drei Antriebe. Jeder Antrieb soll über einen FB MC_Move angesteuert werden, außerdem soll mit MC_Error der Status jedes Antriebs ermittelt werden können und pro Antrieb sollen beliebige Schreib- und Lesezugriffe ausgeführt werden können.

Dazu sind insgesamt drei globale Variablen des Typs DRIVE_DATA notwendig, diese müssen bei den Controller Tags generiert werden:

- Drive_1
- Drive_2
- Drive_3

Für die Ausführung der benötigten Funktionen wird pro Antrieb auch je eine Instanz der FBs MC_Move, MC_Error, MC_ReadParameter und MC_WriteParameter benötigt:

- MC_Move_1, MC_Error_1, MC_Read_1, MC_Write_1:
VAR_IN_OUT „Drive“ → „Drive_1“ eintragen
- MC_Move_2, MC_Error_2, MC_Read_2, MC_Write_2:
VAR_IN_OUT „Drive“ → „Drive_2“ eintragen
- MC_Move_3, MC_Error_3, MC_Read_3, MC_Write_3:
VAR_IN_OUT „Drive“ → „Drive_3“ eintragen

3 Fehlerbeschreibung (Error ID)

Nachfolgend die Fehlercodes, die von den Funktionsbausteinen ausgegeben werden:

| ErrorID (hex) | Beschreibung |
|-----------------------|---|
| 16xF000 (mask) | FB |
| 16#1xxx | Error in MC_Move |
| 16#2xxx | Error in MC_Error |
| 16#3xxx | Error in MC_ReadParameter |
| 16#4xxx | Error in MC_WriteParameter |
| 16#5xxx | Error in MC_Parametrization |
| 16#6xxx | Error in MC_PositionParametrization |
| 16#0F00 (mask) | Internal FB and PD errors |
| 16#x1xx | Error in state machine or other FB internal error |
| 16#x6xx | Unallowed input data change |
| 16#x7xx | Connection Faulted |
| 16#00F0 (mask) | Parameter errors |
| 16#xx1x | Parameter: communication timeout (1000 ms) |
| 16#xx2x | Parameter: invalid parameter number |
| 16#xx3x | Parameter: value is read only |
| 16#xx4x | Parameter: lower or upper limit exceeded |
| 16#xx5x | Parameter: faulty sub-index |
| 16#xx6x | Parameter: not an array |
| 16#xx7x | Parameter: incorrect data type |
| 16#xx8x | Parameter: setting not allowed (resetting only) |
| 16#xx9x | Parameter: request cannot be processed due to operating state |
| 16#xxAx | Other error |
| 16#000F (mask) | Drive errors |
| 16#xxx1 | Drag error |
| 16#xxx2 | Under- or overvoltage motor supply |
| 16#xxx3 | Positioning run aborted |
| 16#xxx4 | Temperature exceeded |
| 16#xxx5 | Absolute measuring system error |
| 16#xxx6 | Block or overcurrent error |
| 16#xxx7 | Manual displacement |
| 16#xxx8 | Incorrect target value |
| 16#xxx9 | Under- or overvoltage during run |
| 16#xxxA | Lower position limit exceeded |
| 16#xxxB | Upper position limit exceeded |

Die Fehler „Drive errors“ sind eine Abbildung der Fehlerbits im Statuswort des PSx.

Beispiele:

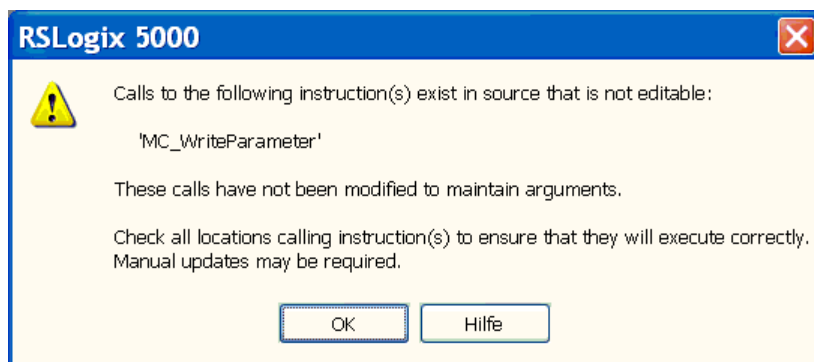
- Fahrauftrag (MC_Move) mit falschem Sollwert → ErrorID = 16#1008

- Parameter schreiben (MC_WriteParameter) mit ungültiger Parameternummer → ErrorID = 16#4020

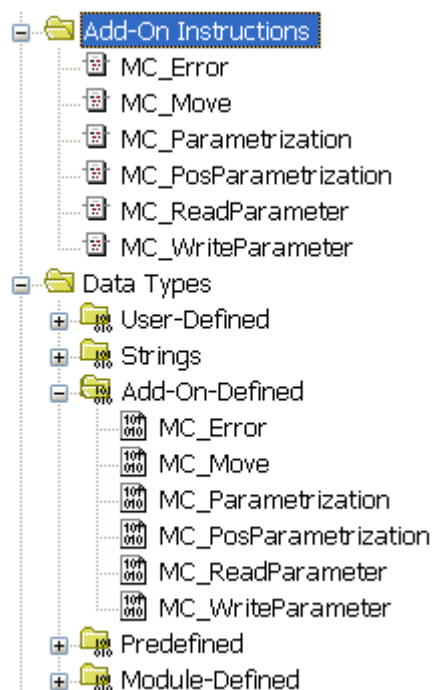
4 Beschreibung der Funktionsbausteine

Zunächst müssen die Bausteine in ein eigenes RSLogix-Projekt eingebunden werden. Dies geschieht im Controller Organizer mit Rechtsklick auf „Add-On Instructions“, dann „Import Add-On Instruction“ und die gewünschten Funktionsbausteine einzeln importieren.

Evtl. erscheint folgende Meldung, die ignoriert werden kann:



Im Ergebnis präsentieren diese sich dann folgendermaßen im Controller Organizer:



4.1 Gemeinsamkeiten aller Funktionsbausteine

Die FBs nutzen jeweils einen Teil der folgenden Variablen, die alle vom Typ VAR_IN_OUT sind. Diese Ein- und Ausgänge müssen in jedem Fall beschaltet werden, ansonsten gibt es beim Download Fehlermeldungen.

Drive

Referenz auf den gewünschten Antrieb (s. auch Kap. 2)

- Typ: DRIVE_DATA
- Art: VAR_IN_OUT

ErrorDescription

Fehlerbeschreibung

- Typ: STRING
- Defaultwert: „“
- mit beliebiger String-Variable aus den Controller Tags verbinden

INPUT_ConnectionFaulted

Zeigt an, ob die Eingangsdaten des betr. Antriebs gültig sind

- Typ: BOOL
- mit I.ConnectionFaulted des betr. Antriebs verbinden

INPUT_Status_Word

Statuswort des betr. Antriebs

- Typ: INT
- mit I.status_word des betr. Antriebs verbinden

INPUT_Actual_Position

Istposition des betr. Antriebs

- Typ: DINT
- mit I.actual_position des betr. Antriebs verbinden

INPUT_Parameter_ID

Parameternummer des Parameterinterface des betr. Antriebs

- Typ: INT
- mit I.Parameter_ID des betr. Antriebs verbinden

INPUT_Parameter_value

Parameterwert des Parameterinterface des betr. Antriebs

- Typ: DINT
- mit I.Parameter_value des betr. Antriebs verbinden

INPUT_Subindex

Array-Subindex des Parameterinterface des betr. Antriebs

- Typ: INT
- mit I.Subindex des betr. Antriebs verbinden

OUTPUT_Control_Word

Steuerwort des betr. Antriebs

- Typ: INT
- mit O.control_word des betr. Antriebs verbinden

OUTPUT_Target_Position

Sollposition des betr. Antriebs

- Typ: DINT
- mit O.target_position des betr. Antriebs verbinden

OUTPUT_Parameter_ID

Parameternummer des Parameterinterface des betr. Antriebs

- Typ: INT
- mit O.Parameter_ID des betr. Antriebs verbinden

OUTPUT_Parameter_value

Parameterwert des Parameterinterface des betr. Antriebs

- Typ: DINT
- mit O.Parameter_value des betr. Antriebs verbinden

OUTPUT_Subindex

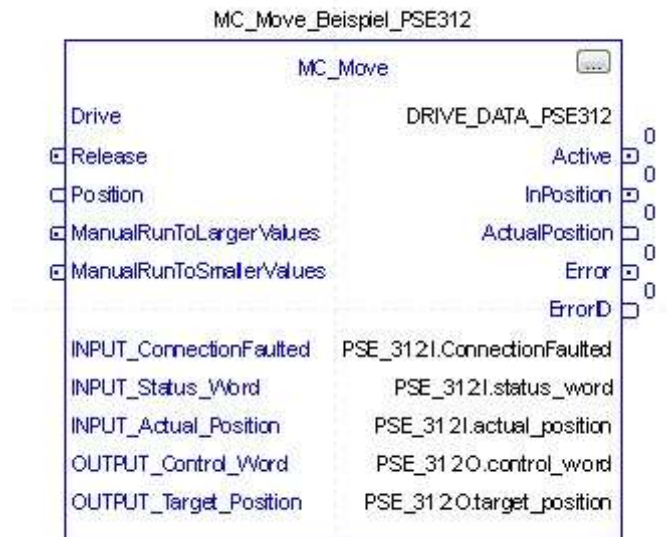
Array-Subindex des Parameterinterface des betr. Antriebs

- Typ: INT
- mit O.Subindex des betr. Antriebs verbinden

Diese Variablen vom Typ VAR_IN_OUT sind in den folgenden Beschreibungen der einzelnen FBs nicht mehr gesondert aufgeführt.

4.2 MC_Move

Dieser FB dient der Positionierung des Antriebs.



Release

Freigabe des Antriebs

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

- Ein Sollwert wird erst angefahren, wenn dieser Eingang gesetzt ist.
- Dieser Eingang wirkt direkt auf das Freigabebit (Bit 4) im Steuerwort. Bleibt der Eingang gesetzt und ist z.B. das Nachregeln im Antrieb aktiv, so regelt der Antrieb automatisch nach.
- Ist der Eingang gesetzt und wird der Sollwert geändert, so fährt der Antrieb diesen sofort an. Eine Flanke ist nicht erforderlich.
- Wird der Eingang während der Fahrt zurückgesetzt, stoppt der Antrieb.

Position

Anzufahrender Sollwert

- Typ: DINT
- Anfangswert: 0
- Art: VAR_INPUT

Beschreibung:

- Wird während einer Fahrt eine neue Sollposition übertragen, wird diese sofort angefahren.
- Ist nach Fahrtende das Release-Bit noch gesetzt und wird der Sollwert geändert, so fährt der Antrieb diesen sofort an.



INFORMATION!

Um den gleichen Sollwert z.B. nach einem Blockieren anzufahren, muss die Freigabe „Release“ zurückgesetzt und erneut gesetzt werden.

ManualRunToLargerValues

Handfahrt zu größeren Werten

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

- Handfahrt zu größeren Werten bis zum oberen Endschalter.
- Der Eingang „Release“ muss zusätzlich gesetzt sein/werden.



ACHTUNG!

Beim Zurücksetzen des Eingangs „ManualRunToLargerValues“ muss auch der Release-Eingang zurückgesetzt werden, da der Antrieb ansonsten den Sollwert (FB-Eingang „Position“) anfährt.

ManualRunToSmallerValues

Handfahrt zu kleineren Werten

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

- Handfahrt zu kleineren Werten bis zum unteren Endschalter.
- Der Eingang „Release“ muss zusätzlich gesetzt sein/werden.



ACHTUNG!

Beim Zurücksetzen des Eingangs „ManualRunToSmallerValues“ muss auch der Release-Eingang zurückgesetzt werden, da der Antrieb ansonsten den Sollwert (FB-Eingang „Position“) anfährt.

Active

Fahrauftrag bzw. Fahrt ist aktiv

- Typ: BOOL
- Art: VAR_OUTPUT

Dieser Ausgang wird gesetzt, wenn:

- die Freigabe („Release“) von 0 auf 1 gesetzt wird
- die Freigabe („Release“) schon vorhanden ist und sich der Sollwert ändert
- das Bit „Antrieb läuft“ im Status des Antriebs gesetzt ist (z.B. beim Nachregeln des Antriebs)

Dieser Ausgang wird zurückgesetzt, wenn:

- am Ende einer Fahrt das Bit „Antrieb läuft“ im Status des Antriebs nicht mehr gesetzt ist
- ein Kommunikationsfehler auftritt

InPosition

Sollposition erreicht

- Typ: BOOL
- Art: VAR_OUTPUT

Dieser Ausgang ist eine Abbildung des Statusbits „Sollposition erreicht“. Falls ein Kommunikationsfehler auftritt, wird er zurückgesetzt.

Actual position

Istwert der Position

- Typ: DINT
- Art: VAR_OUTPUT

Dieser Wert ist eine Abbildung der Istposition. Falls ein Kommunikationsfehler auftritt, wird der Wert auf 0 gesetzt.

Error

Fehler bei der Ausführung des FB oder Fehler im Antrieb

- Typ: BOOL
- Art: VAR_OUTPUT

Das Fehlerbit kann auch gesetzt sein, während der Antrieb fährt (z.B. Schleppfehler).

ErrorID

Fehler-ID

- Typ: INT
- Art: VAR_OUTPUT

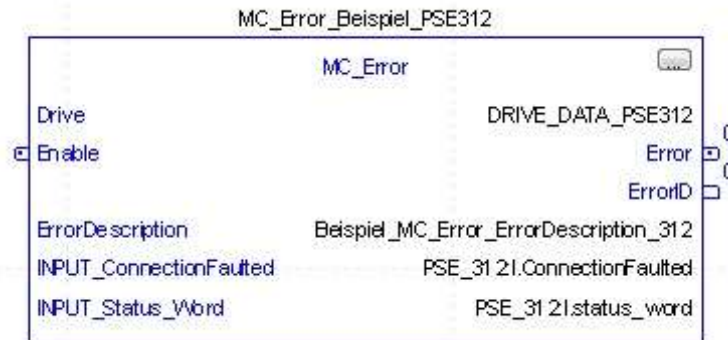
Die ErrorID kann auch gesetzt sein, während der Antrieb fährt (z.B. Schleppfehler). Falls kein Fehler vorliegt, wird 0 ausgegeben.

Falls der Antrieb mehrere Fehler meldet, wird die ErrorID mit der höchsten Priorität ausgegeben. Die Priorität der Ausgabe entspricht der Reihenfolge in der folgenden Tabelle (höchste Prio hat 16#x1xx):

| ErrorID | Beschreibung |
|---------|------------------------------------|
| 16#x1xx | FB internal error |
| 16#x2xx | Invalid PD input address |
| 16#x3xx | Invalid PD output address |
| 16#x4xx | Error while reading PD |
| 16#x5xx | Error while writing PD |
| 16#xxx2 | Under- or overvoltage motor supply |
| 16#xxx4 | Temperature exceeded |
| 16#xxx5 | Absolute measuring system error |
| 16#xxx8 | Incorrect target value |
| 16#xxx9 | Under- or overvoltage during run |
| 16#xxx6 | Block or overcurrent error |
| 16#xxx7 | Manual displacement |
| 16#xxxA | Lower position limit exceeded |
| 16#xxxB | Upper position limit exceeded |
| 16#xxx3 | Positioning run aborted |
| 16#xxx1 | Drag error |

4.3 MC_Error

Dieser FB gibt den Status des Antriebs und des FBs als Fehlerbit, Fehler-ID („ErrorID“) und als Text aus.



Enable

Die Ausgänge Error, ErrorID und ErrorDescription werden ständig vom Antrieb aktualisiert, solange Enable gesetzt ist. Wird das Enable zurückgesetzt, so nehmen diese Ausgänge die angegebenen Defaultwerte an.

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Error

Fehler bei der Ausführung des FB oder Fehler im Antrieb

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

ErrorID

Fehler-ID (siehe folgende Tabelle „ErrorID“)

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

ErrorDescription

Fehlerbeschreibung als Text

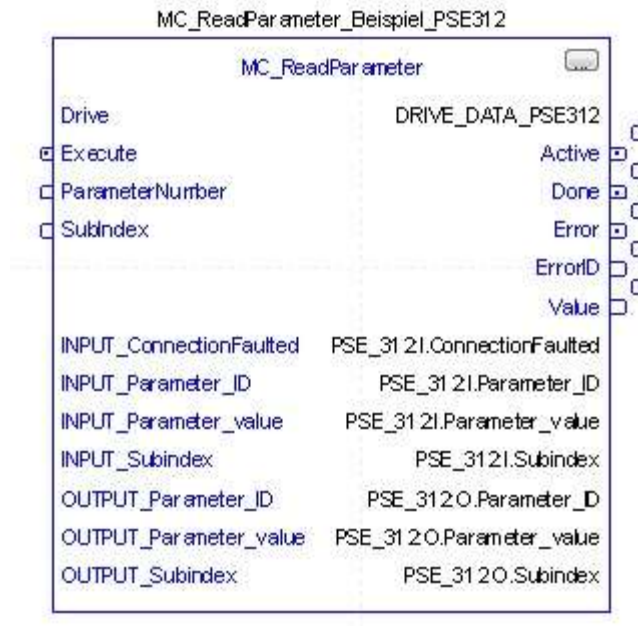
- Typ: STRING
- Defaultwert: „“
- Art: VAR_IN_OUT

Die Priorität der Ausgabe entspricht der Reihenfolge in der folgenden Tabelle (höchste Priorität hat 16#x1xx).

| ErrorID | ErrorDescription |
|---------|------------------------------------|
| 16#x1xx | FB internal error |
| 16#x2xx | Invalid PD input address |
| 16#x4xx | Error while reading PD |
| 16#xxx2 | Under- or overvoltage motor supply |
| 16#xxx4 | Temperature exceeded |
| 16#xxx5 | Absolute measuring system error |
| 16#xxx8 | Incorrect target value |
| 16#xxx9 | Under- or overvoltage during run |
| 16#xxx6 | Block or overcurrent error |
| 16#xxx7 | Manual displacement |
| 16#xxxA | Lower position limit exceeded |
| 16#xxxB | Upper position limit exceeded |
| 16#xxx3 | Positioning run aborted |
| 16#xxx1 | Drag error |

4.4 MC_ReadParameter

Mit diesem FB können Werte von Parametern aus dem Antrieb ausgelesen werden. Alle Parameter außer Par. 23 („Gerätetyp als String“) können gelesen werden.



Execute

Start eines Lesevorgangs

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

Bei einer steigenden Flanke wird ein Lesevorgang des mit „ParameterNumber“ und „Subindex“ spezifizierten Parameters gestartet. Für einen erneuten Lesevorgang muss erneut eine steigende Flanke generiert werden. Wird das Bit zurückgesetzt, so nehmen die Ausgänge die angegebenen Defaultwerte an.

ParameterNumber

Parameternummer des auszulesenden Parameters

- Typ: INT
- Anfangswert: 0
- Art: VAR_INPUT

SubIndex

Subindex des Parameters

- Typ: INT
- Anfangswert: 0
- Art: VAR_INPUT

Active

Bit ist gesetzt, solange der Lesevorgang läuft

- Typ: BOOL

- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird zurückgesetzt, sobald der Wert gelesen wurde oder ein Fehler aufgetreten ist.

Done

Bit ist gesetzt, sobald der Parameter erfolgreich gelesen wurde und an „Value“ anliegt

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird beim Start eines Lesevorgangs zurückgesetzt.

Error

Bit ist gesetzt, wenn während der Ausführung des FBs ein Fehler aufgetreten ist

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

ErrorID

Fehler-ID (siehe Tabelle „ErrorID“ in Kap. 3)

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Antriebsfehler („Drive errors“) werden beim Lesen eines Parameters nicht beachtet.

Value

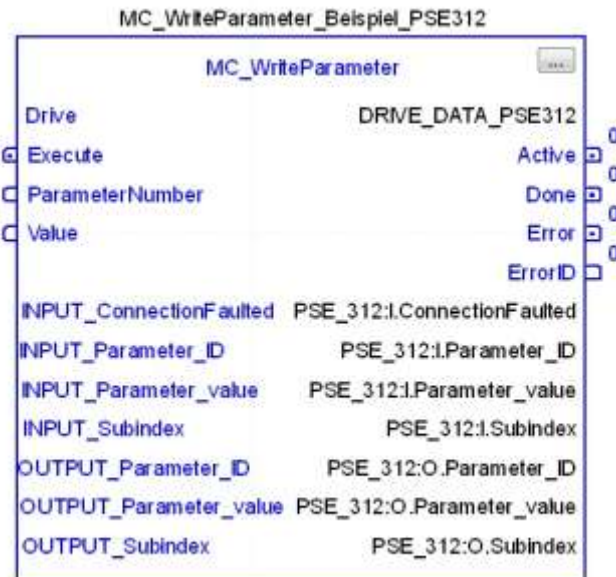
Istwert des ausgelesenen Parameters

- Typ: DINT
- Defaultwert: 0
- Art: VAR_OUTPUT

Bei einem Fehler wird 0 ausgegeben.

4.5 MC_WriteParameter

Mit diesem FB können Parameterwerte in den Antrieb geschrieben werden.



Execute

Start eines Schreibvorgangs

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

Bei einer steigenden Flanke wird ein Schreibvorgang des mit „ParameterNumber“ und „Subindex“ spezifizierten Parameters mit dem Wert aus dem Eingang „Value“ gestartet. Für einen erneuten Schreibvorgang muss erneut eine steigende Flanke generiert werden. Wird das Bit zurückgesetzt, so nehmen die Ausgänge die angegebenen Defaultwerte an.

ParameterNumber

Parameternummer des zu schreibenden Parameters

- Typ: INT
- Anfangswert: 0
- Art: VAR_INPUT

Value

zu schreibender Wert des betr. Parameters

- Typ: DINT
- Anfangswert: 0
- Art: VAR_INPUT

Active

Bit ist gesetzt, solange der Schreibvorgang läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird zurückgesetzt, sobald der Wert geschrieben wurde oder ein Fehler aufgetreten ist.

Done

Bit ist gesetzt, sobald der Parameter erfolgreich geschrieben wurde

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird beim Start eines Schreibvorgangs zurückgesetzt.

Error

Bit ist gesetzt, wenn während der Ausführung des FBs ein Fehler aufgetreten ist

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

ErrorID

Fehler-ID (siehe Tabelle „ErrorID“ in Kap. 3)

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Antriebsfehler („Drive errors“) werden beim Schreiben eines Parameters nicht beachtet.

4.6 MC_Parametrization

Mit diesem FB können sämtliche Parameter des Antriebs geschrieben werden.



Folgendes ist bei der Nutzung des FBs zu beachten:

- Zu jedem Parameterwert gibt es ein Enable, um festzulegen, ob der Parameter geschrieben werden soll.
Bsp.: DirRotation_Enable = 1 → DirRotation_Value wird gesetzt
- Die Reihenfolge der Schreibzugriffe erfolgt wie im FB dargestellt („DeliveryState“ → „DirRotation“ → ...).
- Wahlweise kann vor dem Setzen einzelner Parameter ein Auslieferungszustand angefordert werden. Dazu muss vor der Ausführung des FBs der Eingang „DeliveryState_113“ auf TRUE gesetzt werden. Dadurch werden die Werte aller Parameter auf den Auslieferungszustand gesetzt (zunächst ohne zu speichern).
- Wahlweise können die geschriebenen Werte am Ende auch gespeichert werden. Dazu muss vor der Ausführung des FBs der Eingang „SaveSettings_113“ auf TRUE gesetzt werden.
- Bei einem Schreibfehler eines Parameters werden die nachfolgenden Parameter nicht mehr geschrieben und es erfolgt auch kein Speichern der Werte, falls der Eingang „SaveSettings“ gesetzt ist.

Execute

Start eines Parametriervorgangs

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

Bei einer steigenden Flanke wird ein Parametriervorgang mit den angegebenen Werten gestartet. Für einen erneuten Parametriervorgang muss erneut eine steigende Flanke generiert werden. Wird das Bit zurückgesetzt, so nehmen die Ausgänge die angegebenen Defaultwerte an.

DeliveryState

Laden der Werkseinstellungen (zunächst ohne Speichern)

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

IP-Adresse und Adressvergabemethode bleiben jedoch unbeeinflusst.

x_Enable

Falls gesetzt, wird der betr. Parameter geschrieben

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

x_Value

Sollwert des Parameters

- Anfangswert: 0
- Art: VAR_INPUT

Die Parameternummer ist hinter dem Parameternamen angegeben. Der Datentyp, eine Beschreibung sowie der Wertebereich kann der Betriebsanleitung des PSx-3__-EI entnommen werden.

SaveSettings

Speichern der Einstellungen

- Typ: BOOL

- Anfangswert: FALSE
- Art: VAR_INPUT

Active

Bit ist gesetzt, solange die Parametrierung läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird zurückgesetzt, sobald die Parameterisierung erfolgreich beendet wurde oder ein Fehler aufgetreten ist.

Done

Bit ist gesetzt, sobald die Parameterisierung erfolgreich beendet wurde

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird beim Start einer Parametrierung zurückgesetzt.

Error

Bit ist gesetzt, wenn während der Ausführung des FBs ein Fehler aufgetreten ist

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

ErrorID

Fehler-ID (siehe Tabelle „ErrorID“ in Kap. 3)

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Antriebsfehler („Drive errors“) werden bei einer Parametrierung nicht beachtet.

ErrorParameter

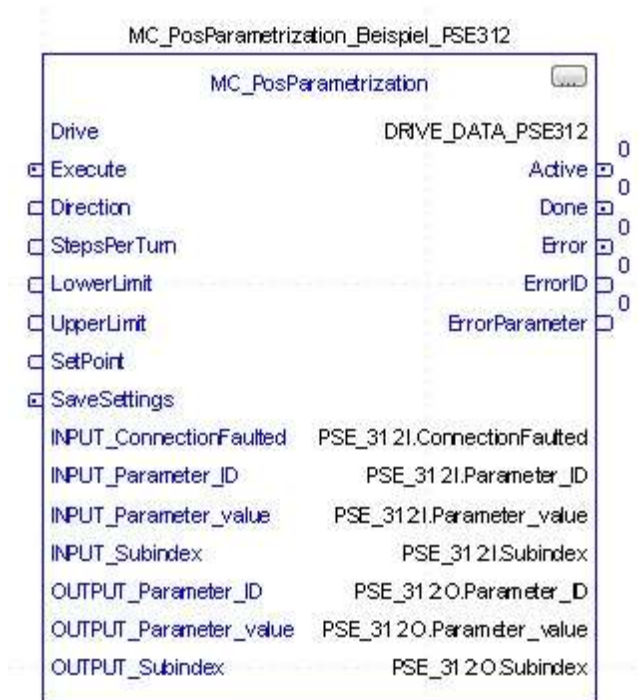
Parameternummer, bei dem im Fall eines Fehlers der Fehler aufgetreten ist

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Falls es keinen Fehler gab, wird 0 ausgegeben.

4.7 MC_PositionParametrization

Mit diesem FB kann die Parametrierung der Positionsdaten vorgenommen werden (Parameter, die die angezeigte Istposition beeinflussen).



Folgendes ist bei der Nutzung des FBs zu beachten:

- Es müssen alle Werte gesetzt werden und die Werte müssen in einem sinnvollen Bezug zueinander stehen. Alle Werte werden verarbeitet, danach werden die folgenden Parameter in der angeg. Reihenfolge geschrieben:
 - Drehsinn (Par. 26) = Direction
 - Istwertbewertung Zähler (Par. 28) = 400
 - Istwertbewertung Nenner (Par. 30) = StepsPerTurn
 - Istwert (Par. 10) = SetPoint
 - Falls (SetPoint > UpperLimit):
 - oberes Mapping-Ende (Par. 34) = SetPoint + (3 x StepsPerTurn)
 - sonst:
 - oberes Mapping-Ende (Par. 34) = UpperLimit + (3 x StepsPerTurn)
 - obere Endbegrenzung (Par. 36) = UpperLimit
 - untere Endbegrenzung (Par. 38) = LowerLimit
- Die Anzahl der Schritte pro Umdrehung „StepsPerTurn“ ergibt unmittelbar den Wert des Parameters „Istwertbewertung Nenner“ (Par. 30). Dabei wird angenommen, dass der Wert von „Istwertbewertung Zähler“ (Par. 28) im Auslieferungszustand ist, also auf 400.
- Vor dem Schreiben der Parameter werden die eingegebenen Werte auf Gültigkeit geprüft.

Nachfolgend die Bedingungen und die Fehlermeldungen, die bei nicht erfüllter Bedingung ausgegeben werden.

| Bedingung | ErrorID | ErrorParameter |
|---|---------|----------------|
| $\text{StepsPerTurn} \geq 1$ | 16#6140 | 39 |
| $\text{StepsPerTurn} \leq 10000$ | 16#6140 | 39 |
| $\text{LowerLimit} \leq \text{UpperLimit}$ | 16#6140 | 42 |
| $(\text{UpperLimit} - \text{LowerLimit}) / \text{StepsPerTurn} \leq 250$ | 16#6140 | 43 |
| Falls $\text{SetPoint} < \text{LowerLimit}$: $(\text{UpperLimit} - \text{SetPoint}) / \text{StepsPerTurn} \leq 250$ | 16#6140 | 3 |
| Falls $\text{SetPoint} > \text{UpperLimit}$: $(\text{SetPoint} - \text{LowerLimit}) / \text{StepsPerTurn} \leq 250$ | 16#6140 | 3 |

- Wahlweise können die geschriebenen Werte am Ende auch gespeichert werden. Dazu muss vor der Ausführung des FBs der Eingang „SaveSettings“ auf TRUE gesetzt werden.
- Bei einem Schreibfehler eines Parameters werden die nachfolgenden Parameter nicht mehr geschrieben und es erfolgt auch kein Speichern der Werte, falls der Eingang „SaveSettings“ gesetzt ist.

Execute

Start eines Parametriervorgangs

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Beschreibung:

Bei einer steigenden Flanke wird ein Parametriervorgang mit den angegebenen Werten gestartet. Für einen erneuten Parametriervorgang muss erneut eine steigende Flanke generiert werden. Wird das Bit zurückgesetzt, so nehmen die Ausgänge die angegebenen Defaultwerte an.

Direction

Richtung, in der der Antrieb bei größeren Werten drehen soll (bei Sicht auf die Abtriebswelle):

0 → CW, 1 → CCW

- Typ: INT
- Anfangswert: 0
- Art: VAR_INPUT

StepsPerTurn

Schritte pro Umdrehung an der Abtriebswelle (Auflösung)

- Typ: INT
- Anfangswert: 0
- Art: VAR_INPUT

LowerLimit

Untere Endbegrenzung

- Typ: DINT
- Anfangswert: 0
- Art: VAR_INPUT

UpperLimit

Obere Endbegrenzung

- Typ: DINT

- Anfangswert: 0
- Art: VAR_INPUT

SetPoint

Wert, auf den das Messsystem referenziert wird (neuer Istwert an der aktuellen Position)

- Typ: DINT
- Anfangswert: 0
- Art: VAR_INPUT

SaveSettings

Speichern der Einstellungen

- Typ: BOOL
- Anfangswert: FALSE
- Art: VAR_INPUT

Active

Bit ist gesetzt, solange die Parametrierung läuft

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird zurückgesetzt, sobald die Parameterisierung erfolgreich beendet wurde oder ein Fehler aufgetreten ist.

Done

Bit ist gesetzt, sobald die Parameterisierung erfolgreich beendet wurde

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

Das Bit wird beim Start einer Parametrierung zurückgesetzt.

Error

Bit ist gesetzt, wenn während der Ausführung des FBs ein Fehler aufgetreten ist

- Typ: BOOL
- Defaultwert: FALSE
- Art: VAR_OUTPUT

ErrorID

Fehler-ID (siehe Tabelle „ErrorID“ in Kap. 3)

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Antriebsfehler („Drive errors“) werden bei einer Parametrierung nicht beachtet.

ErrorParameter

Parameternummer, bei dem im Fall eines Fehlers der Fehler aufgetreten ist

- Typ: INT
- Defaultwert: 0
- Art: VAR_OUTPUT

Falls es keinen Fehler gab, wird 0 ausgegeben.